



SAS[®] Viya[®] 3.3 Administration

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2017. *SAS® Viya® 3.3 Administration*. Cary, NC: SAS Institute Inc.

SAS® Viya® 3.3 Administration

Copyright © 2017, SAS Institute Inc., Cary, NC, USA

All Rights Reserved. Produced in the United States of America.

For a hard copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication, or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a), and DFAR 227.7202-4, and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

December 2017

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

3.3-P1:caladmin

Contents

Chapter 1 / Orientation	1
SAS Viya Administration: Orientation	1
Chapter 2 / Initial Tasks	5
Initial Tasks in SAS Viya Administration	5
Chapter 3 / What's New in Administration of SAS Viya 3.3	7
What's New in Administration of SAS Viya 3.3: Highlights	7
What's New In Administration of SAS Viya 3.3: Details	7
Chapter 4 / SAS Viya Overview	13
Architecture of SAS Viya	13
Security in SAS Viya	17
SAS 9 and SAS Viya	22
Chapter 5 / Auditing	25
Auditing: Overview	25
Auditing: How To	26
Chapter 6 / Authentication	31
Authentication: Overview	31
Authentication: How To	31
Authentication: Concepts	42
Authentication: Guest Access	57
Chapter 7 / External Credentials Management	61
External Credentials: Overview	61
External Credentials: How To	61
External Credentials: Concepts	67
Manage Personal Passwords	68
Chapter 8 / Orientation	71
Two Authorization Systems	72
Impact of Assumable Memberships	77
Chapter 9 / CAS Authorization	79
CAS Authorization: Overview	81
CAS Authorization: How to (Authorization Window)	82
CAS Authorization: How to (CAS Server Monitor)	87
CAS Authorization: Concepts	91
CAS Authorization: Guidelines	102
CAS Authorization: Troubleshooting	103
CAS Authorization: Interfaces	104
CAS Authorization: Host Access Considerations	105
Chapter 10 / General Authorization	107
General Authorization: Overview	109
General Authorization: How to (Authorization Window)	110
General Authorization: How to (Rules Page)	114
General Authorization: Concepts	118

General Authorization: Guidelines	129
General Authorization: Troubleshooting	130
General Authorization: Interfaces	132
Chapter 11 / Backup and Restore	133
Backup and Restore: Overview	133
Backup and Restore: Terms and Concepts	133
Backup and Restore: Getting Started	136
SAS Infrastructure Data Server Binary Backup	137
Backup and Restore: Guidelines	139
Backup and Restore: Service Configuration	141
Backup: How the Process Works	143
Backups: How They Are Stored	144
Backup and Restore: Purging	145
Restore: Perform a Restore	146
Backup and Restore for Programming-Only Deployments	150
Backup and Restore: Backup Manager	152
Backup and Restore: Command-Line Interface	161
Backup and Restore: Schedule a Backup	170
Backup and Restore: Disaster Recovery	176
Backup and Restore: Troubleshooting	178
Chapter 12 / Command Line Interface	181
Command-Line Interface: Overview	182
Command-Line Interface: Preliminary Instructions	185
Command-Line Interface: Syntax	187
Command-Line Interface: Troubleshooting	192
CLI Examples: Audit	192
Command-Line Examples: CAS Authorization	193
Command-Line Examples: General Authorization	196
CLI Examples: Backup	199
CLI Examples: Restore	199
CLI Examples: CAS Administration	199
CLI Examples: Configuration	203
CLI Examples: Compute	203
CLI Examples: Folders	204
CLI Examples: Fonts	205
CLI Examples: Device Management	206
CLI Examples: Identities	207
CLI Examples: Licensing	208
CLI Examples: Job	208
CLI Examples: Reports	209
CLI Examples: Tenant Administration	210
CLI Examples: Transfer	210
Chapter 13 / Configuration Properties	215
Configuration Properties: Overview	215
Configuration Properties: How To Configure Services	215
Configuration Properties: How To Configure SAS Studio	217
Operations	218
Configuration Properties: Concepts	219
Configuration Properties: Reference (Services)	223
Configuration Properties: Reference (Applications)	242
Configuration Properties: Reference (System)	251
Configuration Properties: Interfaces	255

Chapter 14 / Content Management	257
Content Management: Overview	257
Content Management: How To	257
Content Management: Concepts	260
Chapter 15 / Promotion (Import and Export)	263
Promotion Overview	263
How To	263
Reference	270
Chapter 16 / Themes	273
Overview	273
Chapter 17 / Preferences	275
Overview	275
Concepts	275
Guidelines	275
Chapter 18 / Data Administration	277
Data Administration: Overview	277
Data Administration: How to (SAS Environment Manager)	277
Data Administration: How to (CAS Server Monitor)	286
Loading Geographic Polygon Data as a CAS Table	288
CAS Table State Management	290
Preliminary Tasks for User-Defined Formats in SAS Viya 3.3	296
Data Administration: Reference	298
Chapter 19 / SAS Data Explorer	301
Working with SAS Data Explorer	301
Making Data Available to CAS	308
Working with Data in CAS	327
Chapter 20 / SAS Cloud Analytic Services: Fundamentals	335
Introduction	335
Architecture	335
Data	338
Memory	346
Sessions	349
Security	351
Caslibs	352
Chapter 21 / QKB Management	355
Overview	355
How to Access Your QKBs	355
List QKBs	355
Import a QKB	356
View QKB Properties	357
Delete a QKB	358
Set the Default QKB and the Default Locale	358
Customize a QKB	359
Create a QKB Archive (QARC) File	359
Configure Access for CASHostAccountRequired Custom Group	360
Chapter 22 / Encryption in SAS Viya	361
Overview	361
How To	363

Concepts	407
Reference	418
Examples	447
Chapter 23 / Encryption for Data at Rest	455
Encryption for Data at Rest: Overview	455
How To (SAS Environment Manager)	455
How To (Programming Tasks)	461
Encryption for Data at Rest: Concepts	461
Reference	463
Chapter 24 / Using SAS Environment Manager	465
What Is SAS Environment Manager?	465
Accessing SAS Environment Manager	466
Using the Dashboard	466
SAS Environment Manager Functions	469
How To	473
What is Available to a Tenant Administrator?	474
Chapter 25 / Identity Management	475
Identity Management Overview	475
Getting Started with Identity Management	476
Identity Management How To	476
User Management: Guidelines and Best Practices	482
Identity Management Concepts	483
Identity Management Reference	495
User Management: Interfaces	496
Identity Management: Troubleshooting	496
Chapter 26 / SAS Licensing	499
Licensing: Overview	499
Licensing: How to (SAS Environment Manager)	500
Licensing: How To	500
Licensing: Troubleshooting	503
Licensing: Interfaces	504
Chapter 27 / Logging	505
Logging: Overview	505
Logging: How To	505
Logging: Troubleshooting	508
Logging: Reference	509
Chapter 28 / Monitoring	517
Monitoring: Overview	517
Monitoring: Concepts	517
Monitoring: How to (SAS Environment Manager)	518
Monitoring: How to (CAS Server Monitor)	526
Monitoring: How to (Grid Monitor)	529
Monitoring: How to (CAS Options)	535
Monitoring: Troubleshooting	536
Monitoring: Reference	536
Chapter 29 / Operations Infrastructure	539
Operations Infrastructure: Overview	539
How To: Operations Infrastructure Command Line	540
How To: Operations Infrastructure Agent Command Line	548

How To: ETL and Data Mart Operations	553
Operations Infrastructure Reference	554
Chapter 30 / SAS Mobile BI	557
Mobile: Overview	557
Mobile: How To	557
Mobile: Concepts	562
Mobile: Reference	565
Mobile: Troubleshooting	567
Chapter 31 / Multi-Tenancy	569
Multi-tenancy: Overview	569
Multi-tenancy: Initial Tasks	569
Multi-Tenancy: How To Manage Tenants	574
Multi-tenancy: Concepts	580
Tenant Management: Interfaces	584
Multi-tenancy: Troubleshooting	585
Chapter 32 / Scheduling	587
Scheduling: Overview	587
Scheduling: How to (SAS Environment Manager)	587
Scheduling: How to (Command Line Interface)	590
Scheduling: Command-Line Interface Reference	592
Chapter 33 / General Services	597
General Servers and Services: Overview	597
General Servers and Services: Operate	599
General Servers and Services: Locale and Encoding	601
Fault Tolerance in SAS Viya	602
General Servers and Services: Troubleshooting	604
Chapter 34 / SAS Cloud Analytic Services	607
SAS Cloud Analytic Services: Overview	607
SAS Cloud Analytic Services: How To (Scripts)	608
SAS Cloud Analytic Services: How To (SAS Environment Manager)	612
SAS Cloud Analytic Services: How To (CAS Server Monitor)	614
SAS Cloud Analytic Services: Concepts	618
Using CAS Server Monitor	625
SAS Cloud Analytic Services: Troubleshooting	626
SAS Cloud Analytic Services: Reference	627
SAS Cloud Analytic Services: Interfaces	653
Chapter 35 / SAS Server Contexts	655
SAS Viya Server Contexts: Overview	655
Server Contexts: How To	655
Server Contexts: Concepts	657
SAS Server Contexts: Interfaces	657
Chapter 36 / Programming Run-Time Servers	659
Programming Run-Time Servers: Overview	659
SAS Compute Server and Compute Service	660
SAS Launcher Server and Launcher Service	663
SAS Workspace Server and SAS Object Spawner	664
Embedded Web Application Server	668
SAS/CONNECT Server and SAS/CONNECT Spawner	670
Server Configuration Files	675

Chapter 37 / Infrastructure Servers	679
Infrastructure Servers: Overview	679
SAS Configuration Server	680
SAS Secret Manager	681
SAS Infrastructure Data Server	683
SAS Message Broker	708
SAS Cache Locator and Cache Server	710
Apache HTTP Server	711
Chapter 38 / Tuning	713
Overview	713
Tuning the Java Runtime Environment	713
Tuning the JDBC Connection Pool	714
Tuning the LDAP Connection Pool	717
Tuning the Apache HTTP Server	717
Tuning SAS Infrastructure Data Server	719
Tuning SAS Message Broker	719
Tuning the Linux Operating System	721

Orientation

SAS Viya Administration: Orientation

Documentation

This documentation supports administration of the SAS Viya 3.3 components of the SAS platform and the following products:

- SAS Visual Analytics 8.2
- SAS Visual Statistics 8.2
- SAS Visual Data Mining and Machine Learning 8.2

Other products supplement this documentation with their own product-specific administrative information, as needed.

To search this collection of documents, click . For search tips, click , and select **Help Tips**.

Note: If you see only **Orientation** in the left navigation pane, [access the entire collection](#) before you search.

Note: As a convenience, this collection includes one of the deployment guides for SAS Viya: [SAS Viya for Linux: Deployment Guide](#). See also the [SAS Viya Deployment Guides](#) page on the SAS support site.

Deployment Types

Some SAS Viya administrative tasks and tools vary by deployment type. Here is a brief description of each type:

Deployment Type	Description
Full	Includes all of the software to which you are entitled. This is the default type of deployment.
Programming-only	Excludes SAS Home, most graphical user interfaces, and most services. This is the simplest and smallest type of deployment. Note: If SAS Home is available (at your equivalent of <code>http://host/SASHome</code>), you do not have a programming-only deployment.

For details, see [“Diagrams by Deployment Type” on page 16](#).

Note: The programming-only deployment type is a convenience for sites that choose to install and use only a subset of the software. The programming-only deployment type does not correspond to a limitation in software licensing or entitlement.

Administrative Tools

Here are the main administrative tools:

Tool	Deployment Type	
	Full	Programming-Only
<i>SAS Environment Manager</i>	✓	
<i>Command-Line Interface</i>	✓	
<i>CAS Server Monitor</i>	✓	✓

Routine Ongoing Tasks

Here is a summary of routine administrative tasks:

Task	Deployment Type:		<i>Multi-tenant</i> Scope:	
	Full	Programming-Only	Provider-Level*	Intra-tenant**
<i>View logs</i>	✓	✓	✓	
<i>Monitor services</i>	✓	✓	✓	
<i>Start and stop CAS</i>	✓	✓	✓	
Update your software***	✓	✓	✓	
<i>Renew your license</i>	✓	✓	✓	
<i>Create backups</i>	✓	✓	✓	✓
<i>Add caslibs</i>	✓	✓		✓
<i>Manage access to data</i>	✓	✓		✓
<i>Manage access to content</i>	✓			✓
<i>Manage access to functionality</i>	✓			✓
<i>Promote content</i>	✓			✓
<i>Assign users to custom groups</i>	✓			✓
<i>Manage mobile devices</i>	✓			✓
<i>Schedule jobs</i>	✓			✓
<i>Start and stop microservices</i>	✓	Not applicable†	✓	

Task	Deployment Type:		Multi-tenant Scope:	
	Full	Programming -Only	Provider- Level*	Intra- tenant**
Manage tenants			✓	

* These are the most common tasks that the provider in a *multi-tenant* deployment performs.

** An administrator within a tenant can perform these tasks for that tenant.

*** See your deployment guide (for example, [SAS Viya for Linux: Deployment Guide](#)).

† SAS Viya microservices are not installed on programming-only deployment types. However, you must still restart CAS and SAS Object Spawner.

See Also

- [SAS Viya Administration: Initial Tasks](#)

Initial Tasks

Initial Tasks in SAS Viya Administration

Introduction

This topic assumes that all applicable tasks in your [deployment guide](#) have been completed.

Complete only the tasks for your [deployment type](#).

Note: If you have a [multi-tenant](#) deployment, do not use this topic. Instead, see [Multi-tenancy: Initial Tasks](#).

Initial Tasks in a Full Deployment

- 1 Familiarize yourself with the [predefined custom groups](#), and [add members](#) if needed.
- 2 If you have users who access CAS from both programming and visual interfaces, review the [host access considerations](#).
- 3 [Promote](#) any supported content from previous releases.
- 4 Address any applicable [considerations](#) for SAS 9 and SAS Viya integration.
- 5 [Create a backup](#), and set up automated backups.
- 6 Send programmers the following link: [An Introduction to SAS Viya Programming](#).
Send visual interface users the following link: [SAS Viya](#).

Initial Tasks in a Programming-Only Deployment

- 1 [Add members](#) to the Superuser role for your CAS server.
- 2 Address any applicable [considerations](#) for SAS 9 and SAS Viya integration.
- 3 [Create a backup](#) of CAS information and configuration files.
- 4 Send programmers the following link: [An Introduction to SAS Viya Programming](#).

See Also

- [“Routine Ongoing Tasks” on page 2](#)
- [SAS Viya: Overview](#)

What's New in Administration of SAS Viya 3.3

What's New in Administration of SAS Viya 3.3: Highlights

Here are the key administrative features that are introduced in SAS Viya 3.3:

- By default, data in motion is protected. You can increase the default security by replacing the default certificates with custom certificates and upgrading the default cryptography. See [Encryption in SAS Viya: Data in Motion](#).
- You can enable guest users to view reports. See [SAS Viya Administration: Authentication](#).
- You can easily script common administrative tasks. See [SAS Viya Administration: Command-Line Interfaces](#).
- You can easily import and manage batch loading of data with provided job definitions. See [SAS Viya Administration: Data](#).
- You can use a new common interface to import data. See [SAS Data Explorer: User's Guide](#).
- You can load [Geographic Polygon data](#) as a CAS table.
- You can manage your stored credentials (personal passwords). See [SAS Viya Administration: External Credentials](#).
- At installation time, you can enable support for multiple tenants in a single deployment. See [SAS Viya Administration: Multi-tenancy](#).
- In SAS Environment Manager, you can customize the administrative dashboard, view new widgets on the dashboard, and view new administrative reports. New and enhanced interfaces help you manage licenses, view logs, monitor machines, schedule jobs, perform backups, manage personal passwords, and manage user-defined formats. See [SAS Viya Administration: Using SAS Environment Manager](#).

What's New In Administration of SAS Viya 3.3: Details

This topic provides details. A [summary of highlights on page 7](#) is available.

Authorization

- You can use a command-line interface to script CAS access controls and general authorization rules.
- You can view information about the source of an effective access result.

- In CAS authorization, origins information identifies the highest precedence access control (or access controls) that cause a particular result.
- In general authorization, origins information identifies all rules that are applicable to a particular result.
- You can preview the results of your unsaved changes to CAS access controls and general authorization rules.
- In the initial configuration, only members of the SAS Administrators group can create top-level folders.
- In SAS Environment Manager, the **Rules** page provides faceted filtering and paged results.
- In SAS Environment Manager, the **Rules** page no longer supports sorting of the data within a column.
- In SAS Environment Manager, the filtering on the **Rules** page is now case-sensitive.
- In SAS Environment Manager, the Authorization windows always include a row for you, the currently connected user.
- In CAS authorization, the Access Control action set has the following new actions:

startTransaction	Initiates an access control transaction in the current client session. Within a transaction, changes are private. Within a transaction, only exclusively reserved (checked-out) objects and their children can be updated. A transaction terminates when it is committed or rolled back.
checkoutObject	Reserves an object (and all of its children) for update by only the current client session. Prevents an object (and all of its parents) from being checked out exclusively by another session if checkOutType=Shared .
commitTransaction	Persists all changes in an access control transaction to the server, releases all checked-out objects, and terminates the transaction.
rollbackTransaction	Discards all changes in an access control transaction, releases all checked-out objects, and terminates the transaction.
statusTransaction	Shows whether client session has an active transaction.
whatCheckoutsExist	Lists check-outs held on an object, its parents, and its children.
checkInAllObjects	Releases all objects that are checked out. Use this action if the current client session does not have a transaction.

- In CAS authorization, the accessControl.whatIsEffective and accessControl.listAllPrincipals actions have a new parameter, includeMyInfo.
- In CAS authorization, the Anonymous identity type is replaced with the Guest identity type.
- In CAS authorization, when you examine your own effective access information, the returned information does not reflect your CAS role status.
- In CAS authorization, [roles](#) no longer provide unrestricted access to data.
- In CAS authorization, the way that access controls for database caslibs are stored has changed. After you upgrade from SAS Viya 3.2, see [“Preserve Access Controls for Database Caslibs” in SAS Viya for Linux: Deployment Guide](#).
- In general authorization, a new principal type, Guest, facilitates guest access.
- In general authorization, the Update permission on a parent is no longer required in order to move a member to a different location.

Encryption

- You can encrypt data at rest with additional encryption options. By default, data at rest is presumed to be behind the firewall and is not encrypted.
 - To manage encryption of data files interactively, use SAS Environment Manager.
 - To programmatically encrypt data files, use the CASLIB Statement.
- For data in motion, in a full deployment of SAS Viya, all external communication paths are secured by default. You can increase the level of security provided by default in the following ways.
 - The default Apache httpd (reverse proxy server) security configuration can be strengthened. You can replace the default self-signed certificates with your own custom certificates and can strengthen the default cryptography.
 - Trusted CA certificates used for connecting to CAS are signed by HashiCorp Vault-generated root CA and intermediate certificates. These default certificates can be replaced with your own custom certificates post deployment.
 - You can take steps to enable TLS encryption between the data provider (for example, Hadoop or Teradata) and the CAS server. If you are using a SAS Data Connect Accelerator, the data that is transferred between the data provider and the CAS server is not encrypted by default.
 - You can use Ansible utilities to update certificates and distribute them to the truststores in the deployment.

Licensing

- You can use a command-line interface to query SAS license information.
- In SAS Environment Manager, the new **Licensed Products** page provides faceted filtering and paged results.

SAS Environment Manager

Dashboard

- New widgets are provided on the **Dashboard** to monitor the availability of machines, services, and service instances; to view the number of log messages from the top five producers; and to access system monitoring reports.
- You can customize the **Dashboard** by choosing which widgets and predefined system reports to display.
- You can select reports to pin to the **Dashboard** so that the report is displayed together with the predefined system reports.

Monitoring

- In SAS Environment Manager, the availability of machines, services, and service instances is displayed on the **Dashboard**.
- In SAS Environment Manager, the new **Machines** page enables you to view CPU usage and memory usage for each machine.
- In SAS Environment Manager, the **Machines** page enables you to view the status of health checks for each machine.

- In SAS Environment Manager, the **Machines** page enables you to view the status of each service running on each machine.
- In SAS Environment Manager, the **Machines** page displays the properties, system metrics, SAS packages, and system limits for each machine.
- In SAS Environment Manager, reports are provided from the **Dashboard** that enable you to view metric data for application activity, CAS activity, disk space, SAS Infrastructure Data Server tables, message queue activity, system activity, and user activity.
- An operations infrastructure monitors and collects metric data, log events, and notification messages; sends the collected information to message queues; and publishes the information to a data mart. The data is then displayed in SAS Environment Manager and can also be used to generate reports.
- Audit data is automatically collected to track access attempts for folders, data plans, CAS management, and CAS data management, and is automatically displayed in system reports.

Logging

- In SAS Environment Manager, a graph on the **Dashboard** shows the number of logged issues from the top five sources over the previous 30 minutes.
- In SAS Environment Manager, the new **Logs** page displays graphs of the number of log messages, a table of log messages, and provides controls to filter the messages by time range, message content, level, and source.

Scheduling

- In SAS Environment Manager, the new **Scheduling** page enables you to schedule a job using time-based criteria.
- In SAS Environment Manager, the **Scheduling** page enables you to run an available job immediately.

Content Management

- In SAS Environment Manager, the **Content** page enables you to export the reports in a folder to a package file.
- In SAS Environment Manager, the **Content** page enable you to import the reports in an exported package file to a folder.
- In SAS Environment Manager, the **Content** page provides the ability to pin a report to the **Dashboard** so that the report is displayed together with the predefined system reports.

Servers and Services

- There is a new [SAS Compute Server on page 661](#) that enables clients to submit SAS code.
- There is a new [SAS Launcher Server on page 663](#) that enables you to start processes in the SAS Viya environment.
- There is a new [server on page 681](#) to manage secrets.
- SAS Cloud Analytics Services (CAS) enables you to define [CAS backup controllers on page 618](#) to support fault tolerance.
- CAS has been enhanced with [session zero processing on page 622](#) to enable SAS applications that require application-specific configuration and start-up before SAS Cloud Analytic Services begins processing client requests.

- The following CAS server configuration file options have been added:
 - [cas.INITIALBACKUPS](#) on page 634
 - [cas.NODE](#) on page 640
 - [cas.STARTUP](#) on page 643
 - [cas.STARTUPDIR](#) on page 643
 - [cas.TAG](#) on page 644
 - [cas.TENANTID](#) on page 644
- The following CAS server configuration file options have been discontinued:
 - [cas.ADDFMTLIB](#)
 - [cas.FMTCASLIB](#)
 - [cas.TIMEZONE](#)
- In SAS Environment Manager, there is enhanced CAS functionality:
 - You can stop a CAS controller.
 - You can add and remove CAS worker nodes.
- In SAS Environment Manager, the **Contexts** page enables you to create [server contexts](#) for the SAS Compute and the SAS Launcher servers.
- There are new [SAS Cache Locator services on page 710](#) (cachelocator and cacheserver) that replace geodelocator and geodeserver.

Tenancy

- You can configure your deployment to comprise multiple tenants. Activating multi-tenancy is a deployment-time decision. Ensure you understand the implications of this choice before deploying.
- Creation and activation (onboarding) of tenants is achieved by running an Ansible playbook.
- As an overall administrator, you can view and manage tenants from SAS Environment Manager.
- You can create tenant-level administrators whose role is day-to-day administration of the specified tenant.
- Each tenant is completely isolated from other tenants. For example, a use-case would be a tenant for your Human Resources department, to guarantee that access to sensitive data is restricted to appropriate personnel.
- All users of all tenants must share a single LDAP server. The provider and each tenant must be described by a single OU. All tenant OUs must be peers in the LDAP system.

SAS Viya Overview

Architecture of SAS Viya	13
Introduction	13
Key Components	13
Cumulative Functionality	14
Selective Deployment (Optional)	15
Diagrams by Deployment Type	16
Security in SAS Viya	17
Authentication	17
Authorization	20
Encryption	20
Web Security	21
SAS 9 and SAS Viya	22
Summary	22
Considerations: Interacting with SAS 9 Data	22
Considerations: Accessing CAS from SAS 9.4M5	23

Architecture of SAS Viya

Introduction

This section provides a concise summary for new administrators.

Here are related topics:

- To get started with SAS Viya administration, see [SAS Viya Administration: Orientation](#).
- To learn about benefits of SAS Viya, see [SAS Viya](#) on the SAS website.

Key Components

Here are software components that might be of particular interest to administrators.

The analytics engine to SAS Viya	SAS Cloud Analytic Services: Fundamentals
A modular set of supporting services	SAS Viya Administration: General Servers and Services
A web application for basic administration	CAS Server Monitor on page 614

A web application for enterprise administration	SAS Viya Administration: Using SAS Environment Manager
A web application for writing and submitting code	Getting Started with Programming in SAS Studio
A web application for visual reporting, exploration, and modeling	SAS Visual Analytics: Overview
Multiple application programming interfaces	http://developer.sas.com


















TIP For information about other components, search the [SAS Viya administration documentation](#).

Cumulative Functionality

Among some of the products on SAS Viya, available functionality is cumulative.

- SAS Visual Analytics provides baseline functionality, including reporting and basic analytics.
- SAS Visual Statistics provides an additional set of advanced analytic functions.
- SAS Visual Data Mining and Machine Learning provides a second additional set of advanced analytic functions.

For example, if you have SAS Visual Data Mining and Machine Learning, the objects that are available in the SAS Visual Analytics web application are as follows:

- ▶ Tables
- ▶ Graphs
- ▶ Controls
- ▼ Analytics
 -  Forecasting
 -  Network Analysis
 -  Path Analysis
 -  Text Topics
- ▶ Other
- ▼ SAS Visual Statistics
 -  Cluster
 -  Decision Tree
 -  Generalized Additive Model
 -  Generalized Linear Model
 -  Linear Regression
 -  Logistic Regression
 -  Model Comparison
 -  Nonparametric Logistic Regression
- ▼ SAS Visual Data Mining and Machine L...
 -  Factorization Machine
 -  Forest
 -  Gradient Boosting
 -  Neural Network
 -  Support Vector Machine

Note: All three of the preceding products offer both programming and visual interfaces.

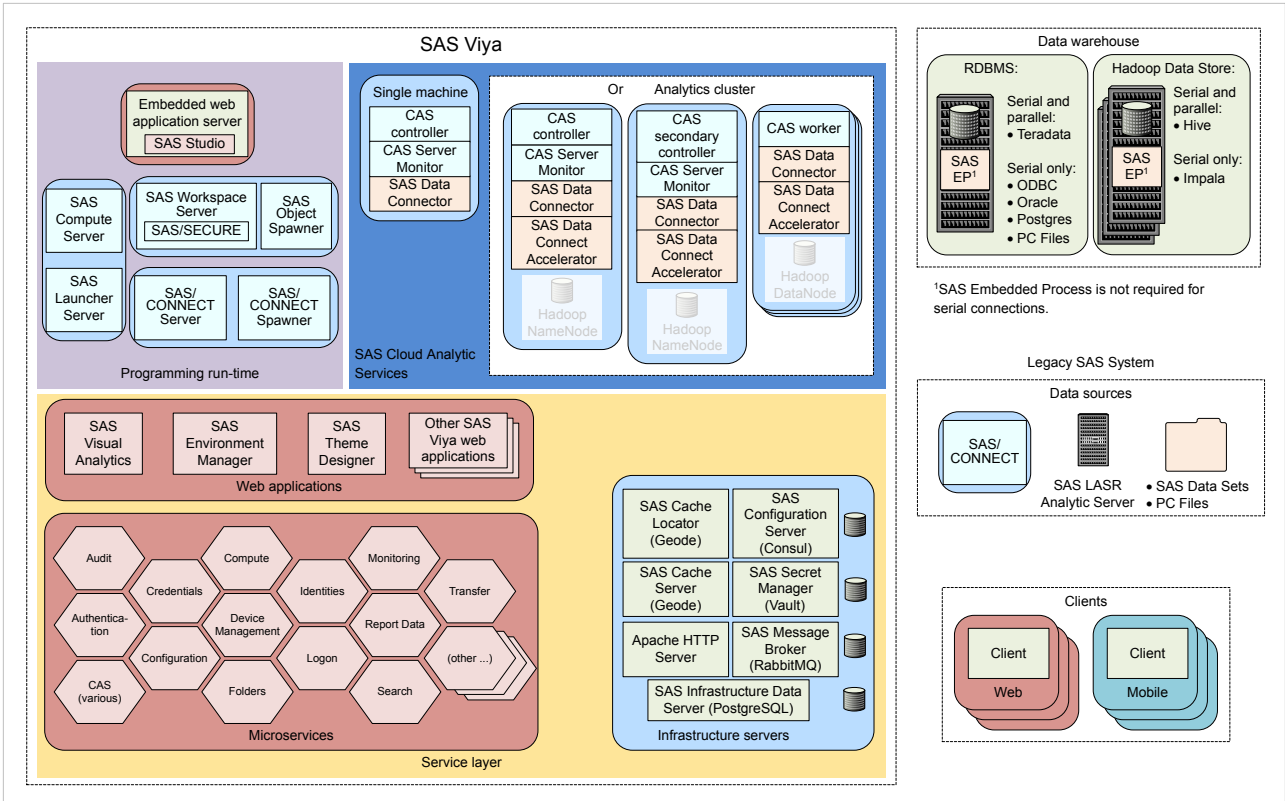
Selective Deployment (Optional)

By default, all of your software is deployed. As a convenience for special circumstances, it is possible to deploy only a subset of components. A programming-only deployment excludes general services and visual interfaces.

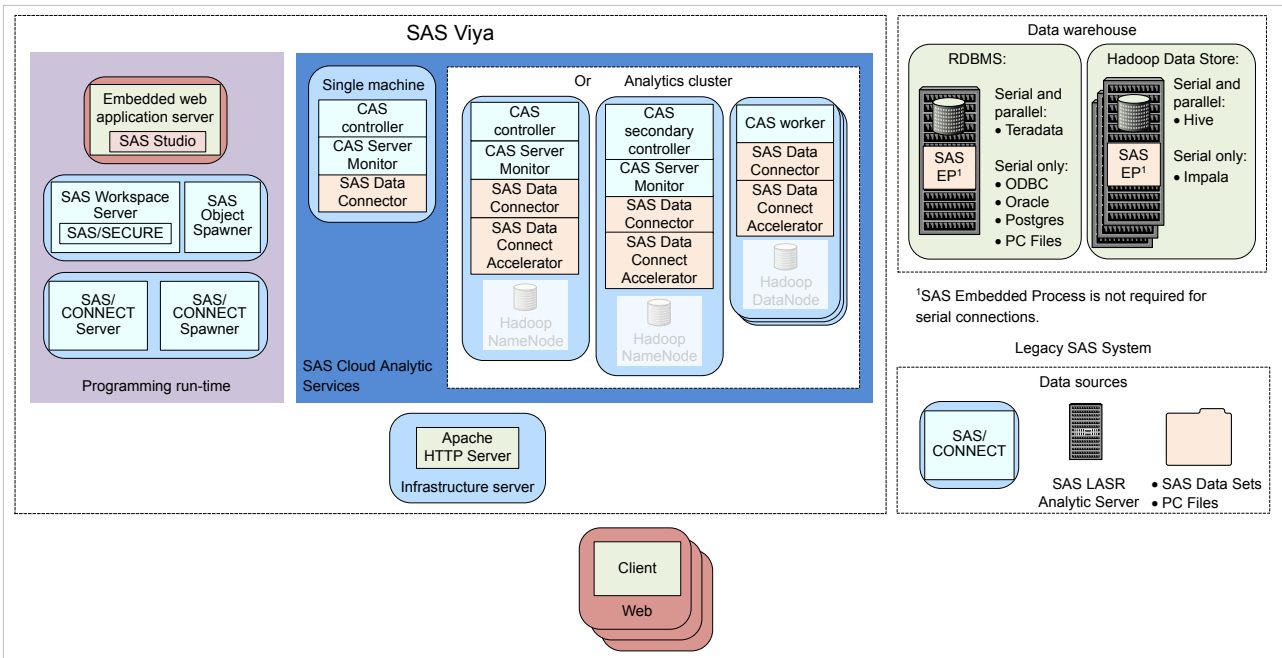
For example, a programming-only deployment of SAS Visual Analytics does not include the SAS Visual Analytics web application.

Diagrams by Deployment Type

Full Deployment (Native Operating Systems)



Programming-Only Deployment (Native Operating Systems)



Security in SAS Viya

Authentication

Authentication is the aspect of security that verifies the identity of a user or service account.

When you sign in, one of the following authentication patterns is used:

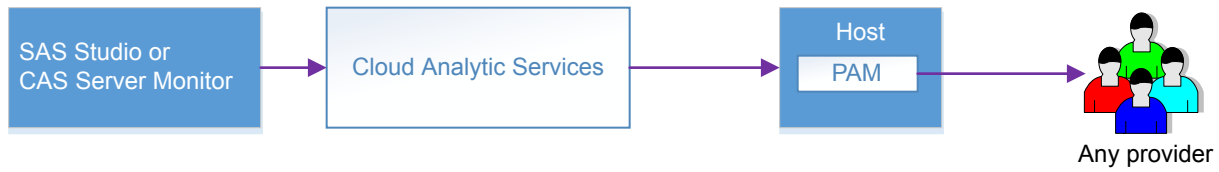
Pattern	Description	Usage
Host authentication	<p>Requests are sent to the appropriate host and processed by any authentication mechanism supported by that host.</p> <p>Programming-only deployments use this pattern exclusively. Other deployments use dual authentication for sign in to CAS Server Monitor and access to CAS from SAS Studio.</p> <p>Note:</p> <p>You can configure the host to use pluggable authentication modules (PAM). SAS provides starter PAM configuration files for CAS and SAS Studio. You can create an Authinfo file for use with PAM in command-line access to CAS.</p>	<p>When you sign in to SAS Studio, the associated object spawner asks its host (which is also the host of the SAS Studio web application) to validate your credentials. That validation enables the object spawner to launch a workspace server for you.</p> <p>When you access CAS from SAS Studio, you must authenticate to the host of the target CAS server.</p> <p>When you sign in to CAS Server Monitor, you must authenticate to the host of the target CAS server.</p>

Pattern	Description	Usage
Direct LDAP authentication	<p>Requests are sent to and processed by your designated direct LDAP provider, unless you configure front-end single sign-on using Kerberos, Open Authorization (OAuth), or Security Assertion Markup Language (SAML).</p> <p>User and group information is always obtained from your designated direct LDAP provider.</p> <p>Note:</p> <p>Kerberos, OAuth, and SAML are alternate mechanisms for identity verification by the logon service, not alternate sources of user and group information for the identities service.</p>	<p>When you sign in to a web application that uses the logon service (for example, SAS Visual Analytics or SAS Environment Manager), you must authenticate using this pattern.</p> <p>Before you can submit a command-line request to a general service (for example, the backup service or the transfer service), you must authenticate using this pattern.</p>
Dual authentication	<p>Requests are authenticated using both host authentication and direct LDAP authentication. If the servicesBaseUrl option is specified, CAS requires dual authentication.</p> <p>To facilitate this pattern, use one of these approaches:</p> <ul style="list-style-type: none"> ■ Ensure that all requests are ultimately processed by the same authentication provider. For example, configure the SAS Studio and CAS hosts to use the same LDAP provider that is designated for direct LDAP authentication requests in your deployment. ■ Ensure that each affected user has a single set of credentials that are valid for all applicable authentication providers. 	<p>In a full deployment, dual authentication occurs for sign in to CAS Server Monitor and access to CAS from SAS Studio.</p> <p>Note:</p> <p>When you access CAS from a web application such as SAS Visual Analytics or SAS Environment Manager, your OAuth token is validated.</p>

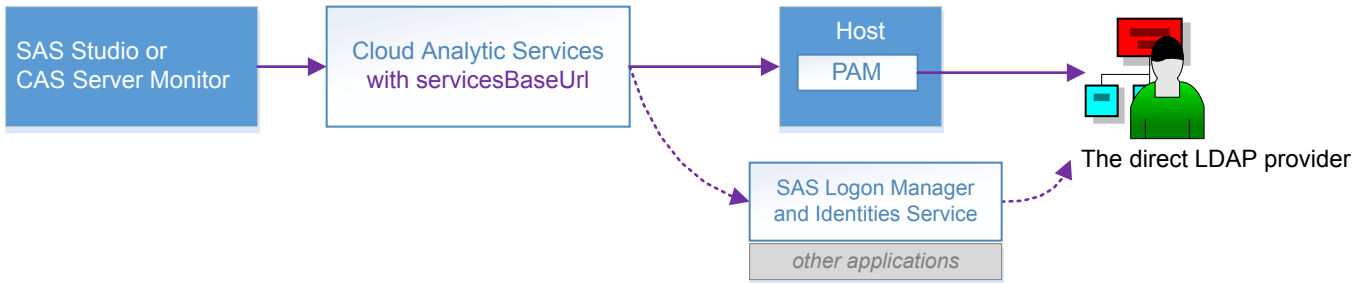
The following high-level conceptual drawings illustrate key points from the preceding table:

Figure A.1 Authentication from SAS Studio or CAS Server Monitor to CAS

Host Authentication



Dual Authentication: Shared Provider



Dual Authentication: Different Providers

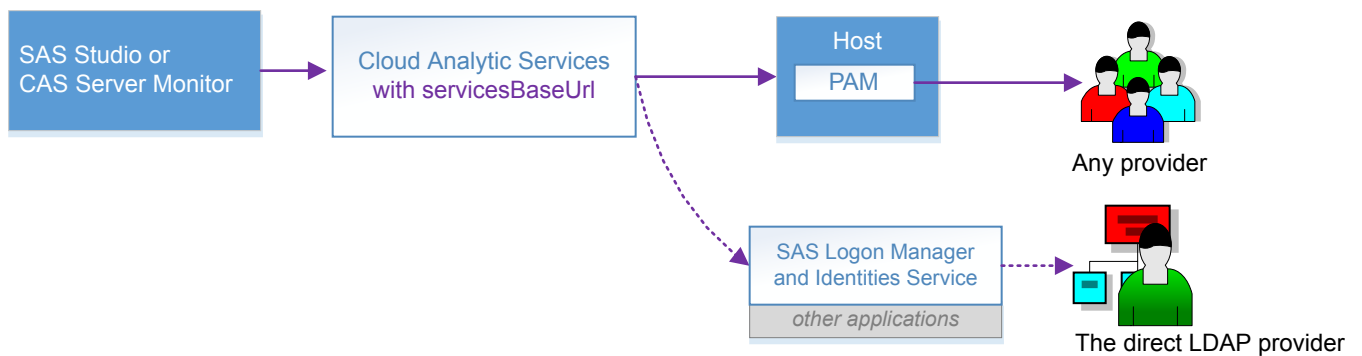
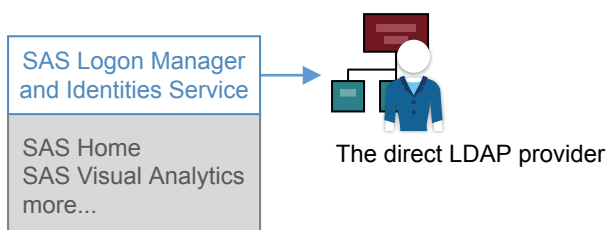
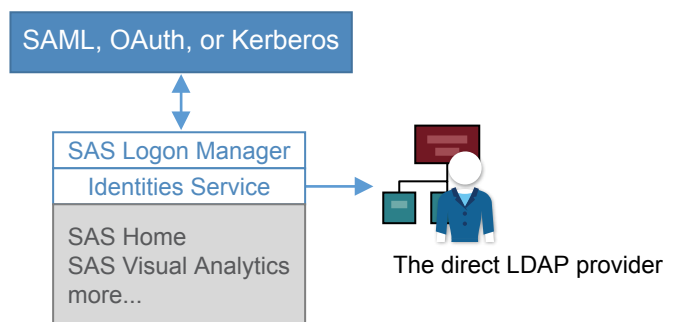


Figure A.2 Authentication from Other Applications

Default: Direct LDAP



Alternative: Front-End Single Sign-On



After you sign in, you have seamless access to SAS Viya and, in some contexts, to external data sources.

For more information, see the following documents:

[SAS Viya Administration: Authentication](#)

[SAS Viya Administration: Identity Management](#)

Authorization

Authorization is the aspect of security that determines which resources are available to which users. The SAS Viya authorization layer consists of two authorization systems:

- CAS authorization system
- general authorization system

Each system uses a distinct model to protect a distinct class of resources. The general authorization system is not applicable in a programming-only deployment.

Initial and default access are restrictive:

- Any access that is not granted is implicitly disallowed.
- Predefined objects are protected by predefined rules or access controls.
- Only members of special groups or roles have access to privileged administrative functionality.
- Access to objects that users add is managed by inheritance, other influencing rules, and any direct settings.
- Regular users have limited Write access. They can write to their personal folder, the shared Public folder, and the shared Public caslib.

For more information, see the following documents:

[SAS Viya Administration: Orientation to Authorization](#)

[SAS Viya Administration: Cloud Analytic Services Authorization](#)

[SAS Viya Administration: General Authorization](#)

[SAS Viya Administration: Identity Management](#)

Encryption

Encryption is the aspect of security that protects data by converting it into an unintelligible form in transmission or in storage.

For data in motion in a new deployment, TLS security is provided and follows the highest standards. At installation SAS Viya provides self-signed certificates to provide HTTP and HTTPS access to SASHome out of the box. You can increase the encryption strength and coverage by completing additional configuration.

For data at rest in a new deployment, encryption is not automatically enabled. You can configure encryption of data that is added to PATH, HDFS, and DNFS caslibs.

For more information, see the following documents:

[Encryption in SAS Viya: Data in Motion](#)

[Encryption in SAS Viya: Data at Rest](#)

Web Security

Web security is the aspect of security that deals with securing against certain types of attacks on web applications and utilizing the security features that are available in modern web browsers.

SAS Viya provides properties that are configured, by default, to protect against the web security risks that are listed below. You can disable or change the properties, based on your environment. For example, you might have to configure Cross-Origin Resource Sharing (CORS) to allow origins in your company's domain. This allows SAS web pages to be included in other web pages inside your company's network.

For more information about the SAS Viya configuration properties, see the following:

Property	Description	Default Settings
Cross-Origin Resource Sharing on page 253	Technique for relaxing the browser same-origin policy, allowing Javascript on a web page to consume a REST API served from a different origin.	The following cross-origin requests are configured: <ul style="list-style-type: none"> ■ User credentials are used ■ All HTTP headers are allowed ■ All HTTP methods are allowed ■ Same origins are allowed
Cross-Site Request Forgery (CSRF) on page 253	Prevents attacks that force a user to execute unwanted actions on a web application in which they are currently authenticated.	The following options are configured: <ul style="list-style-type: none"> ■ Referers internal to the deployment are allowed ■ Requests are not blocked if both the origin and referer headers are absent
X-Frame-Options on page 252	Avoids clickjacking attacks by making sure that your content is not embedded in other sites.	Same origin
Content-Security-Policy on page 252	Exposes and reduces the risk of data injection and cross-site scripting (XSS) attacks.	default-src 'self'; script-src 'self' 'unsafe-inline' 'unsafe-eval'; img-src 'self' *.sas.com blob: data:; style-src 'self' 'unsafe-inline'; child-src 'self' blob: data: mailto;;
X-Content-Type-Options on page 252	Prevents the browser from interpreting files as something other than what is declared by the content type in the HTTP headers (content sniffing).	nosniff
X-XSS-Protection on page 252	Stops web browser from loading pages when XSS attacks are detected.	1; mode=block

For information about these web attacks, see the following OWASP pages:

- [Category:Attack](#)

- [OWASP Secure Headers Project](#)

SAS 9 and SAS Viya

Summary

SAS 9 customers continue to benefit from their investment in SAS 9 as they begin to make use of SAS Viya functionality and features. From within familiar SAS 9 interfaces, projects, and code, customers can access the performance enhancements that SAS Viya provides.

- On most hosts, SAS 9.4M5 is tightly integrated with SAS Viya. See [SAS 9.4M5 Integration with SAS Viya](#) in *What's New in Base SAS: Details*. (The exceptions are z/OS and 32-bit Windows.)
- All releases of SAS can use SAS/CONNECT as a bridge to SAS Viya. See the appendix [Sharing Data Between SAS 9 and SAS Viya using SAS/CONNECT](#) in *SAS/CONNECT for SAS Viya User's Guide*.

Here are some of the methods for accessing SAS 9 data from SAS Viya:

- In SAS Visual Analytics, use self-service import. See [SAS Data Explorer: User's Guide](#).
- In SAS Environment Manager, interactively load data. See [Data Administration: How to \(SAS Environment Manager\)](#) in *SAS Viya Administration: Data*.
- In SAS Enterprise Guide or SAS Add-In for Microsoft Office (7.13 or later), move data from SAS 9 to CAS. See the topic "Configure Your Environment to Use the Upload to CAS Task" in the [SAS Enterprise Guide](#) or [SAS Add-In for Microsoft Office](#) chapter in *SAS Intelligence Platform: Desktop Application Administration*.
- In any programming interface, write code to load data. See [Programming Interfaces](#) in *An Introduction to SAS Viya Programming*.
- If a more seamless method is not available, use SAS/CONNECT for SAS 9 and SAS Viya to move and share data. See the appendix [Sharing Data Between SAS 9 and SAS Viya using SAS/CONNECT](#) in *SAS/CONNECT for SAS Viya User's Guide*.

Note: Not all deployments and releases include all products and support all methods.

Note: Your site must license and install SAS Viya to access SAS Viya functionality. By default, when you order SAS Viya, you receive SAS Visual Analytics. All analytical procedures are separate licenses: SAS Econometrics Procedures, SAS Optimization Procedures, SAS Forecasting Procedures, SAS Visual Data Mining and Machine Learning Procedures, SAS Statistics Procedures, and SAS Viya Procedures.

Considerations: Interacting with SAS 9 Data

Use UTF-8 Encoding

If you access SAS 9 data from SAS Viya, be aware that SAS Viya operates with UTF-8 encoded data. If your SAS 9 data is not UTF-8 encoded, you might need to re-create your data sets. See [Migrating to UTF-8 for SAS Viya](#).

Manage User-Defined Formats

If you access SAS 9 data from SAS Viya, you must make any user-defined formats available to your CAS session. See [SAS Cloud Analytic Services: User-Defined Formats](#).

Considerations: Accessing CAS from SAS 9.4M5

Find CAS

If a SAS 9.4M5 client session cannot find CAS, make information about the host and port of the CAS server available. For example, add the following line to your SAS Application Server `sasv9_usermods.cfg` or `appserver_autoexec_usermods.sas` file:

```
CASHOST=("primary-controller-host-name" <"backup-controller-host-name">) CASPORT=port;
```

Here is an example with a CAS backup controller:

```
CASHOST=("mysrv01" "mysrv02") CASPORT=5570;
```

Here is an example without a CAS backup controller:

```
CASHOST=("mysrv01") CASPORT=5570;
```

For more information, see [CASHOST= System Option](#).

Authenticate to CAS

If a SAS 9.4M5 client session cannot authenticate to CAS, create an `authinfo` file, store CAS credentials in the SAS 9 metadata, or use a different authentication mechanism. See [SAS Viya Administration: Authentication](#).

Conform to CAS Encryption Requirements

If a SAS 9.4M5 client session does not meet the encryption standards of the CAS server, make an appropriate certificate available. See ["Configure SAS 9.4 Clients to Work with SAS Viya" on page 389](#).

Auditing

Auditing: Overview

An audit record is generated whenever these types of events occur:

- an action is performed on a resource (such as a folder or a job). Actions include access to the resource and any changes made to the resource (such as updating, creation, or deletion).
- a security-related action occurred, such as logging on to an application or changing an authorization rule

By default, these actions generate audit records:

- resource read failure
- resource created, updated, or deleted
- security actions (logon attempts, logoff attempts, accessing authorization rules, updating authorization rules)

See [“Change Auditing Configuration” on page 29](#) for information about changing the actions that generate an audit entry.

The audit records are stored in the SAS Infrastructure Data Server and, by default, are retained for seven days. Records older than seven days can be archived to a local storage location. See [“Change Auditing Configuration” on page 29](#) for information about changing the archiving behavior.

All audit records contain this information:

ID

generated identifier of the audit record

Description

description of the action that is recorded (for example, authorization rule access)

Time Stamp

the date and time that the action occurred

Type

the type of action (such as security or resource)

Action

the action that was performed (such as read, create, or update)

State

the outcome of the action (success or failure)

User ID

the user, application, or service that initiated the action

Trace ID

the trace ID of the record

Properties

information unique to the type of record

Application

the application or service that performed the action

In addition, other fields might be included depending on the type of audit record.

In order to access the information in the audit records, commands are provided to list all of the audit records or to list records based on criteria such as date, application name, and user ID. The command also enables you to view details about a specific audit record. See [“List Audit Records” on page 27](#) for more information.

SAS Viya operations infrastructure also includes a predefined task to process the audit records, create a CSV file of the extracted records, and then create a CAS table with the records. Predefined reports enable you to view detailed information about access to reports, applications data, and data plans; about access by user and about access failures. See [“View Audit Record Reports and Tables” on page 26](#) for more information.

Auditing: How To

View Audit Record Reports and Tables

You can use the User Activity report, which is available from the SAS Environment Manager Dashboard, to view graphs and tables of the collected audit record data.

The genAudit task, which runs every two hours, collects information from the audit records that is then used to create these reports. The task runs using the credentials of the SAS installer ID, so the task collects only those records to which the SAS installer has access. The SAS installer ID is not a SAS administrator ID.

Note: This report is not available if you are a tenant administrator.

Follow these steps to view the reports.

- 1 On the SAS Environment Manager Dashboard, select **Show Reports**. A gallery of available reports is displayed at the bottom of the Dashboard.
- 2 Click in the **User Activity** report and select **Open**. Use the control to navigate through the report gallery to locate the **User Activity** report.
- 3 The **User Activity** report contains pages that display the audit information based on different criteria, such as user activity, report access, and data table access. Audit records are retained for seven days, so by default, the report displays information from all of the past seven days. Use the slider on each report page to view information only for a selected time range.

Select the page of the report that contains the type of information that you want to view. These pages are available:

Main

contains thumbnail graphs for the charts **Most active users**, **Activity counts**, **Most active data**, and **User activity**.

Most Active Users

displays the **Most Active Users** and **Activity Over Time** charts, and a table of the audit records ordered by level of user activity. The table does not display audit records from SAS internal users. Select a bar in the **Most Active Users** chart to display the **Activity Over Time** chart for the selected user, and to list the audit records only for the selected user.

Application Usage

displays the **Most used Applications** and **Application Activity** charts, and a table of the audit records orders by level of application activity. Select a bar in the **Most used Applications** chart to display the **Application Activity** chart for the selected application, and to list the audit records only for the selected application.

Report Activity

displays the **Top Report Usage** chart and a table of the audit records for report access. By default, the chart and table display report activity for all users. To view the report usage and audit records only for a specific user, select the user in the **Users** menu.

Data Plan Activity

displays the **Top Report Usage** chart and a table of the audit records for data plan access. By default, the chart and table display data plan activity for all users. To view the data plan usage and audit records only for a specific user, select the user in the **Users** menu.

Data Activity

displays the **Top Report Usage** chart and a table of the audit records for data table access. By default, the chart and table display data table activity for all users. To view the data table usage and audit records only for a specific user, select the user in the **Users** menu.

Failures

displays the **Failed Requests per Application** and **Failed Activities** charts, and a table of the audit records only for failed requests. By default, the **Failed Activities** chart and the audit records table display failures for all applications. To view the **Failed Activities** chart and audit records for a specific application, select the application's bar in the **Failed Requests per Application** chart.

Details

displays a table of audit records. By default, the table displays all audit records. To filter the table, use the menus at the top of the table to display only those records matching your selected criteria. You can filter by user, application, action, and state, and multiple criteria are allowed

Note: If the User Activity report is blank or displays the message `Cannot find the requested data source`, you must verify that the command-line interface (CLI) was deployed properly in your SAS Viya environment. See [“Edit the Inventory File” in SAS Viya for Linux: Deployment Guide](#) for more information.

List Audit Records

Use the command `sas-admin audit list` to list all of the audit records that have been collected. Because the list of records that are returned can be long, you can use these options to manage the records that are returned and more easily locate the records that you want to see:

```
sas-admin audit list --limit "number_of_records"
```

returns only the specified number of audit records. The default value is 50.

```
sas-admin audit list --action action_name
```

returns only audit records that contain the specified action

```
sas-admin audit list --after YYYY-MM-DDTHH:MM:SS.ssssssZhh:mm
```

returns only audit records that occur after the specified date and time

```
sas-admin audit list --application application_name
```

returns only audit records that contain the specified application name

```
sas-admin audit list --application-contains application_string
```

returns only audit records whose application name contains the string `application_string`

```
sas-admin audit list --before YYYY-MM-DDTHH:MM:SS.ssssssZhh:mm
```

returns only audit records that occur before the specified date and time

```
sas-admin audit list --description description
```

returns only audit records that contain the specified description

```
sas-admin audit list --description-contains description_string
```

returns only audit records whose description contains the string `description_string`

```
sas-admin audit list --remote-address address
```

returns only audit records that contain the specified remote address

`sas-admin audit list --remote-address-contains address_string`
 returns only audit records whose remote address contains the string *address-string*

`sas-admin audit list --state state`
 returns only audit records that contain the specified state

`sas-admin audit list --type type`
 returns only audit records that contain the specified type

`sas-admin audit list --user-id user_ID`
 returns only audit records that contain the specified user ID

`sas-admin audit list --user-id-contains user_ID_string`
 returns only audit records whose user ID contains the string *user_ID_string*

`sas-admin audit list --user-id-starts-with user_ID_string`
 returns only audit records whose user ID starts with the string *user_ID_string*

View a Detailed Audit Record

Use the `sas-admin audit show-info --id` command to display detailed information about a single audit record. The information returned look like this

```
ID                c680aeed-479c-493d-921d-5bfd1c5921f3
Description       Authorization rule access
Time Stamp       2017-10-04T11:07:51.673Z
Type             security
Action           update
State            success
User ID          sas.folders
Trace ID         0ef7f213f8085b6a
Properties        id : 4b7514e2-eb7b-40f2-97d8-a665e8bdbdae
                 objectUri : /folders/folders
                 principal : sasapp
                 type : GRANT
Application      authorization
```

View a File of Audit Records

The genAudit task is included in the default task list for the SAS Viya operations infrastructure agent. The task runs automatically every two hours and performs these functions:

- extract the audit records for reports, data plans, CAS management, and CAS access management
- write the extracted audit records to a CSV file in a cache location
- remove audit records in the CSV file from the eighth day of collection
- use the CSV file to create a table in the SystemData caslib called AUDIT

You can use the extracted audit data in the AUDIT table to perform analysis or create reports.

Reset Audit Record Extraction

If the data created by the audit record extraction process becomes corrupted or incorrect, you can reset the extraction process. This action does not alter or remove any of the original audit records. It deletes only the data in the CSV file that is extracted by the genAudit task.

This is an example of a scenario where you should reset the process. The CSV file is designed to hold seven days of audit records, so one step in the process is to remove records only from the eighth day of collection. It does not remove records that are older than the eighth day. If something prevents the genAudit task from running on a particular day, the eighth-day records are not removed, and they will remain in the CSV file from that point forward.

You should reset the extraction process if any of these go down:

- sas-ops-agent
- casManagement service
- audit service

To reset the record extraction process, delete all of the files in the directory `/opt/sas/viya/config/var/cache/auditcli`. The genAudit task will create new extracted audit data when the task runs again after two hours.

Change Auditing Configuration

These configuration properties enable you to change aspects of the audit process:

`sas.audit.archive`

enables you to configure how audit records are archived. Archiving removes the records older than a specified time and stores them in a local file system. This configuration instance enables to you specify whether to archive audit records, how long to keep records before archiving, and location where archived records are stored.

`sas.audit.record`

enables you to configure the actions that generate an audit record. To add an action type, you must specify the property name (such as `resource.action.read.state` and the value (such as `failure`).

See [“Configuration Properties: How To Configure Services” on page 215](#) for more information.

Authentication

Authentication: Overview

Authentication is the process of verifying the identity of a user that is attempting to log on to or access software.

In SAS Viya, authentication options vary, based on which interface is being used in your environment:

- In a full deployment, the pluggable authentication module (PAM) validates the user's credentials when accessing SAS Studio and CAS Server Monitor. Users can be authenticated through SAS Logon Manager, using an LDAP provider, Kerberos, Security Assertion Markup Language (SAML), or OAuth and OpenID Connect.
- In a programming-only deployment, the supported authentication mechanism is PAM.

Authentication: How To

Authentication Mechanisms

Overview

Authentication mechanisms integrate SAS into your computing environment. External mechanisms include direct LDAP authentication (which is referred to as LDAP in this documentation), host authentication, Kerberos, Security Assertion Markup Language (SAML), and OAuth and OpenID Connect. Pluggable authentication modules (PAM) extend UNIX host authentication.

The following sections are listed alphabetically. Configure the authentication mechanism that is appropriate for your environment. For more information, see [“Authentication Mechanisms” on page 45](#).

Configure Kerberos

Verify Kerberos Prerequisites

Before configuring Kerberos, make sure that the following components exist:

Note: These prerequisite components are usually configured by the Active Directory administrator.

- 1 A service account exists in Active Directory.
- 2 A service principal name (SPN) is mapped to the service account:
 - a Verify that there is a mapping already configured:

```
setspn -F -Q HTTP/hostname.example.com
```

Output A.1 Sample SPN Query

```
CN=user-logon-name,OU=Service Accounts,OU=Domain Controllers,OU=Servers,DC=EXAMPLE,DC=com
  HTTP/<hostname.example.com
  HTTP/HOSTNAME

Existing SPN found!
```

If an SPN is not found, then contact your information technology support group for assistance with registering the machine.

- b** Verify that the service is linked to the service account:

```
setspn -L user-logon-name
```

Output A.2 Sample Account Query

```
Registered ServicePrincipalNames for CN=user-logon-name,OU=Service Accounts,OU=Servers,DC=EXAMPLE,DC=com:
  HTTP/<hostname>@<example>.com
  HTTP/<hostname>
```

The value for `user-logon-name` is the same one identified in the common name (CN) from the previous command output, or as the `sAMAccountName` on the service account in Active Directory.

- 3** A keytab file has been generated by issuing the following command:

```
ktutil
rkt path-to/hostname.keytab
list -e
```

```
slot KVNO Principal
-----
  1   3   HTTP/<hostname>@<example>.com (arcfour-hmac)
```

For more information about the `ktutil` command, see the vendor documentation.

How to Configure Kerberos

Note: This information does not apply to a programming-only deployment.


- 1** If you have not already done so, add your user ID or an Active Directory group that contains the environment administrators, as a member of the SAS Administrators group. Then, log off from SAS Environment Manager. For more information, see [“Add or Remove Custom Group Members” on page 477](#).

CAUTION! You must specify your personal user ID. Your user ID must be in your specified LDAP provider. It must match the user ID that you use to log on to your system. Also, your user ID must be added to the SAS Administrators group because once Kerberos is configured, you can no longer sign in as the `sasboot` user.

- 2** Change the permissions, owner, and group on the file by running the following command:


```
chmod 600 keytab-filename
chown sas keytab-filename
chgrp sas keytab-filename
```

- 3** Verify that the SPN is mapped to the principal name.
- 4** Configure the Kerberos authentication properties.

- a Log on to SAS Environment Manager, using your user ID or the ID of a user who is a member of the SAS Administrators group.
- b Navigate to the SAS Logon Manager configuration definitions. For more information, see [“Edit Authentication Configuration Instances” on page 40](#).
- c In the **Definitions** list, select **sas.logon.kerberos**.
- d In the top right corner of the window, click .
- e In the New sas.logon.kerberos Configuration dialog box, enter the values for the following fields, based on your environment:

Field	Value
debug	On
holdOnToGSSContext:	On Note: This option enables Kerberos delegation to SAS Cloud Analytic Services. Enable this option if user delegation is being used.
keyTabLocation	file:///location-of-keytab
servicePrincipal	principal-name-from-keytab
stripRealmForGss	On Note: When enabled, this option strips the realm from the user name.

Note: Contact your administrator for the keytab location and the host name of the service principal.

- f Click **Save**.
- 5 Add Kerberos to the active profile.
 - a In the navigation pane, switch to the **All services** list and select **SAS Logon Manager**.
 - b In the **spring** instance, click .
 - c In the Edit spring Configuration dialog box, add **kerberos** to the **profiles.active** field.
The following value should be specified for the **profiles.active** field:

```
ldap, postgresql, kerberos
```

- d Click **Save**.

- 6 Restart the SAS Logon Manager service by running the following command:

```
sudo service sas-viya-saslogon-default restart
```

Note: It might take several minutes to restart SAS Logon Manager.

Configure Kerberos for SAS Cloud Analytic Services

- 1 Create a keytab file for CAS to use.

The file is used to validate incoming user Kerberos tickets and generate server identity Kerberos tickets for access to kerberized resources, such as Hadoop. By default, the keytab file should reside in the `/etc/sascas.keytab` file. If you save the file in a different directory or use a different filename, set the `KRB5_KTNAME` environment variable (for example, `env.KRB5_KTNAME = 'fully-qualified-filename'`). For more information, see “[CAS Environment Variables](#)” on page 648.

- 2 If you changed the default principal name, set the `CAS_SERVER_PRINCIPAL` environment variable (for example, `env.KRB5_KTNAME = 'fully-qualified-filename'`).

By default, CAS uses the following Kerberos principal name: `sascas/fully-qualified-DNSname`. CAS searches for this principal in the keytab file.

- 3 Specify the 'kerb' option for the `cas.PROVLIST` configuration file option (for example, `cas.PROVLIST='kerb'`).

For more information about the configuration file option, see “[Configuration File Options](#)” on page 627.

Configure the Microsoft Internet Explorer to Use SPNEGO

Configure Security Settings

- 1 Select **Tools** ⇒ **Internet options** ⇒ **Security**.
- 2 Select **Local intranet** and then click **Sites**.
- 3 Configure the intranet domain settings:
 - a Verify that the check boxes for the following items are selected:
 - **Include all local (Intranet) sites not listed in other zones**
 - **Include all sites that bypass the proxy server**
 - b Click **Advanced** and add your domain name to the **Websites** list to ensure that Internet Explorer recognizes any site with your domain name as the intranet.
- 4 Configure intranet authentication:
 - a In the **Security level for this zone** area, click **Custom level**.
 - b Scroll to the **User Authentication** section, select **Automatic Logon only in Intranet Zone**, and click **OK**.

Configure Connection Settings

If your site uses a proxy server, follow these steps:

- 1 Select **Tools** ⇒ **Internet options** ⇒ **Connections**.
- 2 Click **LAN settings**.
- 3 Verify that the proxy server address and port number are correct.
- 4 Click **Advanced**.
- 5 Verify that the correct domain names are entered in the **Exceptions** field on the Proxy Settings dialog box.

Configure Integrated Windows Authentication

- 1 Select **Tools** ⇒ **Internet options** ⇒ **Advanced**.
- 2 Scroll to the **Security** section, and verify that **Enable Integrated Windows Authentication** is selected.
- 3 Click **OK** and restart the browser to activate the changes.

Configure the Mozilla Firefox to Use SPNEGO

- 1 From a browser window, navigate to `about:config`.
- 2 Click **I accept the risk!** to accept the security warning.
- 3 In the **Search** field, enter `network.negotiate`.
- 4 Double-click the **network.negotiate-auth.trusted-uris** Preference Name, enter `http://hostname.example.com`, in the **Enter string value** field, and then click **OK**.

Note: The values in the **Enter string value** field are comma-separated.

Configure the Google Chrome to Use SPNEGO

Configure Security Settings

- 1 Click the **Chrome menu** key on the browser toolbar, and then select **Settings**.
- 2 Select **Show advanced settings**.
- 3 Scroll to the **Network** section, and click **Change proxy settings**.
- 4 In the Internet Properties dialog box, select **Security**.
- 5 Select **Local intranet**, and then click **Sites**.
- 6 Configure the intranet domain settings:
 - a Verify that the check boxes for the following items are selected:
 - **Include all local (Intranet) sites not listed in other zones**
 - **Include all sites that bypass the proxy server**
 - b Click **Advanced** and add your domain name to the **Websites** list to ensure that Internet Explorer recognizes any site with your domain name as the intranet.
- 7 Configure intranet authentication:
 - a In the **Security level for this zone** area, click **Custom level**.
 - b Scroll to the **User Authentication** section, select **Automatic Logon only in Intranet Zone**, and click **OK**.

Configure Connection Settings

If your site uses a proxy server, follow these steps:

- 1 In the Internet Properties dialog box, select **Connections**.
- 2 Click **LAN settings**.
- 3 Verify that the proxy server address and port number are correct.



- 4 Click **Advanced**.
- 5 Verify that the correct domain names are entered in the **Exceptions** field on the Proxy Settings dialog box.

Configure Advanced Settings

- 1 In the Internet Properties dialog box, select **Advanced**.
- 2 Scroll to the **Security** section, and verify that **Enable Integrated Windows Authentication** is selected.
- 3 Click **OK** and restart the browser to activate the changes.

Configure User Delegation

Set the sas.logon.kerberos configuration property

- 1 Using SAS Environment Manager, in the applications menu () , select **Administration** ⇒ **Manage Environment**. In the navigation bar, select .
- 2 Update the configuration instance for **sas.logon.kerberos**, ensuring that you enable the **holdOnToGSSContext** option. For more information, see [“Create Configuration Instances” on page 216](#).

Configure User Delegation for Microsoft Internet Explorer

- 1 Verify that Integrated Windows Authentication is enabled. For more information, see [Configure Advanced Settings on page 35](#).
- 2 Select **Tools** ⇒ **Internet options** ⇒ **Security**.
- 3 Select **Trusted Sites**, and then click **Sites**.
- 4 Enter the middle-tier host name in the **Add this website to the zone** field and click **Add**.
- 5 Click **Close**, and then click **OK**.

Note:

For Internet Explorer to pass a forwardable ticket to the SAS Viya machine, the service account in Active Directory holding the SPNs must be trusted for delegation.

Configure User Delegation for Mozilla Firefox

- 1 From a browser window, navigate to `about:config`.
- 2 Click **I accept the risk!** to accept the security warning.
- 3 In the **Search** field, enter `network.negotiate`.
- 4 Double-click the **network.negotiate-auth.delegation-uris** Preference Name, enter `http://hostname.example.com` in the **Enter string value** field, and then click **OK**.

Configure User Delegation for Google Chrome

By default, Chrome disables the delegation of Kerberos credentials. The Windows registry must be updated. Microsoft recommends performing a system backup before editing the registry. Complete the following steps to enable Kerberos delegation:

- 1 Open the Windows registry editor.
- 2 Add the following REG_SZ keys:

Software\Policies\Google\Chrome\AuthServerWhitelist

Specifies which servers should be whitelisted for integrated authentication. Set the value to the SAS Web Server host name: *hostname.example.com*.

Software\Policies\Google\Chrome\AuthNegotiateDelegateWhitelist

Specifies which servers Chromium can delegate to. Set the value to the SAS Web Server host name: *hostname.example.com*.

Note: You might also need to add Google and Chrome under Policies.

Configure SAML

Overview

Note: This information does not apply to a programming-only deployment.


To configure the Security Assertion Markup Language (SAML), complete the following:

- 1 [“Configure the SAML Service Provider” on page 37](#)
- 2 [“Configure the SAML Identity Provider – SAS Environment Manager Configuration” on page 38](#)
- 3 [“Configure the SAML Identity Provider – Relying Party Configuration” on page 39](#)

Configure the SAML Service Provider

- 1 Log on to SAS Environment Manager.
- 2 Navigate to the SAS Logon Manager configuration definitions. For more information, see [“Edit Authentication Configuration Instances” on page 40](#).
- 3 In the **Definitions** list, select **sas.logon.saml**.

Note: If you change any of the `sas.logon.saml` properties, the new metadata must be provided to the Relying Party in ADFS. If it is not, the SAML connections might fail.

- 4 In the top right corner of the window, click .
- 5 In the New `sas.logon.saml` Configuration dialog box, enter values for the required fields, based on your environment. The following table provides guidance on what information needs to be provided for the listed fields:


Field	Description
entityBaseURL	The external URL that is called by the browser (for example, <code>https://hostname.example.com/SASLogon</code>).
entityID	The unique ID that represents the service provider that is included in protocol messages between relying parties. Change from the default value that is pre-populated.
serviceProviderCertificate	Paste a copy of the PEM-encoded (base64) certificate, which is used by the service provider.
serviceProviderKey	Paste a copy of the PEM-encoded (base64) key, which is used by the service provider.

Field	Description
serviceProviderKeyPassword	Provide the password for the service provider, or leave blank if there is no password.
setProxyParams	Note: This field should not be modified. The value should remain false . Specifies whether to allow the base URL to reside behind an HTTP proxy.
signMetaData	Specifies whether the local service provider should sign the metadata.
signRequest	Specifies whether the local service provider should sign the SAML requests.
wantAssertionSigned	Specifies whether the assertions should be signed.

6 Click **Save**.

Configure the SAML Identity Provider – SAS Environment Manager Configuration

1 Complete the following steps in SAS Environment Manager:

- a** In the **Definitions** list, select **sas.logon.saml.providers.external_saml**.
- b** In the top right corner of the window, click .
- c** In the New **sas.logon.saml.providers.external_saml** Configuration dialog box, enter values for the required fields, based on your environment. The following table provides guidance on what information needs to be provided for the listed fields:

Field	Description
idpMetadata	The URL to the location of the identity provider metadata (for example, <code>https://hostname.example.com/filename.xml</code>). This information is provided by your information technology support group. Use the URL to configure Active Directory Federation Services (ADFS) or another service, along with endpoints. Note: In this document, ADFS is used for configuration.
metadataTrustCheck	Specify whether to trust the identity provider certificate.
nameID	The field is populated with a default value. Verify with your information technology support group that the value is correct.
showSamlLoginLink	Determines whether a link should be displayed on the login page for this identity provider.

d Click **Save**.

- 2** Edit the `SAS-Viya-configuration-directory/etc/sysconfig/sas-javaesnt1/sas-java-services` file, and uncomment the highlighted line in the following block:

```

if [ -f $truststore ]; then
    export java_global_option_truststore="-Djavax.net.ssl.trustStore=$truststore"
    export java_global_option_truststore_password="-Djavax.net.ssl.trustStorePassword=changeit"
fi

```

- 3 As an administrator with sudo privileges, restart the SAS Logon Manager service by running the following command:

```
sudo service sas-viya-saslogon-default restart
```

Note: It might take several minutes to restart SAS Logon Manager.

Configure the SAML Identity Provider – Relying Party Configuration

You can either configure the relying party trust or supply the required information to your information technology support group, in order for them to add the relying party trust. Here is an overview of the steps to perform, if you configure the relying party trust. The steps might vary, depending on which tool you use for configuration.

- 1 If the identity provider requires it, configure Transport Layer Security (TLS), if it has not already been configured. For more information, see [“Secure Apache HTTP Server” on page 363](#).
- 2 Download the application metadata.xml file, which contains information about the service provider, or provide the `https://hostname/SASlogon/saml/metadata` link to your information technology support group.
- 3 Request that your information technology support group configure a relying party in the identity provider.

Configure PAM

Default pluggable authentication module (PAM) configuration files are installed for both the CAS server and SAS Studio.

- 1 As a user with root authority, edit the `/etc/pam.d/service` file. For the CAS server, `service` is `cas`. For SAS Studio, `service` is `sasauth`.

The following information is displayed for the CAS server:

```

$ vi /etc/pam.d/cas
#%PAM-1.0
auth    include      password-auth
account include      password-auth
password include      password-auth
session include      password-auth

```

The following information is displayed for SAS Studio:

```

$ vi /etc/pam.d/sasauth
#%PAM-1.0
auth    include      password-auth
account include      password-auth

```

- 2 Make any modifications to the file that are necessary for your environment.
- 3 Save the file and exit.

Session Management

Overview

Note: This information does not apply to a programming-only deployment.

The following sections provide information about customizing SAS Logon Manager and the user's session experience.

Edit Authentication Configuration Instances


- 1 From the side menu , under **SAS Environment Manager**, click **Resources** ⇒ **Configuration**.
- 2 In the top left corner of the window, select **Definitions** from the drop-down box.

See Also

- [SAS Viya Administration: Configuration Properties](#)

Customize Sign-in, Sign-out, and Session Time-out Content


You can configure customized content that is displayed when users of SAS web applications sign in, sign out, or the session reaches the time-out interval. To enable the display of customize content, follow these steps:

- 1 In the **Definitions** list, select **sas.logon.custom**.
- 2 In the top right corner of the window, click .
- 3 In the New sas.logon.custom Configuration dialog box, specify the URI that contains the custom content that you want to display. Here are the available fields:
 - login
 - logout
 - timedout

For a description of the properties, see "[sas.logon.custom](#)" on page 245.


- 4 Click **Save**.

Customize Concurrent Sign-in Sessions

- 1 In the **Definitions** list, select **sas.logon.sessions**.
- 2 In the top right corner of the window, click .
- 3 In the New sas.logon.sessions Configuration dialog box, you can set the following properties:
 - maxConcurrentSessions**
Set this property to limit a user to a certain number of concurrent sessions.
 - rejectNewSessionsIfMaxExceeded**
When sessions are limited, the default behavior is to cause an existing session to expire and grant a new session to the user attempting to authenticate. To override this behavior and prevent a new session from being granted, set this property to *true*.

- 4 Click **Save**.

Configure the HTTP Session Time-out Interval

- 1 In the **Definitions** list, select **server**.
- 2 In the top right corner of the window, click .

- 3 In the New server Configuration dialog box, complete the following:
 - a Select **SAS Logon Manager** from the **Services** drop-down list.
 - b Click **+**.
 - c In the **Name** field, specify `session.timeout`.
 - d In the **Value** field, specify the amount of time a session has to be idle before it times out, in seconds.
 - e Click **Save**.
- 4 Click **Save**.
- 5 Restart all services to reflect the new time-out interval. For more information, see [“All Servers and Services” on page 599](#).

Disable Logins

As a SAS administrator, you can disable logins through operating system firewall rules or using LDAP. This disables new sessions, ends current sessions, and prevents others from using the deployment. For more information, see the appropriate documentation for your operating system.

Additional Authentication Topics

Obtain an Access Token Using Password Credentials

You can use the following commands to register a client ID and secret and obtain a token that can be used to call a SAS Viya API and to access SAS Viya credentials from SAS 9.4.

- 1 Register a new client ID and secret by completing the following steps:

Note: You must register a client ID once.

- a Obtain a token to register a new client ID and secret. For more information, see [“Obtain an Access Token to Register a New Client ID” on page 403](#).
- b Use the token to register the new client ID and secret by running the following curl command:

Note: The initial line of the curl command must be entered on one line. It is shown on more than one line for display purposes only.

```
curl -X POST http://localhost/SASLogon/oauth/clients -H "Content-Type: application/json"
-H "Authorization: Bearer token-from-previous-step"
-d '{
  "client_id": "client-id",
  "client_secret": "client-secret",
  "scope": ["openid", "*"],
  "resource_ids": "none",
  "authorities": ["uaa.none"],
  "authorized_grant_types": ["password"]
}'
```

Note: The value for the **scope** parameter can be a list of scopes and groups that you **might** request when obtaining a token. You might also specify the wildcard "*" to request all scopes always. Ensure that you specify the list correctly. SAS Viya treats group memberships as scopes. Therefore, the list of scopes is the list of group memberships that you might request when obtaining a token. The "openid" is a special scope that represents authentication only and should always be included.

- 2 A token can be used until it expires. By default, this is 12 hours. To acquire a token, run the following curl command:

```
curl http://localhost/SASLogon/oauth/token
-H "Accept: application/json"
-H "Content-Type: application/x-www-form-urlencoded"
-d "grant_type=password&username=username&password=password"
-u "client-id:client-secret"
```

Note: The values for *client-id* and *client-secret* should be the same as the values that were specified in [Step 1b](#).

- 3 Retrieve the access token information from the results of the curl command in [Step 2](#). This access token is used to perform the following tasks:
 - a Call a SAS Viya API by passing the HTTP Authorization header as a Bearer token: `Authorization: Bearer access_token`.
 - b Assign the access token to the `SAS_VIYA_TOKEN` environment variable. Setting this environment variable enables you to access SAS Viya credentials from SAS 9.4. For example, when the `AUTHDOMAIN=` option is set on a `CAS` or `LIBNAME` statement, an attempt is first made to retrieve credentials from the SAS Viya Credentials service before searching the metadata.

Create an Authinfo File

The Authinfo file supplies a user name and password that is sent to CAS for authentication. For information about how to create an Authinfo file, see [Create an Authinfo File](#).

Authentication: Concepts

Authentication Architecture

In a full deployment, authentication services are provided by SAS Logon Manager. SAS Logon Manager is based on the Cloud Foundry User Account and Authentication (UAA) server. The security architecture is built around Open Authorization (OAuth) and OpenID Connect. By default, authentication is performed via a Lightweight Directory Access Protocol (LDAP) provider. Authentication support is also available for Kerberos, OAuth, and Security Assertion Markup Language (SAML).

In a programming-only and full deployment, host authentication is supported. You can configure the host to use only pluggable authentication modules (PAM).

Authentication Options

Authentication for Visual Interfaces

With visual interfaces, users are authenticated through SAS Logon Manager. SAS Logon Manager is a web application that handles all authentication requests for SAS web applications and is accessed via the Apache HTTP Server.

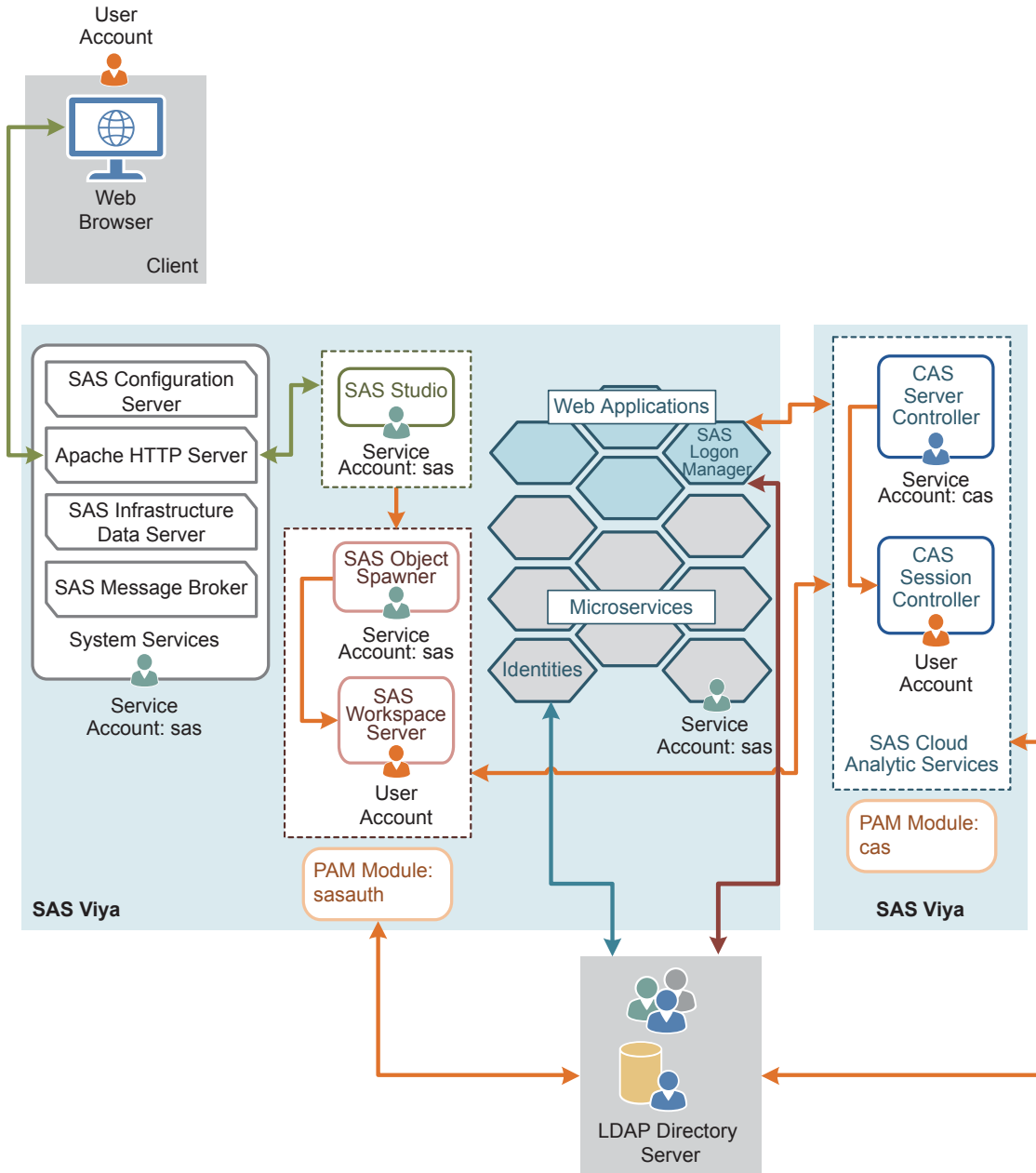
The following figure shows how a user authenticates to SAS Logon Manager and the supported authentication mechanisms.

- The fifth option is pluggable authentication module (PAM). In this configuration, SAS Logon Manager uses the operating system PAM stack. The identity service verifies users in LDAP.

With all five options, the connection to the SAS Cloud Analytic Services (CAS) environment is performed using internal OAuth tokens that are generated by SAS Logon Manager. In most cases, the session that is started by the CAS controller runs on the operating system as the same user who launched the CAS operating system service. This defaults to the cas account.

Authentication for Programming Interfaces

The following figure shows how a user is authenticated while using programming interfaces.



In a deployment with programming interfaces, the user's credentials are entered into SAS Studio via the Apache HTTP Server. Then SAS Object Spawner uses pluggable authentication module (PAM) configuration files on the host to validate the user ID and password. The user ID and password can be a local account on the host or, depending on the PAM configuration, an account in the LDAP provider. Once the user is authenticated, the SAS

Workspace Server is started. The PAM configuration file for SAS Studio is `sasauth` and includes the password module.

SAS Workspace Server connects to the CAS environment using the user ID and password that were used to start the SAS Workspace Server. The CAS controller uses its own PAM configuration to validate the user's credentials and launch the session process as the user. The PAM configuration file for CAS is `cas` and includes the password module.

The CAS controller uses the user ID and password to obtain an internal OAuth token from SAS Logon Manager. This requires the user ID and password to be valid in the LDAP provider that is configured for SAS Logon Manager. Otherwise, CAS cannot obtain an OAuth token, and the session will fail. Therefore, PAM for SAS Studio (`sasauth`), PAM for CAS (`cas`), and SAS Logon Manager should all use the same or equivalent LDAP providers. If these three components are not sending the user ID and password that was entered into SAS Studio to the same provider, errors might be generated when trying to connect.

Authentication Mechanisms

LDAP Authentication

Overview of LDAP

Note: This information does not apply to a programming-only deployment.

In SAS Viya, LDAP is used for identifying and authenticating users. Third-party LDAP server implementations are supported, including Microsoft Active Directory and OpenLDAP.

How It Works in SAS Viya

LDAP is the default authentication mechanism. The Identities service always makes a direct connection to LDAP to obtain user and group information. By default, SAS Logon Manager authenticates users using a direct connection to the configured LDAP provider. To ensure that network connections are secure, the connection between the browser and the Apache HTTP Server can be secured with HTTPS. In addition, the connection between SAS Logon Manager and the LDAP provider can be secured with LDAPS.

For information about configuring LDAP, see [Configure the Connection to Your Identity Provider](#).

Kerberos Authentication

Overview of Kerberos

Note: This information does not apply to a programming-only deployment.

Kerberos is a network authentication protocol that is used to verify user or host identity. The Kerberos protocol uses strong cryptography so that a client can prove its identity to a service (and vice versa) across an insecure network connection. During Kerberos authentication, a user's credentials (user ID and password) are not sent over the network. Instead, both the client and the service use the credentials that they have been supplied as a key in an encryption algorithm to encrypt the message that is sent between the client and the service. If the client sends an encrypted message, and the service uses the same key to decrypt the message, it is proven that the credential is known without having to transmit the credentials.

Key Terms

Client	An application that is attempting to connect to and access a resource, on behalf of a user. Resources include reports that are viewed, services that are accessed, and databases that are queried. In SAS Viya, the client is the web browser.
--------	--

Service	A service, or server, that hosts a resource the user wants to connect to. The service must be able to validate the service tickets presented by the client.
Key Distribution Center	A trusted third party within Kerberos that verifies the authenticity of the client and service. Both the client and service must trust the KDC. In addition, end users and services must register with the KDC.
Service Principal Name	A unique name that is used to identify a web service that is running on a server. Before a service principal name (SPN) can be used, it must be registered. Every web service that uses Kerberos authentication needs to have an SPN set for it so that clients can identify the server on the network. An SPN usually matches the pattern of <code>HTTP/hostname.example.com</code> .
Keytab File	A file containing pairs of Kerberos principals and encrypted keys. The keys are associated with a password for the principal. The principals are SPNs. Keys can use different encryption algorithms. For a single principal, you might have several entries that correspond to each encryption type.
Ticket-granting ticket	An encrypted identification file that is valid for a limited amount of time. After a user is authenticated, this file is granted to a user for data traffic protection by the KDC. The TGT file contains the session key, its expiration date, and the user's IP address.

How It Works in SAS Viya

In addition to using the LDAP provider to obtain user and group information, you can configure SAS Logon Manager for Kerberos authentication. This option replaces the option to use the default LDAP provider for authentication to SAS Logon Manager. Kerberos provides the user with single sign-on capabilities from the browser on their desktop. Single sign-on allows the user to access the SAS Viya visual interfaces without being prompted to enter their credentials.

Integrated Windows Authentication

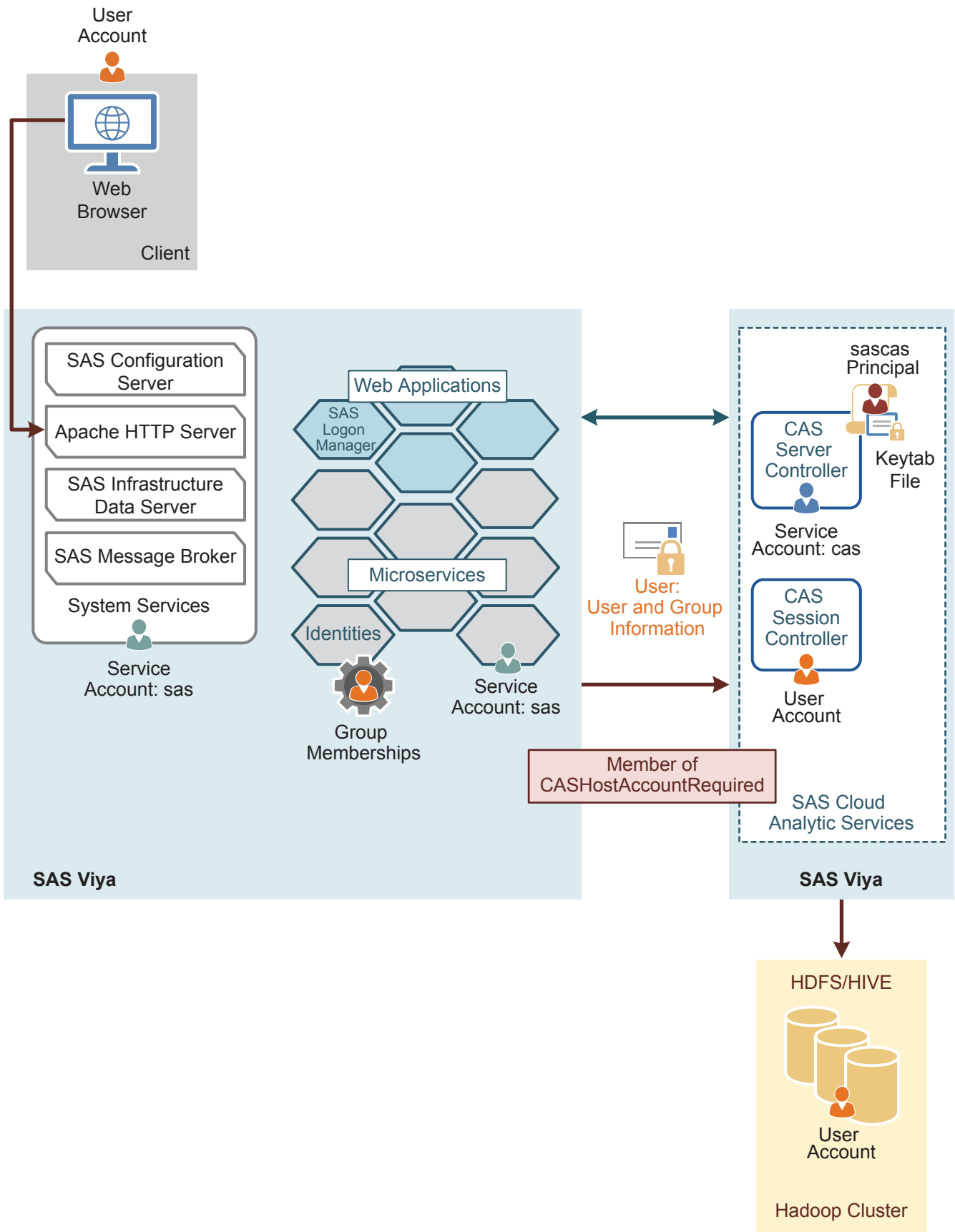
Integrated Windows Authentication (IWA) uses Kerberos authentication and is a Microsoft technology that is used in an environment where users have Windows domain accounts. With IWA, the credentials are hashed before being sent across the network. The client browser proves its knowledge of the password through a cryptographic exchange with the web application server. When IWA is used in conjunction with Kerberos, IWA enables the delegation of security credentials. The Kerberos protocol uses strong cryptography so that a client can prove its identity to a server (and vice versa) across an insecure network connection.

For information about how to configure IWA, see [Configure Advanced Settings on page 35](#).

Kerberos Delegation to SAS Cloud Analytic Services

Kerberos delegation to CAS, or user delegation, is a feature that allows a SAS Viya application to reuse the end-user credentials to access Kerberized systems. Delegation allows a server to forward a user's credentials to the CAS server where they can be used to access other Kerberized services, such as Hadoop. By default, user delegation is not enabled and must be configured.

There are four possible scenarios for accessing a Hadoop environment that is secured by Kerberos. The following figure illustrates the first scenario:



In this scenario, the `CASHostAccountRequired` custom group notifies CAS that the user session needs to be launched under the user's operating system account. Kerberos is used for authentication, and the user is a

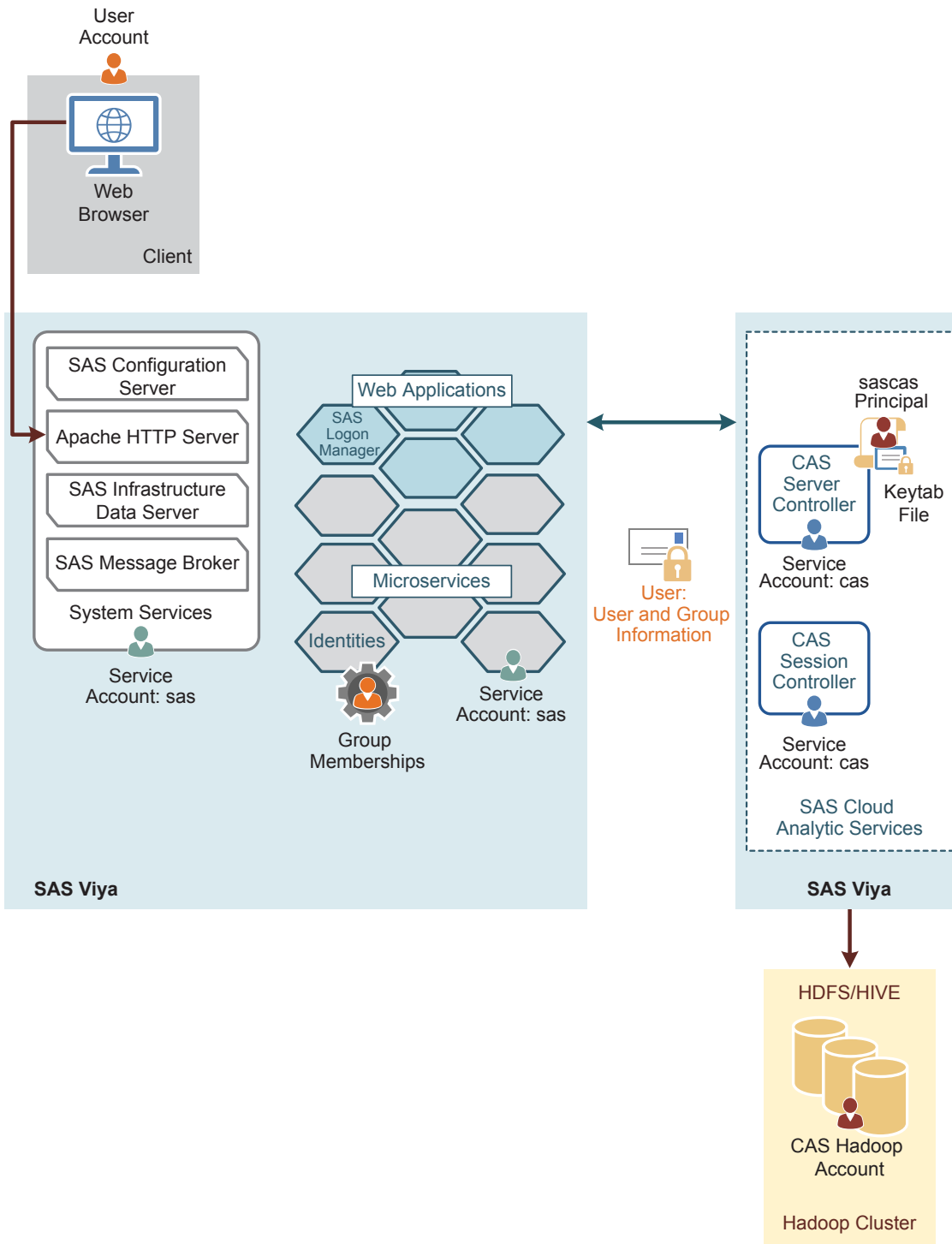
member of the CASHostAccountRequired group. Therefore, the user's credentials are delegated to CAS, and the user can access Hadoop using Kerberos as himself or herself. For more information about the CASHostAccountRequired custom group, see ["Initial Rules for Other Predefined Custom Groups" on page 495](#). The service account that is associated with the Apache HTTP Server has to be trusted for delegation. An OAuth token is generated by SAS Logon Manager. It indicates to CAS that Kerberos was used to authenticate the user. If the user is a member of the CASHostAccountRequired group, CAS attempts to obtain a Kerberos ticket from SAS Logon Manager after the OAuth token is validated. The session is launched under the user's operating system identity, and the user's Kerberos ticket is used for secured Hadoop access.

See Also

["Configure Kerberos for SAS Cloud Analytic Services" on page 34](#)

Kerberos in Visual Interfaces without Delegation

The following figure illustrates the second scenario:



In this scenario, there is a full deployment and a user that relies on Kerberos to log on to SAS Logon Manager as part of his or her access to the visual interfaces. The user accesses the visual interfaces and connects to CAS and a secured Hadoop environment. The connection between the visual interfaces and CAS uses OAuth to authenticate the user. Users obtain an OAuth token from the SAS Logon Manager as part of their initial authentication, and this token is used to authenticate to CAS. The OAuth token provides user and group information that enables CAS to provide integration with the authorization information that is stored in the SAS services.

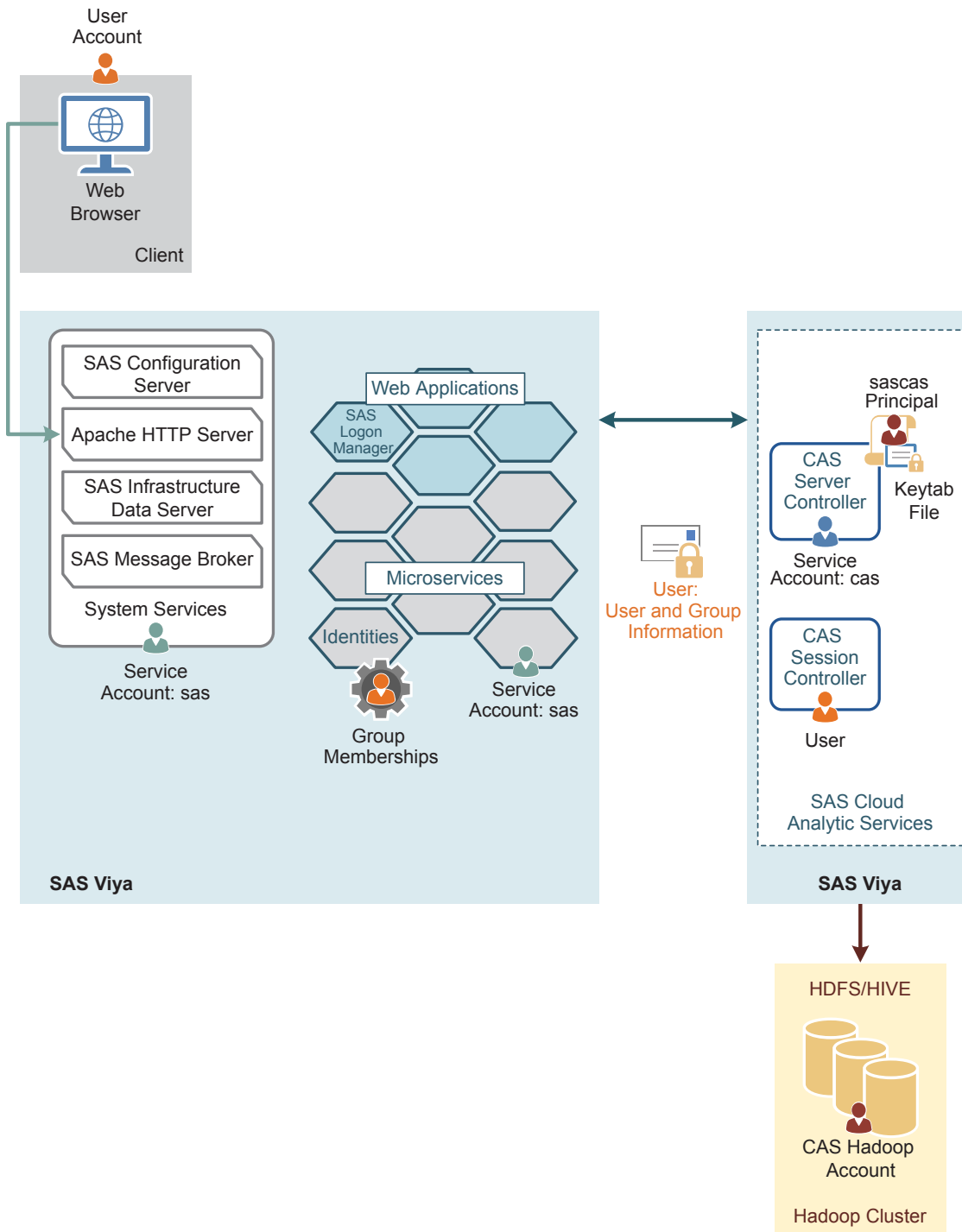
The user is not a member of CASHostAccountRequired. Therefore, CAS does not try to obtain a delegated Kerberos ticket and does not attempt to launch the user's session under the user's operating system identity.

The CAS Session Controller and CAS Session Workers all run as the service account that launched the CAS operating system service. By default, this is the cas account.

If the user needs to access a secured Hadoop cluster, where Kerberos Service Tickets are required to access the data, then the CAS Server Controller Kerberos credentials must be provided. The credentials are provided in the form of a Kerberos keytab file. This keytab file enables the CAS Server Controller to initialize a set of Kerberos credentials at start-up. This set of Kerberos credentials does not have to be for the cas user. The credentials in the keytab can be for any user, but all access from CAS to Hadoop uses this single, shared credential. In the diagram, the shared credentials are listed as “CAS Hadoop Account”.

Kerberos in Visual Interfaces with Outbound Authentication from SAS Cloud Analytic Services

In the third scenario, there is a full deployment in which users attempt to access Hadoop using a CAS Hadoop account.



In this scenario, the user of the visual interfaces has an experience closer to the programming users when launching the CAS session. An administrator can create the `CASHostAccountRequired` custom group and add members to the group. For users in this group, their CAS session and worker processes run under their operating system identity instead of the CAS service account. For more information about the `CASHostAccountRequired` custom group, see [“Initial Rules for Other Predefined Custom Groups” on page 495](#).

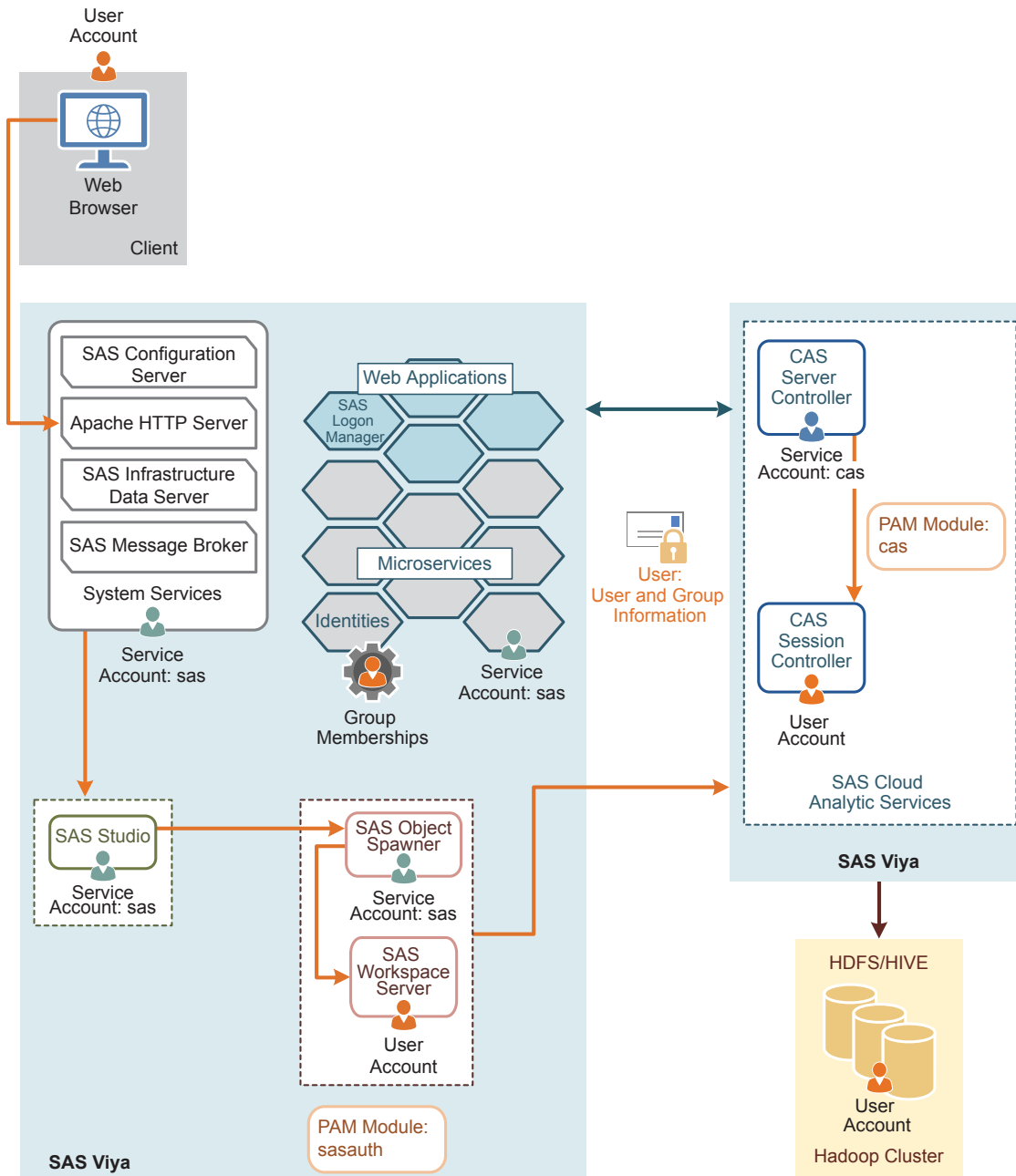
The CAS Server Controller directly launches the CAS Session Controller using the user ID from the OAuth token rather than passing any credentials to the PAM stack on the host. Therefore, the user ID from the OAuth token must be known to the operating system where the CAS Session Controller and CAS Session Workers are running. The process runs as the user on the host, even though the user was not authenticated on the host.

Since the PAM stack was not used to authenticate the user and launch his or her CAS session, the user's Kerberos credentials were never initialized.

In addition, since the CAS session is now running as the user, it is unable to access the Kerberos credentials cache initialized by the CAS Server Controller on start-up from the Kerberos keytab. Therefore, the user cannot access the secured Hadoop cluster.

Kerberos in Programming Interfaces

The following figure illustrates the fourth scenario:



In this scenario, there is a full deployment and a user that provides his or her user ID and password to CAS. CAS uses its own pluggable authentication modules (PAM) configuration to validate the user's credentials and launch the CAS Session Controller process running as the user. In addition, the CAS Server Controller also uses the user ID and password to obtain an OAuth token from SAS Logon Manager. The OAuth token provides the

user's group memberships from the Identities service. These memberships are essential in enforcing access control.

The PAM stack is configured to generate a Kerberos credentials cache during authentication. The resulting cache can be used to access Hadoop as the user.

Depending on the deployment options that you chose, users who access both the programming interface and the visual interface might have different access to Hadoop.

OAuth Authentication

Overview of OAuth

Note: This information does not apply to a programming-only deployment.

Open Authorization (OAuth) is a token-based authentication standard on the internet. OAuth acts as an intermediary on behalf of the user, giving the third-party service an access token that authorizes specific account information.

Key Terms

Access token	Specifies identifying information for a user, including the user's credentials, groups, and privileges.
OpenID Connect	An authentication layer built on top of OAuth 2.0.
Flow	The process for obtaining an OAuth token.

How It Works in SAS Viya

An OAuth and OpenID Connect provider can be internal to the customer's environment, or it can be an external provider, such as Google Authenticator or Facebook. When the OAuth option is configured, this does not completely replace the default LDAP provider. Instead, when users access the SAS Logon Manager, they are presented with a link to authenticate using OAuth and the standard logon form using the LDAP provider. Users can select which to use. The user identity and group membership information is looked up in LDAP. OAuth can provide single sign-on from the OAuth provider. For example, when a user signs in to his or her Google account, the user can access the visual interfaces of SAS Viya without being prompted any further for credentials.

SAML Authentication

Overview of SAML

Note: This information does not apply to a programming-only deployment.

The Security Assertion Markup Language (SAML) standard defines a framework for exchanging security information about users between an identity provider and service provider. This security information is packaged in the form of portable XML assertions that applications working across security domain boundaries can trust. SAML allows for single sign-on to web browser applications.

Key Terms

Federation	Allows multiple identity management systems to work together and establish trust.
------------	---

Assertion	A package of information, in the form of an XML document, that is created and sent during a federated access request.
Claims	Information that a federation member is asserting to be true.
Identity provider	A federation member that authenticates users and keeps track of their information. Creates assertions for the users, and sends them to service providers.
Service provider	A federation member that consumes assertions to make access control decisions for its applications.
Metadata	An XML document that is produced by a SAML provider to describe its service endpoint URLs, x.509 certificate, and other information in a standard way for consumption by partners in the federation.
Relying party	A server providing access to secure software.

How It Works in SAS Viya

SAML supports configuring the SAS Logon Manager to be integrated with an external SAML identity provider. This identity provider can be internal or external to the customer's environment. If it is internal, a tool similar to Oracle Access Manager can be used. If it is external, something like salesforce.com can be used. SAML does not completely replace the default LDAP provider. End-users accessing the SAS Logon Manager can choose SAML authentication or the default LDAP provider. The user identity and group membership information is looked up in LDAP. This option also provides single sign-on with the third-party SAML provider.

When a user attempts to access a service URL, the service provider, which is SAS Logon Manager, initiates the exchange with an authentication request. The identity provider sends a response that contains the assertion. The SAML protocol defines the structure and content of these request and response messages. When the user logs on to a service or system, the service provider trusts the identity provider to validate the credentials, instead of providing credentials to the service provider. Therefore, users do not have to provide their credentials directly to anyone but the identity provider.

For configuration information, see [“Configure SAML” on page 37](#).

PAM Authentication

Overview of PAM

Pluggable authentication modules (PAM) enable you to determine how applications use authentication to verify the identity of a user. It is an industry-standard technology that extends UNIX host authentication to recognize additional authentication providers. PAM uses *modules* or libraries to access multiple authentication methodologies. SAS Viya supports host authentication.

How It Works in SAS Viya

Default PAM configuration files, `/etc/pam.d/service`, are installed as a part of the SAS Viya deployment process.

Note: For the SAS Cloud Analytic Services (CAS) server, *service* is *cas*. For SAS Studio, *service* is *sasauth*.

For *sasauth* to perform authentication, entries must be made in the PAM configuration files that are provided by SAS. These entries describe what authentication services are used when *sasauth* performs an authentication. This includes the account and auth module types. The session and password modules are not supported.

TIP In a multi-machine deployment, configure PAM on the host with SAS Object Spawner and the host with CAS controller.

For configuration information, see [“Configure PAM” on page 39](#).

Authinfo File

Authentication is used to control access to the CAS server and its resources. Your identity must be successfully authenticated before your session is created. SAS Studio authenticates the connection to CAS by using your user credentials. When password information is not available, an attempt is made to find an Authinfo file (.authinfo is the default filename on Linux). The Authinfo file provides a user name and password to CAS for PAM authentication. It is an alternative to including passwords in programs.

You can also force the use of the Authinfo file by specifying the CAS_AUTH_METHOD environment variable. For more information, see [CAS_AUTH_METHOD on page 648](#).

The Authinfo file is required when you are using the command line to submit commands for the following tasks:

- Run programs in batch mode. The USER= option in the CAS statement or SAS system option CASUSER= can be specified.
- Perform limited server administration using the **casadmin** command.
- Run commands in line mode.

Note: SAS Studio user credentials are used to authenticate your connection to CAS. SAS Studio does not use the Authinfo file for authentication.

Typically, the Authinfo file resides in the `$HOME` directory.

The Authinfo file format is based on the .netrc file specification. The .netrc file format is an older format. You can see the file specification at [Netrc Format](#). In addition to the standard .netrc file standards, the Authinfo specification allows for putting commands in the file as well as using quoted strings for passwords. The quoted strings allow for spaces within passwords.

If the Authinfo file contains values that match the host, port, or user name. The information contained in the Authinfo file is used to connect to CAS.

The following system options and environment variables can be used to override the Authinfo file. These options point to Authinfo files that are located in a different directory or are named differently.

Here are the ways that the AUTHINFO system option, environment variable, and the statement option can be used to override the Authinfo file:

- Environment variable AUTHINFO takes precedence over the Authinfo file.
- SAS system option AUTHINFO= (alias CASAUTHINFO=) overrides the AUTHINFO environment variable as well as the Authinfo file.
- AUTHINFO= option in the CAS statement overrides the AUTHINFO= system option, the AUTHINFO environment variable, and the Authinfo file.

For more information, see the following documents:

- [AUTHINFO= System Option](#)
- [CAS Statement](#)
- [CAS_AUTH_METHOD environment variable on page 648](#)
- [USER=user-ID argument](#)
- [Batch Mode in UNIX Environments](#)

Additional Authentication Topics

SAS/CONNECT Authentication

As an administrator, you might want to enable SAS Viya to accept connections for existing SAS 9 environments. SAS/CONNECT enables that connection, and passes credentials that can be used in the SAS Viya environment.

With SAS Viya, your credentials are used to authenticate to CAS when you are using SAS/CONNECT. When additional SAS/CONNECT servers are spawned, SAS/CONNECT forwards your credentials to the spawned SAS/CONNECT server session.

Here are the ways that SAS/CONNECT and CAS authenticate your user credentials:

- When the user is using any environment that is not a SAS Viya environment, and is connecting to SAS Viya via the SAS/CONNECT spawner, the spawner passes the SIGNON credentials to the SAS/CONNECT server where the credentials can be used to connect to CAS.
- When the user is in the SAS Viya environment using SAS Studio and starting SAS/CONNECT server sessions (using SASCMD SIGNON or the CONNECT Spawner), the CAS credentials (if they exist) are passed to the SAS/CONNECT server in SAS Viya.
- When running SAS Viya in batch or line mode, the Authinfo file is used to authenticate to CAS. If you specified the USER= option in the CAS statement, CASUSER= system option, or if you specified the CAS_AUTH_METHOD environment variable, Authinfo file authentication is used.

For more information, see the following documents:

- [USER=user-ID](#)
- [CAS AUTH_METHOD environment variable on page 648](#)
- [SAS/CONNECT User's Guide](#)
- ["Operate" on page 670.](#)

Single Sign-On

Note: This information does not apply to a programming-only deployment.

Single sign-on (SSO) is an authentication model that enables users to access a variety of computing resources without being repeatedly prompted for their user IDs and passwords. For example, SSO can enable a user to access SAS servers that run on different platforms without interactively providing the user's ID and password for each platform. SSO can also enable someone who is using one application to launch other applications based on the authentication that was performed when the user initially logged on.

The SAS Logon Manager is the central point for handling changes to authentication mechanisms, such as the addition of third-party SSO products. SAS Viya supports the following SSO products:

- [Kerberos on page 45](#)
- [SAML on page 53](#)
- [OAuth on page 53](#)

Dual Authentication

In a dual authentication environment, users are validated against the LDAP server and the host authentication mechanism. The following conditions exist:

- If PAM is configured to use local accounts and those users also log on to the visual components, then those local accounts must match the LDAP server used for SAS Logon Manager.

- If PAM is configured to use an LDAP server, SAS Logon Manager should be configured to use the same LDAP server.
- When directly connecting to the CAS server using SAS Studio or a batch job, the user ID and password that are supplied are authenticated against both the LDAP server and PAM.

Authentication: Guest Access



About Guest Access

Guest access is an optional feature that provides anonymous Read-Only access to a subset of resources and functionality in participating applications. Guest access is supported for viewing reports in the SAS Report Viewer and SAS Mobile BI.

For information about multi-tenancy, see [“Guest Access in Multi-tenancy” on page 583](#).

Enable Guest Access

Note: In a multi-tenancy environment, the following steps must be repeated for each tenant that supports guest access.

- 1 Set the `sas.logon.provider.guest` configuration property, using SAS Environment Manager:
 - a In the applications menu () , select **Administration** ⇨ **Manage Environment**. In the navigation bar, select .
 - b Create a new configuration instance for **sas.logon.provider.guest**, ensuring that you enable the guest access option. For more information, see [“Create Configuration Instances” on page 216](#).
- 2 Add rules that provide the necessary access to functionality:
 - a From the SAS Viya machine where the command line interfaces are installed, create a default profile, if you have not already created one, and sign in. For more information, see [“Create at Least One Profile” on page 185](#).
 - b To modify the authorization rules, run the following commands:


```

sas-admin authorization facilitate-guest

sas-admin authorization update-rule --id guest-files-2 --object-uri /files/files

sas-admin authorization create-rule --object-uri=/files/files/** --permissions Read --guest
--description "Guest Access: Grants permission to a file based on the file's parent" --condition
"isAuthorized(#file?.parentUri) "

sas-admin authorization update-rule --id guest-identities-2 --object-uri /identities/users

sas-admin authorization create-rule --object-uri=/identities/users/@currentUser --permissions
Read --guest --description "Guest Access: Users can see identity information for the current user"
          
```
 - c To modify the direct access controls for the predefined caslibs on the server, using the controls that are defined in the specified source file, run the following command:

Note: The following command must be executed by a user who is a member of the Superuser role.

```

sas-admin cas facilitate-guest --source-file path-to-controls-file --server CAS-server-name
  
```

```
--superuser
```

For more information about the controls file, see [“Facilitate Guest Access” on page 200](#).

3 Add access controls that provide Read access to caslibs that should be accessible to guest users:

- a** From the SAS Viya machine, if you have not already signed in to SAS Viya, sign in using the default profile that was created in the previous step.
- b** Run the following commands as a user who is a member of the Superuser role:

```
sas-admin cas caslibs add-control --server server-name --caslib caslib-name --grant readInfo
--guest --superuser
```

```
sas-admin cas caslibs add-control --server server-name --caslib caslib-name --grant select
--guest --superuser
```

```
sas-admin cas caslibs add-control --server server-name --caslib caslib-name --grant limitedPromote
--guest --superuser
```

4 Use SAS Environment Manager to grant Read access to folders and reports that should be accessible to guest users:

- a** From the **Content** page, identify the folder to which you want to grant Read access to guest users.
- b** Right-click and select **Edit Authorization**.
- c** Click **+** and select **Add Guest**. Grant Read and Read (convey) access. For more information, see [“General Authorization: How to \(Authorization Window\)” on page 110](#).
- d** Click **Save**.

Note: From the **Content** page, you can move folders and objects into the **My Favorites** and **My Folder** folders for the guest user. You can also create and add folder and report shortcuts. For more information, see [“Content Management: How To” on page 257](#).

Connect as Guest Users

Once guest access is enabled, guest users can view reports using SAS Report Viewer and SAS Mobile BI.

- SAS Report Viewer displays a guest login button.
- SAS Mobile BI displays a guest login button when a mobile connection is established.


See Also

- [“SAS Report Viewer Documentation” in SAS Report Viewer Documentation](#)
- [SAS Mobile BI Documentation](#)

Generate Custom Links to Reports

You can create a custom web link for guest users, allowing them to access a specific report. If guest access is enabled, the custom link is configured to bypass the logon page and automatically connect the user as guest. If guest access is disabled, a logon page is displayed, where users can choose to connect as a guest or log on with their credentials.




- 1** From SAS Report Viewer, open the report to which you want to generate a link.

- 2 Click  and then select **Share report** ⇨ **Link**.
- 3 In the Generate Link window, customize the link, if necessary, in the **Link** field.
- 4 Click **Copy Link**. You can paste the link and distribute to guest users.

See Also

“SAS Report Viewer Documentation” in [SAS Report Viewer Documentation](#)

Disable Guest Access

- 1 Set the `sas.logon.provider.guest` configuration property, using SAS Environment Manager:
 - a In the applications menu () , select **Administration** ⇨ **Manage Environment**. In the navigation bar, select .
 - b From the **Definitions** view, select **sas.logon.provider.guest**.
 - c Click . In the Edit `sas.logon.provider.guest` Configuration window, select the option to disable guest access.

Note: The `sas.logon.provider.guest` option is tenant-specific and must be disabled for each tenant.
 - d Click **Save**.

- 2 (Optional) Remove the rules that provide the necessary access to functionality:

- a From the SAS Viya machine, navigate to the `/opt/sas/viya/home/bin` directory.
- b At the command prompt, create a default profile and sign in by entering the following commands:

```
sas-admin profile init
sas-admin auth login
```

- c Modify the authorization rules by running the following command:

```
sas-admin authorization disable-guest-access
```

Note: This command removes the rules that were automatically loaded by the `facilitate-guest` command. If you manually created any custom rules, using either SAS Environment Manager or the command-line interface, you must manually remove those rules. This includes the two rules that you created in [Step 2b on page 57](#). A list of the remaining guest rules can be viewed on the SAS Environment Manager **Rules** page.

- 3 (Optional) Run the following commands as a user who is a member of the Superuser role to remove CAS Access grants:

```
sas-admin cas sessions create --server server-name --name clisession --superuser
```

```
sas-admin cas caslibs remove-control --server server-name --caslib VAModels --grant readInfo
--guest --session-id session-id
```

```
sas-admin cas caslibs remove-control --server server-name --caslib VAModels --grant select
--guest --session-id session-id
```

```
sas-admin cas caslibs remove-control --server server-name --caslib VAModels --grant limitedPromote
--guest --session-id session-id
```

```
sas-admin cas caslibs remove-control --server server-name --caslib ReferenceData --grant readInfo
--guest --session-id session-id
```

```
sas-admin cas caslibs remove-control --server server-name --caslib ReferenceData --grant select  
--guest --session-id session-id
```

```
sas-admin cas caslibs remove-control --server server-name --caslib ReferenceData --grant  
limitedPromote --guest --session-id session-id
```

```
sas-admin cas caslibs remove-control --server server-name --caslib AppData --grant readInfo --guest  
--session-id session-id
```

```
sas-admin cas caslibs remove-control --server server-name --caslib AppData --grant select --guest  
--session-id session-id
```

```
sas-admin cas caslibs remove-control --server server-name --caslib AppData --grant  
limitedPromote --guest --session-id session-id
```

```
sas-admin cas caslibs remove-control --server server-name --caslib Formats --grant readInfo --guest  
--session-id session-id
```

```
sas-admin cas caslibs remove-control --server server-name --caslib Formats --grant select --guest  
--session-id session-id
```

```
sas-admin cas caslibs remove-control --server server-name --caslib Formats --grant limitedPromote  
--guest --session-id session-id
```

```
sas-admin cas sessions delete --server server-name --session-id session-id
```

Note: These commands remove the grants that were automatically defined by the `facilitate-guest` command. If you manually created any custom grants, using either SAS Environment Manager or the command-line interface, you must manually remove those grants.

External Credentials Management

External Credentials: Overview

In addition to logon credentials, users on SAS Viya systems might need external credentials for accessing databases and other third-party products. This document describes administrative tasks to manage external credentials using SAS Environment Manager.

To enable users to retrieve data from existing sources such as databases (for example, Oracle and Teradata connections), the business user must have the appropriate credentials, and SAS Viya must be able to use those credentials.

This document assumes that you are familiar with the data and caslib concepts that are explained in [SAS Cloud Analytic Services: Fundamentals](#).

External Credentials: How To

About the Domains Page



The **Domains** page in SAS Environment Manager enables you to manage the following types of domains and credentials:

Note: The Domains area is available only if you are a member of the SAS Administrators group.

authentication domain	Makes stored credentials (user IDs and passwords) available to designated identities to facilitate connections to servers that require a password.
connection domain	Makes stored credentials (user IDs) available to designated identities to facilitate connections to servers that do not require authentication.
encryption domain	Makes a stored credential (an encryption key) available to designated identities to facilitate loading of encrypted files. See Encryption in SAS Viya: Data at Rest .

Navigation

The Domains area is available if you are a member of the *SAS Administrators* group and you have opted into your assumable groups.

- 1 In the applications menu () , select **Administration** ⇨ **Manage Environment**.
- 2 In the navigation bar, locate the **Security** section and click **Domains** .

- 3 You can select one of two views from the **Domains** page. The default view is **Domains**. From the **View** drop-down list, select one of the following views:

Domains	<p>Lists all domains displayed. Domains is the default view. There are three types of domains: Authentication, Connection, and Encryption. On the Domains page, you can view the information for each domain that is defined, or you can create a new domain.</p> <p>Note:</p> <p>This view is available only to SAS Administrators.</p> <p>Note:</p> <p>This document discusses only the Authentication and the Connection domains. For information about the Encryption domain, see Encryption in SAS Viya: Data at Rest.</p>
Credentials	<p>Enables you to access external databases and other third-party products requiring authentication. Credentials are associated with a specific domain for use with a specific data source type.</p>

Manage Domains

The Domains area is available if you are a member of the *SAS Administrators* group and you have opted into your assumable groups.

Create a New Domain

Create an Authentication or Connection domain.



- 1 In the **Domains** view, click .

TIP You can also create a new domain when you are adding a caslib.


- 2 In the New Domain window, you can specify the following settings:

ID	Create an ID name. Required for both Authentication and Connection domains.
Type	There are three domain types, Authentication, Connection, and Encryption. Select Authentication or Connection from the list of domains.
Identities	From the Select Identities window, you can select from users, groups, and custom groups. To add an identity, see the following item.
User ID	Enter the User ID that has access to the external data. All identities connect using this user ID.
Password	Enter the password for the user ID that can connect to the external database. This is not needed for the Connection domain.
Confirm password	<p>Confirm the password. This is not needed for the Connection domain.</p> <p>Note:</p> <p>If the passwords do not match, you cannot save the domain.</p>
Description	Add a description.


- 3 For additional information about identities, from the New Domain window, click ⓘ.
- 4 Add **Identities**. From the New Domain window, click **+**.
 - a In the left pane of the Select Identities window, select **Users**, **Groups**, or **Custom Groups** from the drop-down box.

Note: A best practice is to use a custom group. Then you can add additional users to this custom group as needed to grant access to the external data. Be sure to assign correct permission for this custom group in the associated caslib Authorization.
 - b Select a user, group, or custom group. You can scroll through the list or search .
 - c Move the selected user, group, or custom group to the **Selected Identities** pane. Click .
 - d Click **OK**.
- 5 Add a user ID.
- 6 Add a password. Click **Confirm password**.
- 7 After you enter all of the parameter settings needed, click **Save**.

View Properties of the Domain


- 1 In the **Domains** view, select an ID of type Authentication or Connection.
- 2 Right-click, and select **Properties**. Or select  from the taskbar. In the Domain Properties window, properties pertaining to that domain are displayed. Examples of properties displayed are the ID, type, description, date created, date modified, who created the domain, and who modified the domain.
- 3 Click **Close**.

View Credentials and Edit Identities of the Domain


- 1 In the **Domains** view, select an ID of type Authentication or Connection.
- 2 Right-click, and select **Credentials**. Or select  from the taskbar. In the Credentials for Domain window, the credentials associated with your domain are displayed.
- 3 You can add, delete, or edit credentials. See [“Manage Credentials” on page 64](#) for the details.

Edit a Domain



If you are a member of the domain, you can change the description. You cannot change the type of domain.

- 1 In the **Domains** view, select an ID that is of domain type Connection or Authentication.
- 2 Right-click, and select **Edit**. Or select  from the taskbar.
- 3 Edit the description of the Authentication Domain.
- 4 Click **Save**.


Refresh a Domain

- 1 In the **Domains** view, you can refresh the view.
- 2 Select  from the taskbar.

Delete Credentials from a Domain

- 1 In the **Domains** view, select an ID that is of domain type Connection or Authentication.
- 2 Right-click, and select **Credentials** . Or select  from the taskbar.
- 3 From the Credentials for Domain window, right-click, and select **Delete**. Or select  from the taskbar.
- 4 In the Caution window, this message is displayed: "Are you sure you want to delete the credential for identity 'identity' with user ID 'userid'?"
- 5 Click **Yes**.

Delete a Domain

- 1 From the Domains area, in the **Domains** view, select the Domain ID. Then right-click and select **Delete**. Or select  from the taskbar.
- 2 In the Delete window, this message is displayed: "If a library is associated with this domain, you will not be able to access the data in the library after you delete the domain. Are you sure you want to delete the connection or authentication domain named '*your-domain*' and all credentials associated with the domain?"
- 3 Click **Yes**.


Manage Credentials

The Domains area is available if you are a member of the *SAS Administrators* group, and you have opted into your assumable groups.

Create New Credentials


From the Domains page, select the **Credentials** view from the drop-down list.

Note: You are creating new credentials for an existing domain.




- 1 To add new credentials, click .
- 2 In the New Credential window, specify general settings as follows:

Domain	Select an existing <i>Authentication</i> or <i>Connection</i> domain.
Identities	In the Select Identities window, you can select from users, groups, and custom groups. See below for instructions on selecting an identity.
User ID	Enter the user ID and password required to access the external data.

Password	Enter the password associated with Identities. Note: If a Connection domain is selected, a password is not required.
Confirm Password	Enter the same password as above.

- 3 For additional information about identities, in the New Credential window, click ⓘ.
- 4 To add identities, in the New Credential window, click +.
 - a In the left pane of the Edit Members window, select **Users**, **Groups**, or **Custom Groups** from the drop-down box.
 - b Move the user, group, or custom group to the right to the **Selected Identities** pane. Click .
 - c Click **OK**.
- 5 Add a user ID.
- 6 Add a password. Click **Confirm password**.
Note: If a Connection domain is selected, a password is not required.
- 7 After you have entered all of the parameter settings needed, click **Save**. These credentials are now associated listed in the Credentials view for the domain that you specified.

View Credential Properties

- 1 In the **Credentials** view, right-click a user ID that you want to view, and select **Properties**. Or select  from the taskbar. The properties are displayed in the Credentials Properties window.
Note: Properties that can be displayed can include User ID, Identity, Identity type, Domain ID, Domain type, Modified By, Date created, Date modified, and Created By. Not all columns are displayed by default.
- 2 Manage which columns are displayed:
 - a At the right edge of the table, click the **Options** icon , and select **Manage Columns**.
 - b From the Manage columns window, select items to move from the **Hidden columns** pane to the **Displayed columns** pane. After selecting the **Hidden columns**, click the **Add** arrow .


TIP To select more than one item, use the Shift key.

Note: You can add or remove columns to be displayed.

 - c Click **OK**.
- 3 (Optional) You can also create a new credential for an existing domain from this view. When you create a new credential, the Domain is already filled in for you in the New Credential window. You can also, delete a credential from this view. Follow the steps in [“Create New Credentials” on page 64](#) and [“Delete Credentials” on page 66](#) for details.


Edit a Credential

If you are a member of the group or custom group, or are a user associated with the selected credentials, you can add Identities and remove Identities (users, groups, and custom groups) from an existing credential. You must supply the credentials to edit.


- 1 In the **Credential** view, select **User ID**.
- 2 Right-click, and select **Edit**. Or select  from the taskbar.

- 3 To add identities, in the Edit Credentials window, click **+**.

To add an identities member, in the Select Identities window, perform the following steps.

- a In the left pane of the Edit Members window, select **Users**, **Groups**, or **Custom Groups** from the drop-down box.
- b Move the user, group, or custom group to the right to the **Selected Identities** pane.
- c Click .
- d Click **OK** to save the information.


To remove an identity, in the Select Identities window, move the user, group, or custom group to the left pane.

- a In the right pane of the Edit Members window, select **Users**, **Groups**, or **Custom Groups** from the drop-down menu.
 - b Move the user, group, or custom group to the left out of the **Selected Identities** pane. Click .
 - c Click **OK** to save the information.
- 4 Change the user ID.

Note: If a Connection domain is selected, a password is not required.
 - 5 Change the password. Click **Confirm password**.
 - 6 After you have edited all of the parameters, click **Save**.

Delete Credentials

Perform the following tasks in the Credentials view.

- 1 Right-click the user ID of the credential that you want to delete, and select **Delete**. Or select  on the taskbar.
- 2 In the Caution window for SAS Environment Manager, this message is displayed: "Are you sure you want to delete the credential for identity 'name' with user ID 'userid'?"
- 3 Click **Yes**.

External Credentials: Concepts

What Are Credentials?

Credentials are associated with identities. Identities can be individual users, groups, or custom groups. A credential enables you to assign a user ID for external data to one or more identities. Passwords are optional. For example, you can use a custom group called OracleUsers as an Identity and assign an Oracle user ID and password. The individual users or groups in this OracleUsers custom group do not need to know the Oracle credentials. These individual users have access through this OracleUsers custom group credential definition.

What Is a Domain?

Overview

Domains are used to store both the credentials required to access external data sources and the identities that are allowed to use those credentials. A domain contains one or more references to identities (users or groups) who have access to the credentials in the domain. A user can access the credentials either directly with their user ID or indirectly as a member of a group that is defined as an identity.

The ID, or name, of a domain is used in the definition of a non-path-based caslib to access and load tables from external databases. A domain is associated with a caslib to provide access. External sources include SAS LASR, Oracle, Teradata, Hadoop, Postgres, and Impala. Users of a caslib with an associated domain do not have to know or enter database credentials to access or load external data.

Note: Cloud Foundry supports these caslib types: Path-based, DNFS, HDFS, LASR, and Hadoop.

There are three domain types: authentication, connection, and encryption.

What Is an Authentication Domain?

An authentication domain is a name that facilitates the matching of logons with the servers for which they are valid. Authentication domains are used to store credentials that are used to access an external source (for example, an Oracle database) that can then be associated with a caslib of the appropriate type.

Each user ID and password is valid within a specific scope. For example, the user ID and password that you use to log on to your computer at work are probably not the same as the user ID and password that you use to log on to a personal computer at home. It is also common for database servers and web servers to have their own authentication mechanisms, which require yet another, different, user ID and password.

The software attempts to use only the credentials that it expects to be valid for a particular resource or system. The software's knowledge of which credentials are likely to be valid is based entirely on authentication domain assignments. For this reason, you must correctly assign an authentication domain to each set of resources that uses a particular authentication provider, and also assign that same authentication domain to any stored credentials that are valid for that provider.

For example, assume that the user wants to define an Oracle caslib name "oralib" and to allow the Oracle users to access this caslib. From SAS Environment Manager, the administrator first creates a custom group of users called "orausers," and then defines a domain called "oracle." For the domain, they add "orausers" to the list of identities that have access to the credentials. They then provide a set of credentials (user ID and password) that give access to the oracle database and schema that they plan to use in the caslib. Note that in this case, all users are accessing the schema using a shared set of credentials. Finally, when defining the caslib, the administrator associates the caslib with the "oracle" domain.

Note: If the intent is for each user to access the schema using his or her own credentials, then each user must be entered as a unique identity with access to the domain, and the specific credentials must be provided with each user's record.

Authenticating to SAS can be done through SAS Logon. See [“Authentication: Overview” on page 31](#).

What Is a Connection Domain?

A Connection domain is used when the external database has been set up to require a user ID but no password.

What Is an Encryption Domain?

An Encryption domain is used to store an encryption key. This key is required to read data at rest. See *Encryption in SAS Viya: Data at Rest* for information about how to use Encryption domains.

Manage Personal Passwords

Introduction to the My Passwords Page

These instructions explain how to manage your stored credentials (personal passwords) for authentication domains using SAS Environment Manager. These domains are used to store credentials for authenticating to a third-party provider. The credentials are associated with users, groups, or both users and groups. This feature provides the ability to edit your passwords.

The My Passwords page is visible to all users, not just administrators. If you access the page while assuming administrator privileges, the functionality on the My Passwords page is the same as if you had not assumed administrator privileges.


Note:

You cannot add or delete credentials from the My Passwords page. You can edit the passwords only for your existing credentials or display the properties for existing credentials. Only a user with Administrator privileges can manage Identities, Domains, and User IDs using the **Domains** page.

The My Passwords page shows credentials only for the currently logged-in user, and only for Authentication domains.

Navigation

The My Passwords area is available to members of the *SAS Administrators* group. When you log on to SAS, you can opt in (yes) or opt out (no) of assumable groups to manage your passwords.


- 1 In the applications menu (☰), select **Administration** ⇒ **Manage Environment**.
- 2 In the navigation bar, locate **Security** and click the **My Passwords** icon .

View Properties



The properties displayed by default are your User ID, Domain ID, Modified By, and Date Modified. Date Created and Created By are hidden columns.

Note: Your credentials are available to edit or view only after an administrator has added your identity to an Authentication Domain. See [“Manage Domains” on page 62](#).

- 1 Select a row to enable the **Properties** and **Edit** buttons on the toolbar.

- 2 To view your Credential Properties, select the **Properties** icon .
- 3 Click **Close**.

View Hidden Columns


- 1 To display hidden columns on the My Passwords page, at the right edge of the table, click the **Options** icon , and select **Manage Columns**.
- 2 From the Manage columns window, select items to move from the **Hidden columns** pane to the **Displayed columns** pane. After selecting the items in **Hidden columns**, click the **Add** arrow .

TIP To select more than one item, use the Shift key.


- 3 Click **OK**.

Edit Your Password

Note: Your credentials are available to edit or view only after an administrator has added your identity to an Authentication Domain. See [“Manage Domains” on page 62](#).

- 1 To change your password, select the **Edit** icon .
- 2 From the Change Password window, select **Password** and enter a new password. Select **Confirm password** and confirm your new password.
- 3 Click **Save**.

Refresh Password Properties

Manually refresh the password properties shown on the My Passwords page by selecting the **Refresh** icon . A toast message is shown indicating that the page has been refreshed.

Delete External Credentials

Your password can be deleted only by an administrator. This must be done from the Domains view for your user identity. See [“Manage Domains” on page 62](#).

Orientation

<i>Two Authorization Systems</i>	72
Introduction	72
Similarities	72
Differences	72
Influences	72
Key Terms	73
Demonstration	73
<i>Impact of Assumable Memberships</i>	77
Introduction	77
Effective Access	77
Access Outcomes	77
Demonstration	77

Two Authorization Systems

Introduction

Authorization is the aspect of security that determines which resources are available to which users. This document introduces the SAS Viya authorization layer, which consists of two authorization systems:

- Cloud Analytic Services (CAS) authorization system
- general authorization system (not applicable to a programming-only deployment)

Each system uses a distinct model to protect a distinct class of resources.

Similarities

- Both systems can share the same identity provider.
- Both systems implicitly disallow any access that is not granted.
- Both systems can be administered using SAS Environment Manager or a command-line interface.

Differences

	CAS Authorization System	General Authorization System
Basis:	DBMS-style access control.	Attribute-based access control.
Targets:	CAS objects, such as caslibs and tables.	Most other objects, such as folders and reports.
Inheritance:	Through a hierarchy of objects (for example, from a caslib to its tables).	Through a hierarchy of containers (for example, from a folder to its members).
Precedence:	By object hierarchy (closest wins), then by identity type (user wins), and then by type of setting (denial wins).	By type of setting (Prohibit always wins).
Row-level access:	You can attach a filter to a grant of the Select permission on a table.	(Not applicable).
Conditional access:	(Not applicable).	You can attach a Boolean expression to any rule.
Highest privileges:	An assumable role (Superuser) is exempt from authorization requirements throughout a CAS server, except for data access requests.	An assumable group (SAS Administrators) is granted broad access throughout the general authorization system.*

* The SAS Administrators group is not unrestricted (exempt from authorization requirements). Access is provided by a predefined rule.

Influences

In the CAS authorization system, memberships, inheritance, and row-level filters can influence access.

In the general authorization system, information about the requesting user, the target resource, and the environment can influence access. Each access request has a context that includes environmental data such as time and device type. Environmental constraints can be incorporated using conditions.

Key Terms




Access control or rule	A composite of authorization elements. CAS example: An access control grants the ReadInfo permission to groupA on caslibA. General example: A rule grants the Add permission to groupA on folderA.
Setting	An indication of whether (and to what extent) access is provided. CAS values: Grant, Row-Level Grant, Deny General values: Grant, Conditional Grant, Prohibit, Conditional Prohibit
Permission	A type of access. CAS values: ReadInfo, Select, LimitedPromote, Promote, CreateTable, DropTable, DeleteSource, Insert, Update, Delete, AlterTable, AlterCaslib, ManageAccess General values: Create, Read, Update, Delete, Secure, Add, Remove
Principal	The user, group, or construct to which an access control or rule is assigned. Examples: UserA, GroupA, Authenticated Users
Target	A resource or set of resources. CAS examples: tableA, caslibA General examples: folderA, reportA
Condition	In a conditional rule, the constraint expression. General example: <code>currentUser() == #preferenceOwner</code>
Filter	In a row-level grant, the constraint expression. CAS examples: <code>User='SUB::SAS.Userid', sales>1000</code>
Effective access	A context-neutral description of the net result of all relevant access controls or rules. Effective access does not incorporate evaluation of conditions. CAS values: Authorized, Not Authorized, Row-Level General values: Authorized, Not Authorized, Conditional
Access outcome	The authorization decision for a specific access request. CAS values: Authorized, Not Authorized, Row-Level Authorization General values: Authorized, Not Authorized

Demonstration



In this demonstration, you enable a set of users to access a caslib and a folder.

- Users can read and write data in the caslib and objects in the folder.
- Administrators can manage the caslib and the folder.

Note: This demonstration assumes that you are a member of the SAS Administrators group, and that the initial settings and memberships for that group are in place.

- 1 Sign in to SAS, and opt in to your [assumable](#) groups.
- 2 To represent the set of users, create a group.
 - a In the applications menu (☰), under **Administration**, select **Manage Environment**. In the navigation bar, click .
 - b At the top of the navigation pane, make sure **Custom Groups** is selected.
 - c Click .
 - d In the New Custom Group window, enter `groupA` as the name and as the ID. Click **Save**.
 - e In the right pane, click  for the **Members** section.
 - f In the Edit Members window, assign members by moving them to the **Selected Identities** pane. Click **OK**.


Note: For this demonstration, adding members is optional.

- 3 To hold shared data, add a caslib.
 - a In the navigation bar, click .
 - b In the **View** drop-down list, select **Libraries**.
 - c Click .
 - d In the New Caslib window, provide information as follows:
 - i Select a server.


Note: The **Server** drop-down list includes only those servers to which you are authorized to add global caslibs.

- ii Make sure the selected data source type is **PATH**.
- iii Enter a path that is relative to and accessible by the CAS server.
- iv Enter the name `caslibA`.
- v In the **Data Source** section, make sure the **Enable encryption** check box is not selected.
- vi Click **Save**.

Note: For details and alternatives, see [SAS Viya Administration: Data](#).




- e Give groupA appropriate access to caslibA.
 - i On the **Data** page, in the **Libraries** view, select `caslibA`. Right-click, and select **Edit authorization**.
 - ii In the Edit Authorization window, click , and add groupA to the window.

TIP In the **Select Identities** window, make sure **Custom Groups** is selected in the drop-down list. Move groupA to the **Selected Identities** list. Click **OK**.

- iii In the Edit Authorization window, notice that groupA has an effective access value of  (Not Authorized) for all permissions. In the **Access Level** column, adjust groupA's gauge to the **Write** access level.

iv Click **Save**.

4 To hold shared reports, create a folder.

- a In the navigation bar, click .
- b Above the list of folders, make sure you are at the top level (**Folders**). Click  to create a new top-level folder. Enter the name `folderA`, and press the Enter key.
- c Give groupA access to folderA and conveyed access to folderA's members.
 - i Select **folderA**, right-click, and select **Edit authorization**.
 - ii In the Edit Authorization window, click , and add groupA.

TIP In the **Select Identities** window, make sure **Custom Groups** is selected in the drop-down list. Move groupA to the **Selected Identities** list. Click **OK**.

iii In folderA's Edit Authorization window, click the effective access icon for groupA's Read permission. In the pop-up window, select **Grant** as the direct setting. Repeat that process for groupA's Add and Remove permissions.



These settings target folderA's object URI and affect access to the folder.

iv In folderA's Edit Authorization window, click the effective access icon in groupA's **Read (convey)** column. In the pop-up window, select **Grant** as the direct setting. Repeat that process in groupA's **Update (convey)** and **Delete (convey)** columns.

These settings target folderA's container URI and affect access to the folder's members. For details, see ["Inheritance" on page 119](#).



v Click **Save**.


5 (Advanced) View direct rules that affect access to folderA.

- a In the navigation bar, click .
- b In the **Rules Filter** pane, under **Object URI**, select **URI** from the drop-down list.
- c In the Choose an Item window, select **folderA**. Click **OK**.
- d In the **Rules Filter** pane, click **Apply**.
- e To ensure that you are seeing all available information, click . Notice that there are two rules that target folderA's object URI:
 - The rule that grants all permissions to you was automatically generated because you added folderA as a top-level folder. Lower level folders do not have automatically generated rules.
 - The first set of changes that you made in the Authorization window created the rule that grants the Read, Add, and Remove permissions to groupA.

Note: For groups and users, the **Principal** column on the **Rules** page contains IDs, not display names.

6 (Advanced) View direct rules that affect access that folderA conveys to its members.

- a At the right edge of the table, click , and select **Manage columns**.
- b In the Manage Columns window, move **Container URI** to the **Displayed columns** pane.
- c In the **Displayed columns** pane, select **Container URI**, and click . Click **OK**.
- d At the top of the **Rules Filter** pane, click the **Reset all** link to clear all filters that are currently in effect.

- e In the **Rules Filter** pane, under **Container URI**, select **URI** from the drop-down list.
- f In the Choose a Location window, select **folderA**. Click **OK**.
- g In the **Rules Filter** pane, click **Apply**.
- h To ensure that you are seeing all available information, click . Notice that there are two rules that target folderA's container URI:
 - The rule that grants all permissions to you was automatically generated because you added folderA as a top-level folder. Notice that the generated rule targets both folderA's object URI and folderA's container URI.
 - The second set of changes that you made in the Authorization window created the rule that targets folderA's container URI, granting the Delete, Read, and Update permissions to groupA. That rule provides conveyed access to the members of folderA.

Note: The **Rules** page does not display CAS access controls. You can use the [command-line interface](#) to view the direct access controls for a CAS object (such as a caslib or table).

See Also

- [SAS Viya Administration: General Authorization](#)
- [SAS Viya Administration: Cloud Analytic Services Authorization](#)

Impact of Assumable Memberships

Introduction

Most memberships are always in effect. For example, if UserA is a member of GroupA, that membership affects UserA all of the time. UserA cannot temporarily opt in or opt out of experiencing the effects of his membership in GroupA.

The most highly privileged memberships are assumable. Assumable memberships are in effect in only certain circumstances. Here are examples:

- In a programming interface or SAS Environment Manager, members of a CAS role can temporarily experience that role's elevated privileges by assuming that role at any time.
- In most visual interfaces, members of the SAS Administrators group can temporarily experience that group's elevated privileges by opting in to that group at sign-in time.

Effective Access

When you examine effective (net) access for a user who has assumable memberships, information about whether those memberships are currently in effect is, in most cases, unavailable.

- In general authorization, effective access information presumes that all assumable memberships are in effect.
- In CAS authorization, effective access information presumes that no assumable memberships are in effect.



Access Outcomes





When a user who has assumable memberships makes an access request, the outcome of that request is affected by whether those memberships are currently in effect.

Demonstration

This demonstration uses SAS Environment Manager to explore the availability of access that you get exclusively through your assumable memberships.

Note: This demonstration assumes that you are a member of the SAS Administrators group, and that the initial settings and memberships for that group are in place. This demonstration is not applicable to a programming-only deployment.

- 1 Examine the availability of your elevated privileges in the general authorization system.
 - a If you are currently signed in to SAS, click your user name in the banner, and select **Sign out**.
 - b In the Sign in to SAS window, click **Sign In**, and sign back in. In the Assumable Groups window, click **No**.
 - c In the applications menu (≡), under **Administration**, select **Manage Environment**. In the navigation bar, notice that there is no  icon.
 - d Sign out, and then sign in again. In the Assumable Groups window, click **Yes**.
 - e In the applications menu (≡), under **Administration**, select **Manage Environment**. In the navigation bar, notice that additional items are present, including a  icon.

- 2 Examine the availability of your elevated privileges in the CAS authorization system.
 - a In the navigation bar for SAS Environment Manager, click .
 - b In the **View** drop-down list, select **Servers**.
 - c Select a CAS server, right-click, and select **Properties**.
 - i In the Server Properties window, notice that none of the sections have an edit icon ().
 - ii Expand the **Superuser Role Membership** section. Verify that you are an indirect member of the Superuser role through your membership in the SAS Administrators group.
 - iii Click **Close**.
 - d Right-click the CAS server again, and select **Assume the Superuser role**. Notice that a message at the top of the page indicates your elevated status.
 - e Right-click the CAS server again, and select **Properties**. Notice that several of the sections have an edit icon (). Click **Close**.
 - f Right-click the CAS server again, and select **Relinquish the Superuser role**.
- 3 Examine the relationship between your assumable memberships in the SAS Administrators group and the Superuser role.
 - a Click your user name in the banner, and select **Sign out**.
 - b Sign back in. In the Assumable Groups window, click **No**, so that your membership in the SAS Administrators group is not in effect.
 - c Navigate back to the **Servers** view on the **Data** () page in SAS Environment Manager.
 - d Select the CAS server that you used in the preceding steps, right-click, and select **Assume the Superuser role**. A message indicates that you cannot assume the Superuser role in your current session.

Note: Initially, your membership in the Superuser role is indirect, through the SAS Administrators group. If you opt out of your assumable membership in SAS Administrators, you do not experience any of the privileges that you obtain exclusively from that membership.

TIP If you want to always be able to assume the Superuser role, add yourself to that role in a way that does not involve the SAS Administrators group. For example, make yourself a direct member of the Superuser role.

See Also

- [SAS Viya Administration: Identity Management](#)

CAS Authorization

CAS Authorization: Overview	81
CAS Authorization: How to (Authorization Window)	82
Introduction	82
Navigation	82
Examine Access	82
Set an Access Level	83
Add a Direct Access Control	84
Remove a Direct Access Control	84
Remove Multiple Direct Access Controls	84
Provide Row-Level (Filtered) Access	85
Identify the Source of Effective Access	86
CAS Authorization: How to (CAS Server Monitor)	87
Introduction	87
Navigation	87
Examine Access to a Caslib	87
Provide Public Access to a Caslib	88
Selectively Share Access to a Caslib	88
Selectively Limit Access to a Caslib	89
Resolve Duplicate Access Controls	90
CAS Authorization: Concepts	91
Scope	91
Key Terms	91
Principals	92
Administrators	92
Inheritance	92
Permissions	92
Permissions by Task	93
Row-Level Access	94
Column-Level Access	97
Access to Actions	97
Authorization Decisions	98
Access Control Transactions	99
Application and Persistence	99
Origins of Effective Access	100
Reduced Visibility: Hidden Caslibs	101
CAS Authorization: Guidelines	102
CAS Authorization: Troubleshooting	103
Unrecognized Principals	103
Unintended Loss of Access	103
CAS Authorization: Interfaces	104

Interfaces to CAS Authorization	104
CAS Authorization: Host Access Considerations	105
Why Host Access Matters	105
Which Host Account Matters	105
Host Access in a Programming-Only Deployment	105
Host Access in a Full Deployment	105
Using CAS to Modify Host Access	106
Protecting Files in the Public Caslib	106

CAS Authorization: Overview

To learn about the Cloud Analytic Services (CAS) authorization system, see “[CAS Authorization: Concepts](#)”.

To manage access, use the [interface](#) that best meets your needs. Here are suggestions:

- To interactively manage access at the caslib, table, and row levels, use the [Authorization window](#) in SAS Environment Manager.
- If SAS Environment Manager is not deployed, use CAS Server Monitor to interactively manage access to caslibs. See “[CAS Authorization: How to \(CAS Server Monitor\)](#)”.
- To script management of CAS access controls, use the command-line interface. See [SAS Viya Administration: Command-Line Interfaces](#).
- For comprehensive programmatic management of CAS access controls, use the [Access Control](#) action set.


CAUTION! Do not rely exclusively on CAS access controls to protect data. You must also consider direct host access. See “[CAS Authorization: Host Access Considerations](#)” on page 105.


CAS Authorization: How to (Authorization Window)

Introduction

These instructions explain how to manage access to caslibs, tables, and rows using [SAS Environment Manager](#).

Navigation

- 1 In the applications menu (☰), under **Administration**, select **Manage Environment**. In the navigation bar, click .

TIP Do not begin by clicking . Authorization information is not displayed in a user or group definition.

- 2 Locate and select a global caslib or table.
- 3 Right-click, and select **View authorization** or **Edit authorization**.

Note: If the **Edit authorization** item is not enabled, you are not authorized to modify access to the object.

Examine Access

Scope

The scope of the display is as follows:

- There is always a row for Authenticated Users.
- There is always a row for you, the currently connected user who is using the display.
- There is a row for each principal that is assigned to an access control that affects access to the current object.
- If you add an identity and do not give that identity at least one direct setting, that identity is automatically removed from the display.
- You cannot directly remove a row. If you remove all direct settings for an identity and there is no other reason for that identity to be displayed, that identity is automatically removed from the display.
- Only the permissions that are relevant for an object (directly or for inheritance purposes) are displayed for that object.
- The display does not reflect the impact of CAS role membership or status.

Permissions


For each [principal](#) and [permission](#), the following icons depict effective (net) access to the current caslib or table.



Authorized



Not Authorized

	Row-Level
	Unknown

Note: A diamond indicates that a permission is directly assigned to the specified principal on the current object.

Access Levels

The **Access Level** column provides an alternative to interacting with individual permissions.

- When you manage access, each access level is a shortcut for adding a set of direct access controls.
- When you view access information, each access level is a shorthand description of a set of effective permissions.

Table A.1 Access Levels

Access Level	Authorized Permissions
No access	None
Read	Only ReadInfo and Select
Write	All except ManageAccess and AlterCaslib
Full control	All
Custom	Any other combination, including any row-level access

Note: Access levels exist only in the presentation layer in SAS Environment Manager. CAS stores and evaluates individual permissions, not cumulative access levels.

Set an Access Level

- 1 Open the Edit Authorization window for a CAS object.
- 2 If the principal that you want to work with is not already listed, click **+**. In the Select Identities window, move the user or group to the right pane, and click **OK**.

Note: If guest access is enabled, you must select **Add Identities** after you click **+**. Or, if you need to add Guest to the display, select **Add Guest** after you click **+**. For information about guest access, see [SAS Viya Administration: Authentication](#).

- 3 In the **Access Level** column, click and drag a gauge to adjust access.

Note: If a gauge is not displayed, select an access level other than **(custom)** from the drop-down list.


Note: Each time you change an access level, direct access controls are added as needed to meet the definition of the new access level. If you want to discard all unsaved changes, click **Cancel**.

CAUTION! Reducing the access level for a group that you belong to might block your access. To preserve your access, make sure you have a higher [precedence](#) (offsetting) direct grant. If you are a Superuser, this precaution is not strictly necessary.

- 4 If you modified access for a group, click **Preview**. Examine the impact of the change on other principals. For example, increasing the access level for Authenticated Users from **No access** to **Full control** affects all authenticated users who do not have a more specific denial.
- 5 Click **Save**.


Add a Direct Access Control

- 1 Open the Edit Authorization window for a CAS object.
- 2 If individual permissions are not already displayed, select the **Show individual permissions** check box.
- 3 If the principal that you want to work with is not already listed, click **+**. In the Select Identities window, move the user or group to the right pane, and click **OK**.

Note: If guest access is enabled, you must select **Add Identities** after you click **+**. Or, if you need to add Guest to the display, select **Add Guest** after you click **+**. For information about guest access, see [SAS Viya Administration: Authentication](#).
- 4 Click the effective access icon (for example, ) for the principal and permission that you want to modify.
- 5 In the pop-up window, select **Grant** or **Deny** in the **Direct Setting** drop-down list.


CAUTION! Before you deny access for a group that you belong to, make sure you have a higher [precedence](#) (offsetting) direct grant. If you are a Superuser, this precaution is not strictly necessary.
- 6 If you modified access for a group, click **Preview** in the Edit Authorization window.
 - Notice that a diamond is displayed in the cell that you modified. The diamond indicates that effective access comes from a direct setting.
 - Examine the impact on other principals. For example, a direct denial for GroupA affects all members of GroupA who do not have their own direct settings.
- 7 Click **Save**.

Remove a Direct Access Control

- 1 Open the Edit Authorization window for a CAS object.
- 2 In a cell that includes a diamond, click the effective access icon.
- 3 In the pop-up window, select **(none)** from the **Direct settings** drop-down list.
- 4 In the Edit Authorization window, notice that the new effective access value is unknown (). Click **Preview**.
 - Notice that the new effective access value is known. Or, if you removed the only setting that made the associated user or group a relevant principal for the current object, the user or group is no longer included in the display.
 - If you modified access for a group, examine the impact on other principals.
- 5 Click **Save**.

Remove Multiple Direct Access Controls

- 1 Open the Edit Authorization window for a CAS object.

- 2 In a row that includes at least one direct setting, click the first cell. The row is selected.
- 3 Click  to remove all direct access controls for the selected identity.
- 4 Notice that effective access for any affected cells is unknown (○). Click **Preview**.
 - Notice that all effective access values are known. Or, if the associated user or group is no longer a relevant principal for the current object, the user or group is no longer included in the display.
 - If you modified access for a group, examine the impact on other principals.
- 5 Click **Save**.

Provide Row-Level (Filtered) Access

To make different subsets of rows available to different identities, set one or more row-level grants. Each row-level grant includes a filter that limits the available rows.

- 1 Open the Edit Authorization window for a CAS table.
- 2 If the principal that you want to work with is not already listed, click **+**. In the Select Identities window, move the user or group to the right pane, and click **OK**.

Note: If guest access is enabled, you must select **Add Identities** after you click **+**. Or, if you need to add Guest to the display, select **Add Guest** after you click **+**. For information about guest access, see [SAS Viya Administration: Authentication](#).

- 3 If individual permissions are not displayed, select the **Show individual permissions** check box.
- 4 In the **Select** column, click an effective access icon.
- 5 In the pop-up window, select **Row-level Grant** from the **Direct setting** drop-down list.
- 6 In the **Row-Level Filter** window:
 - a Specify an expression that includes only the rows that the principal should be able to access. The basic format is: `column-name operator value`. Here are basic examples:

Numeric filter	<code>sales<1000</code>
Character filter	<code>model='ford'</code>
Dynamic filter	<code>user='SUB::SAS.Userid'</code>

For details, see [“Row-Level Access”](#).

Note: If you view or edit a filter that was initially created programmatically, you might see escape characters and a different pattern of quotation marks.

- b Click **OK**.
- 7 In the Edit Authorization window, next to the new setting, notice that a diamond is displayed. The diamond indicates that effective access comes from a direct setting.
- 8 If you modified access for a group, click **Preview**. Examine the impact on other principals.
- 9 Click **Save**.

Identify the Source of Effective Access

To determine which access control causes a particular effective access result, examine the origins information for that result.

- 1 Open the View Authorization window for the target CAS object.
- 2 Click the effective access icon for which you want origins information.
- 3 In the pop-up window, next to the **Effective Access** value, click ⓘ.

Note: The icon is disabled if you have changes that you have neither saved nor previewed.

- 4 In the Origins window, review the displayed information.
 - The **Source object** field indicates where the determinative access control is set.
 - The **Principals** field indicates which identity the determinative access control are assigned to.

Note: If multiple access controls of equal precedence cause the result, multiple principals are listed.

For more information, see [“Origins of Effective Access” on page 100](#).


CAS Authorization: How to (CAS Server Monitor)

Introduction


These instructions explain how to manage access to global caslibs using [CAS Server Monitor](#).

TIP To manage access at the table, column, or row level, use another [interface](#).

Navigation

- 1 In CAS Server Monitor, beneath the **SAS Cloud Analytic Services** banner, click .
- 2 On the **Configuration** page, select **Access Controls**.
- 3 In the **Caslibs** list, select the caslib.

Here are details:

- All of the global caslibs that you are authorized to see are listed.
- To see any new global caslibs, click . The **Caslibs** list is not updated automatically.
- Session and personal caslibs are not listed because you cannot set access controls on them. They are intrinsically private.
- The **Global Caslib Creation** and **Session Caslib Creation** caslibs do not contain data. These special caslibs determine which non-administrators can add and delete caslibs. See the information about caslib management privileges in [SAS Viya Administration: SAS Cloud Analytic Services](#).
- Any predefined caslibs have appropriate access controls.

Examine Access to a Caslib

On the **Configuration** page, under **Access Controls**, the columns for the selected caslib are populated as follows:

Applies To	Specifies a type of access control principal .
Identity	Specifies the name (unique identifier) for an access control principal.
Grant	A check mark indicates that access to the selected caslib is authorized for the specified principal and permission .
Deny	A check mark indicates that access to the selected caslib is not authorized for the specified principal and permission.
Activity	Specifies a permission, such as Select.

The rows for a new custom caslib are initially populated as follows:


- For the user who added the caslib, there is initially a row for every permission, because that user has a direct setting for each permission.

- For Authenticated Users, there is always a row for every permission, because Authenticated Users has an inherited setting for each permission.
- For other identities, there is a row for each direct setting. For example, if UserA has a direct grant of the Select permission, there is a Select row for UserA.

Provide Public Access to a Caslib

To give all users access to a new global caslib that you added:

- 1 On the **Configuration** page, select **Access Controls**.
- 2 In the **Caslibs** pane, select the caslib.
- 3 Click **Edit**.
- 4 In the Edit Access Controls window, adjust settings as follows:
 - a In the **Authenticated Users** row for **Read Info**, select the **Grant** radio button.
 - b Click **Add Row**. In the new row at the end of the page, select **Authenticated Users**, the **Grant** radio button, and the **Select** activity.
 - c If you want to also provide Write access, add rows that grant the following additional permissions to Authenticated Users: **Insert**, **Update**, **Delete**, **Create Table**, **Drop Table**, **Delete Source**, **Alter Table**, **Limited Promote**, **Promote Table** (Promote).


TIP As an alternative to adding each row individually, click **Add Set**, and select **Add Set** from the drop-down list. In the new **Authenticated Users** rows for **Alter Caslib** and **Manage Access**, click  to delete those direct access controls. In the remaining new rows, select the **Grant** radio button.

Note: If you want to provide access to guest users, you must grant access to Guest. Guest is not affected by access controls that are assigned to Authenticated Users. By default, guest access is not enabled. See [SAS Viya Administration: Authentication](#) for information about guest access.

- 5 Click **OK** to save your changes.
- 6 Under **Access Controls**, review the results of your changes.

Selectively Share Access to a Caslib

This example gives UserA Read and Write access to a new global caslib that you added:

- 1 On the **Configuration** page, under **Access Controls**, select the caslib.
- 2 Click **Edit**.
- 3 In the **Edit Access Controls** window, click **Add Set**, and select **Add User Set** from the drop-down list.
- 4 In the next window, enter *UserA* in the **User name** field. Click **OK**.
- 5 In the **Edit Access Controls** window, select the **Grant** radio button in each **UserA** row, except the **Alter CASLib** and **Manage Access** rows.
- 6 At the end of UserA's **Alter CASLib** and **Manage Access** rows, click .
- 7 Click **OK** to save your changes.


8 Under **Access Controls**, review the results of your changes.

Note: UserA does not have rows for **Alter CASLib** and **Manage Access**, because you did not give UserA direct access controls for those permissions. Unless UserA is a member of a group that has direct access controls for those permissions, UserA's effective access for those permissions comes from Authenticated Users.

Here are some additional details about editing access controls:

- To add a single row, click **Add Row**.
- To provide a complete set of editable rows for an identity, click **Add Set** and then select the appropriate item:
 - For a group, select **Add Group Set**.
 - For a user, select **Add User Set**.
 - For Authenticated Users, select **Add Set**.
 - For guest users, select **Add Guest Set**.

Note: By default, guest access is not enabled. See [SAS Viya Administration: Authentication](#) for information about guest access.

- To delete a direct access control, click .
- You cannot change or delete inherited settings. You can add a direct access control that has precedence over an inherited setting.

Selectively Limit Access to a Caslib

About Setting Direct Denials

CAUTION! Identity names that you enter are not validated. For sensitive data, do not grant access to Authenticated Users and then rely on selective direct denials. The safer practice is to verify that access is broadly denied, and then grant access selectively.

CAUTION! Do not block your own access. Before you add a direct denial for a group that you belong to, make sure you have a higher [precedence](#) (offsetting) direct grant. If you are a CAS administrator (Superuser) or Data administrator, this precaution is not strictly necessary.

Block All Access for an Identity


To block all access for an identity:

- 1 On the **Configuration** page, under **Access Controls**, select the caslib.
- 2 Click **Edit**.

Note: If the **Edit** button is disabled, you are not authorized to set permissions for the selected caslib.
- 3 In the **Edit Access Controls** window, click **Add Row**.
- 4 In the new row, select an identity type, enter a name (unique identifier), make sure the **Deny** radio button is selected, and select the **Read Info** activity.
- 5 Click **OK** to save your changes.
- 6 Under **Access Controls**, review the results of your changes.

Block Write Access for an Identity

To block Write access for an identity:

- 1 On the **Configuration** page, under **Access Controls**, select the caslib.
- 2 Click **Edit**.
Note: If the **Edit** button is disabled, you are not authorized to set permissions for the selected caslib.
- 3 In the **Edit Access Controls** window, click **Add Set**, and select **Add Group Set** or **Add User Set**.
- 4 In the next window, enter the user or group name (unique identifier). Click **OK**.
- 5 In the identity's **Read Info** and **Select** rows, click .
- 6 In the identity's remaining rows, make sure the **Deny** radio button is selected.
- 7 Click **OK** to save your changes.
- 8 Under **Access Controls**, review the results of your changes.

Resolve Duplicate Access Controls

You cannot save more than one direct setting for a particular caslib, principal, and permission. Here are examples:

- You cannot save two direct grants of the Update permission for UserA on caslibA.
- You cannot save both a direct denial and a direct grant of the ReadInfo permission for UserA on caslibA.

If the error message `Access control has duplicates` is displayed twice in the Edit Access Controls window, there is one duplicate direct setting. Delete one of the settings that have the error message. One error message remains. You can now save your changes by clicking **OK** again.

CAS Authorization: Concepts

Scope

CAS authorization manages access to the following CAS objects:

- caslibs
- CAS tables and columns
- CAS action sets and actions

CAS authorization requirements do not apply in the following circumstances:

- The requesting user has assumed a role that is exempt from all applicable authorization requirements. For example, the user has assumed the Superuser role and the request is to add a caslib.
- The target object is intrinsically private. For example, the target is a table in a personal caslib, a session caslib, or the session scope of a global caslib.

Note: Not all [interfaces](#) expose all aspects of CAS authorization.

Key Terms

Access control	A composite of authorization elements. Example: An access control grants ReadInfo to groupA on caslibA.
Target	A resource. Examples: tableA, caslibA
Principal	The user, group, or construct to which an access control is assigned. Examples: UserA, GroupA, Authenticated Users
Permission	A type of access. Values: ReadInfo, Select, LimitedPromote, Promote, CreateTable, DropTable, DeleteSource, Insert, Update, Delete, AlterTable, AlterCaslib, ManageAccess
Setting	An indication of whether (and to what extent) access is provided. Values: Grant, Row-Level Grant, Deny
Filter	In a row-level grant of the Select permission, the constraint expression. Example: User= ' SUB : :SAS .Userid', sales>1000
Effective access	A context-neutral description of the net result of all relevant access controls. Values: Authorized, Not Authorized, Row-Level
Access outcome	In an access request, the authorization decision. Values: Authorized, Not Authorized, Row-Level Authorization

Principals

The principal in an access control is the user, group, or construct to which the access control is assigned. The CAS authorization system supports the following principals:

- an individual authenticated user
- a user group (a custom group or a group in your authentication provider)
- Authenticated Users (the construct that represents all authenticated users)

Note: In some programmatic contexts, this construct corresponds to the group that is named *.

- Guest (the identity type that facilitates [guest access](#))

Note: Guest is not part of Authenticated Users.

Administrators

CAS roles provide per-server assumable access to administrative functionality. For example, the Superuser role is exempt from authorization requirements throughout a CAS server, except for data access requests. See [SAS Viya Administration: Identity Management](#).

Inheritance

Access flows through a hierarchy of objects. Each parent object conveys settings to its child objects. Each child object inherits settings from its parent object.

Here are the inheritance relationships:

- Access flows from a caslib to its tables.
- Access flows from a table to its columns.
- Access flows from an action set to its actions.

Note: Each caslib always has inherited denials of all permissions for Authenticated Users. Those inherited denials prevent access if there are no higher precedence grants.

Permissions

Permission	Data Enforcement Levels*			Affected Activities
	Caslib	Table	Column	
ManageAccess	✓	✓	✓	Set access controls.
ReadInfo	✓	✓	✓	View and traverse objects.
LimitedPromote		✓		Promote from source in the same caslib.
Promote	✓			Promote from any caslib.
CreateTable	✓	✓		Save (persist) a table.
DeleteSource		✓		Delete a physical source table.**
DropTable		✓		Remove a table from global scope.**

Permission	Data Enforcement Levels*			Affected Activities
	Caslib	Table	Column	
Select		✓	✓	Read data values.
AlterCaslib	✓			Change the properties of a caslib.
AlterTable		✓		Change the attributes or structure of a table.
Insert		✓		Add rows.
Delete		✓		Delete rows.
Update		✓		Change data values.
Execute				Run an action.
Load				Load an action set.

* You can set permissions at or above the level where they are enforced. See also [“Access to Actions”](#).

** To delete any direct access controls, the ManageAccess permission is required.

Permissions by Task

To complete a task, you must have sufficient access to all relevant data objects. The following tables document permissions that are required for selected tasks.

Table A.2 Simple Tasks

Task (CAS Action)	Caslib	Table	Column
Modify caslib properties	ReadInfo AlterCaslib	-	-
Set table permissions	ReadInfo	ReadInfo ManageAccess	-
Load a table from a caslib's data source (loadTable)	ReadInfo	ReadInfo Select	-
Transfer and load an entire file (upload)	ReadInfo	ReadInfo Select	-
Transfer rows to the server (addTable)	ReadInfo	ReadInfo Select	-
Move a table to global scope (promote)	ReadInfo Promote	ReadInfo	-
Remove a table from global scope (dropTable)	ReadInfo	ReadInfo DropTable	-

Task (CAS Action)	Caslib	Table	Column
Delete a file (deleteSource)	ReadInfo	ReadInfo DeleteSource	-
Persist a file (save)	ReadInfo CreateTable	ReadInfo CreateTable (DeleteSource)	-
Read data	ReadInfo	ReadInfo Select	ReadInfo Select
Insert rows	ReadInfo	ReadInfo Insert	-
Update rows	ReadInfo	ReadInfo Select Update	-

* For promotion of a table within the same caslib, LimitedPromote for the table (instead of Promote for the caslib) is sufficient.

** If the request involves deletion of direct access controls, ManageAccess is also required.

*** DeleteSource is required to replace a source table.

Table A.3 Compound Tasks

Task (CAS Actions)	Required Permissions
Just-in-time load (loadTable + promote)	Caslib: ReadInfo, Promote* Table: ReadInfo, Select
Delete a global-scope table (dropTable + deleteSource)	Caslib: ReadInfo Table: ReadInfo, DeleteSource, DropTable

* LimitedPromote for the table (instead of Promote for the caslib) is sufficient.

For information about who can add and delete caslibs, see the caslib management privileges documentation in [SAS Viya Administration: SAS Cloud Analytic Services](#).

Row-Level Access

Overview of Row-Level Access

A row-level grant includes a filter that limits the Select permission on a table. A user who has row-level access to a table can view only those rows that are within the associated filter. See also [“Application and Persistence” on page 99](#).

For example, you can use a row-level grant to enable groupA to see only those rows in tableA where the value in the Toy_Price column is 25. Here is an overview of the process:

- 1 On tableA, give groupA a row-level grant of Select permission.

Specify the following filter: Toy_Price=25

Note: For detailed instructions, see [“Provide Row-Level \(Filtered\) Access”](#).

- 2 Make sure that groupA has ReadInfo access to tableA and its parent caslib.
- 3 Make sure that groupA is not a member of another group that has a grant or denial of the Select permission on tableA.
- 4 Verify that when a member of groupA accesses tableA, the expected rows are returned.

Syntax for Row-Level Filters

Operator (Alias)	Example Filter
Contains (?) Not Contains	Toy_Type Contains 'cars'
In Not In	Toy_Type In ('dolls' 'cars' 'animals')
Between -inclusive Not Between -inclusive	Toy_Price Between 20 AND 30
Like	Toy_Type Like 'd%'
= (EQ) > (GT) < (LT) <> >= (GE) <= (LE) ^= (NE, ~=)	Toy_Price=25
+ -addition - -subtraction / -division * -multiplication ** -exponentiation () -parentheses -string concatenation	Profit > (Sales * .5)
AND (& OR (, !) NOT	Toy_Type='cars' OR Toy_Type='dolls'
Is Missing Is Not Missing Is Null Is Not Null	Toy_Type Is Not Null

Identity-Based Substitution

Identity-based substitution is a powerful and concise technique for defining row-level access. You can use substitution to implement any number of per-user access distinctions with a single row-level filter.

Identity-based substitution parameters map a user's authenticated ID or group memberships to values in a specified column in your data. Values are dynamically substituted into the filter at run time, as appropriate for each requesting user. Here are the supported substitution parameters:

Substitution Parameter	Example Filter
SUB::SAS.Userid	Determines whether a data value is the same as the requesting user's authenticated ID. <i>empID='SUB::SAS.Userid'</i>
SUB::SAS.IdentityGroups	Determines whether a data value matches any of the requesting user's group memberships. <i>FacilityRegion In ('SUB::SAS.IdentityGroups')</i> Note: The comparison is against each group's unique name, so your data must contain unique group names. In a group definition in SAS Environment Manager, the Group ID field contains the group's unique name.

Example: User ID Substitution

If a tableB has an empID column with values that match the user IDs with which users authenticate, you might assign this filter to Authenticated Users:

```
empID='SUB::SAS.Userid'
```

At request time, each user's ID is substituted into the right side of the expression. In a request from userA, the expression resolves as:

```
empID='userA'
```

As a result, userA gets only those rows where the value in the empID column is **userA**.

Example: Membership Substitution

If tableC has a FacilityRegion column with values that match the unique names for user groups, you might assign this filter to an AllRegions group:

```
FacilityRegion In ('SUB::SAS.IdentityGroups')
```

At request time, each affected user's list of group memberships is substituted into the right side of the expression. In a request from user13 (who is a member of the grp7, grp9, and AllRegions groups), the expression resolves as:

```
FacilityRegion In ('grp7','grp9','AllRegions')
```

As a result, user13 gets only those rows where the value in the FacilityRegion column is **grp7**, **grp9**, or **AllRegions**.

Note: **Authenticated Users** is not one of the listed memberships, because it is an access control principal, not a user group.

Multiple Filters and Cumulative Access

If multiple row-level filters are applicable to a user, only the highest precedence filter provides access. If there is an identity precedence tie (the user is a member of multiple groups, each of which has a filter), the user can access any row that meets any of the filters.

Here are details:

The filters for multiple row-level grants provide cumulative access only if all of the following circumstances exist:

- The requesting user does not have a direct access control for the Select permission.
- None of the requesting user's groups have a direct grant or denial for the Select permission.
- Two or more of the requesting user's groups have row-level grants.

Note: All custom and LDAP groups have equal precedence, regardless of any nested memberships.

Note: A filter for a row-level grant that is assigned to Authenticated Users is never cumulative (joined with other filters by OR). Authenticated Users is a construct that has lower precedence than any group.

Column-Level Access

CAS supports column-level permissions, where a user can access some (but not all) columns in a table. See also [“Application and Persistence” on page 99](#).

You can use the Access Control action set to set column-level permissions. You cannot use SAS Environment Manager or CAS Server Monitor or the command-line interface to set column-level permissions.

To prevent a user from accessing a column, deny the user both the ReadInfo permission and the Select permission for that column. Denying both permissions ensures that the user cannot access the column through any CAS action or interface.

Note: Do not rely exclusively on a denial of the ReadInfo permission on a column to hide that column. Not all CAS actions require the ReadInfo permission at the column level.

CAUTION! Not all interfaces can successfully interact with tables that have column-level permissions. Before you provide a production implementation of column-level permissions, verify that results in all applicable interfaces are acceptable.

For example, in SAS Visual Analytics, column-level access is not supported and can yield unexpected results. If userA lacks access to any column that is included in a SAS Visual Analytics report object, userA cannot see any data in that report object.

Access to Actions

Action sets and actions that have no access controls are available to all users. As a result, almost all action sets and actions are available to all users. In general, the ability to perform a particular task is managed by access controls on the target data, not by access controls on actions.

An exception is actions for adding nodes and stopping the server. The initial configuration denies Authenticated Users the Execute permission for those actions. Initially, only Superusers can add nodes or stop the server.

Here are additional details:

- The ReadInfo and Execute permissions are enforced for actions.
- The ReadInfo and Load permissions are enforced for action sets.
- Unregistered action sets are subject to access constraints that are defined on the `_UNREGISTERED` action set. The initial configuration denies Authenticated Users the Load permission on the `_UNREGISTERED` action set.

Note: An unregistered action set is an action set that is not listed in the database of action sets that SAS provides. SAS solutions use only registered action sets.

Note: An attempt to load an action set that does not exist generates an `access denied` error message, because no such action set is known and registered. For example, if you do not correctly specify the action set name in a load request, an `access denied` error message is generated.

- The identity type Guest has the same access to non-administrative actions as Authenticated Users. That access is effective only for sites that choose to enable guest access.

Authorization Decisions

Precedence

In the CAS authorization system, precedence is determined by where an access control is set and who an access control is assigned to.

- Direct access controls have precedence over [inherited](#) settings.
- The [principal](#) precedence hierarchy is relatively flat. It consists of only the following three levels: 1) individual users, 2) user groups, and 3) the construct Authenticated Users.

Note: All user group memberships are at the same level of precedence, even if groups are nested.

Direct access controls have precedence over inherited access controls, regardless of who the principal is. For example, if only the following access controls exist, then UserA cannot access TableA:

- UserA has a direct grant of ReadInfo on caslibA.
- Authenticated Users has a direct denial of ReadInfo on TableA, which is in caslibA.

Note: One way to enable UserA to access TableA is to add a direct grant of ReadInfo for UserA on TableA. UserA's direct grant has precedence over the direct denial for Authenticated Users.

How Access is Evaluated

Each access request initiates an authorization decision process. That process terminates when an outcome is reached. For example, here is the authorization decision process for the Select permission in a request to access data in a CAS table:

- 1 If there are relevant direct access controls on the table, those access controls determine the outcome as follows:
 - a If there is a setting that is specifically assigned to the requesting user, that setting determines the outcome.
 - b If there is a denial from a group, the outcome is Not Authorized.
 - c If there is a grant from a group, the outcome is Authorized.
 - d If there is exactly one row-level grant from a group, the outcome is Row-Level Authorization (authorized for rows within the applicable filter).
 - e If there are two or more row-level grants from groups, the outcome is Row-Level Authorization (authorized for any row that is within any of the applicable filters). See [“Multiple Filters and Cumulative Access”](#).
 - f If there is a setting for Authenticated Users, that setting determines the outcome.
- 2 If there are no relevant direct access controls on the table, direct access controls on the parent caslib determine the outcome as follows:

- a If there is a setting that is specifically assigned to the requesting user, that setting determines the outcome.
 - b If there is a denial from a group, the outcome is Not Authorized.
 - c If there is a grant from a group, the outcome is Authorized.
 - d If there is a setting for Authenticated Users, that setting determines the outcome.
- 3 If there are no relevant direct access controls on the table or the parent caslib, the outcome is Not Authorized.

Access Control Transactions

If you want to preview the results of changes to CAS access controls before you save those changes, use an access control transaction. Here are examples:

- When you have unsaved changes in a CAS object's Authorization window, you can click **Preview**. Review the results, and then save or cancel your changes.
- When you manage CAS authorization using the command-line interface, you can choose to check out an object, modify its access controls, and then commit or roll back the changes. In the command-line interface, a transaction is automatically started when you check out an object.
- When you manage CAS authorization programmatically, you can choose to start a transaction and check out one or more objects before you make changes. When the transaction is open, any whatIsEffective actions that you run incorporate the effects of your unsaved changes. Review the results, and then commit or roll back your changes.

Here are key points about access control transactions:

- This feature is for only changes to access controls. This feature does not provide transaction support for interactions with data or with any other aspect of CAS objects.
- In SAS Environment Manager, access control transactions are always used when CAS access controls are managed.
- In CAS Server Monitor, access control transactions are never used.
- In programmatic and command-line interfaces, use of access control transactions is optional. You do not have to use a transaction in order to modify CAS access controls. Use access control transactions if you want to preview the results of your changes or ensure that nobody else is modifying access controls for the same objects at the same time.

Application and Persistence

[Row-level filters](#) and [column-level access controls](#) are applied to requests to access data, not to requests to load data. For example, if tableA has a row-level filter that enables userA to see only those rows where the value in the MAKE column is Ford, the filter is applied as follows:

- If userA loads tableA, all rows are loaded.
- If userA accesses tableA, he sees only those rows where the value for MAKE is Ford.

Row-level filters and column-level access controls are not replicated when a user saves a table as a different table. The following information continues the preceding example:

- If userA saves tableA in place, the replaced table contains only those rows where the value for MAKE is Ford. This reduction in scope affects all subsequent loads of tableA and all access by subsequent users. Some users might access fewer rows than intended. The row-level filter still exists and can further reduce access, so no users access more rows than intended.

- If userA saves tableA to a different caslib or with a different name, the new table contains only those rows where the value for MAKE is Ford. This reduction in scope affects all subsequent loads of tableA and all access by subsequent users. Some users might access fewer rows than intended. Because the new table does not have the row-level filter, some users might access more rows than intended.
- If userA deletes tableA, the row-level filter is (by default) persisted.

Like row-level filters and column-level access controls, table-level access controls are not replicated when a user saves a table as a different table.

CAUTION! When a user saves a table as a different table, any direct access controls on the original table, its rows, and its columns are not replicated on the new table. Make sure that any users who are authorized to perform a "save as" interaction on a table understand that the new table is initially protected only by its inherited access controls.

Origins of Effective Access

Origins information explains effective access by answering the question: Why does this identity have this effective access to this object?

Origins information identifies the highest precedence access control that causes the access outcome for a particular identity, object, and permission. If there are multiple tied highest precedence access controls, origins information includes all of them. Additional, lower precedence controls are not included.

The following table provides simple examples of origins information. Each row in the table is for a different (independent) scenario. In each example, we are looking at why UserA has an effective access value of Not Authorized for the ReadInfo permission on TableA. UserA is a member of GroupA and GroupB. TableA is in CaslibA.

Table A.4 Origins: Examples

Highest-Precedence Access Control (or Controls)	Origins Information	
	Object	Principals
On TableA, a direct denial for UserA.	TableA	UserA
On TableA, a direct denial for GroupA.	TableA	GroupA
On TableA, direct denials for GroupA and GroupB.	TableA	GroupA, GroupB
On TableA, a direct denial for Authenticated Users.	TableA	Authenticated Users
On CaslibA, a direct denial for UserA.	CaslibA	UserA
On CaslibA, a direct denial for GroupA.	CaslibA	GroupA
On CaslibA, direct denials for GroupA and GroupB.	CaslibA	GroupA, GroupB
On CaslibA, a direct denial for Authenticated Users.	CaslibA	Authenticated Users
There are no relevant access controls.	Caslib default	

TIP To obtain origins information in the Authorization window, see "Identify the Source of Effective Access" on page 86.

Reduced Visibility: Hidden Caslibs

A hidden caslib is omitted from most lists of caslibs. Tables in a hidden caslib are omitted from most lists of tables.

Hiding a caslib does not protect the caslib or limit access to the caslib's data. Hiding a caslib just reduces visibility by preventing the caslib and its tables from being listed in certain contexts. In those contexts, hidden caslibs and their tables are unlisted for all users and administrators, regardless of roles and access controls. Hiding a caslib affects everyone equally.

Here are some contexts where hidden caslibs are listed:

- In SAS Environment Manager and CAS Server Monitor, hidden caslibs are listed. In CAS Server Monitor, the display indicates which caslibs are designated as hidden.
- In the `tables.caslibinfo` action, hidden caslibs are listed if you specify the value `true` for the `showHidden` parameter.

Two of the [predefined caslibs](#) (`AppData` and `ReferenceData`) are hidden. Users can use the data in those caslibs because appropriate predefined access controls are in place. However, those caslibs (and their tables) are not listed in most contexts. For example, they are excluded from selection lists in SAS Visual Analytics.

TIP Hide a caslib only if you have users who must be able to use that caslib's data but should not see that caslib or its tables in most lists.

A caslib is hidden if its `hidden` parameter is set to `true`. You can set the `hidden` parameter when you add a caslib in [CAS Server Monitor](#) or when you use the `tables.addCaslib` action directly. You cannot view or set the `hidden` parameter in SAS Environment Manager.

CAS Authorization: Guidelines

The following guidelines can contribute to simplicity and security:

- Limit membership in administrative roles.
- Minimize use of individual tables as targets.
- Minimize use of individual users as principals.
- Remember that any access that is not granted is implicitly denied. Do not set unnecessary denials.
- If you deny someone access to part of a table (using a column-level or row-level access control), make sure that identity cannot update or insert rows in that table.
- Perform a backup before and after you make significant changes to your system.

CAS Authorization: Troubleshooting

Unrecognized Principals

If the Authorization window displays a warning icon next to a principal's name, that principal does not exist in the identities service.

- If the principal is a host account (for example, `cas`) that does not exist in your LDAP provider, you can ignore the warning icon.
- Otherwise, determine whether the principal still exists in the identities service.
Note: Deletion of a custom group does not cause automatic deletion of rules in which that custom group is the principal.
- Make sure the identities service can still contact your LDAP provider.

Unintended Loss of Access

Reinstate Access: Instructions for Users


If you inadvertently block your own access to an object, contact an [administrator](#) for assistance.

Note: Anyone who has the `ReadInfo` and `ManageAccess` permissions for the object can reinstate your access.

Reinstate Access: Instructions for Administrators

- 1 In the applications menu () , under **Administration**, select **Manage Environment**.

Note: In a programming-only deployment, use an alternate [interface](#).

- 2 In the navigation bar, click .
- 3 In the **View** drop-down list, select **Servers**.
- 4 Right-click on the server, and select **Assume the Superuser role**.

Note: If the **Assume the Superuser role** action is not available, you are not a member of the Superuser role for the selected CAS server. If your Superuser role membership is exclusively through the SAS Administrators group, make sure you have opted in to that membership in your current session.

CAS Authorization: Interfaces

All CAS authorization requirements and constraints are always fully enforced. However, not all interfaces expose all CAS authorization features.

In the following table, the shaded part of each circle is an approximation of the amount of CAS authorization functionality that a particular interface exposes. The shading indicates relative coverage. The shading does not indicate alignment of coverage across interfaces.

Table A.5 Interfaces to CAS Authorization

● Access Control action set	Programmatic interfaces for CASL (the CAS procedure), Python, Lua, and R.
● REST API	The REST interface for CAS.
● SAS Java Client interface for Viya	The Java programming interface for CAS actions.
◐ Command-line interface	A simple, scriptable interface that includes commands for managing access at the caslib, table, and row levels.
◑ SAS Environment Manager	A graphical enterprise web application for managing access at the caslib, table, and row levels.
◒ CAS Server Monitor	A graphical web application that is embedded in the CAS server. Supports managing access at the caslib level.

CAS Authorization: Host Access Considerations

Why Host Access Matters

If host-layer access requirements are not met, grants in the CAS authorization layer do not provide access.

If host-layer access protections are not in place, denials in the CAS authorization layer do not fully prevent access.

Which Host Account Matters

The account under which a CAS server process runs must have appropriate host-layer access to target directories and files. That account can be an individual user's personal host account or a CAS server's shared service account.

- For requests from a programming context such as SAS Studio, each CAS process runs under the host account of a specific requesting user.
- For requests from a visual context such as SAS Visual Analytics, each CAS process runs under a shared service account, unless the requesting user is a member of the custom group `CASHostAccountRequired`.

Your deployment type and authentication setup impact your host access requirements.

Host Access in a Programming-Only Deployment

In a programming-only deployment, every user accesses a CAS server's host files and directories using his or her individual host account. Users must have host access, so it is possible for them to access the back-end machine directly, bypassing the CAS authorization layer.

If you have sensitive data, ensure that all CAS access distinctions are mirrored in the host authorization layer. For example, if you use the CAS authorization system to deny userA Read access to a path-based caslib called caslibA, you must also set up host access controls that prevent userA from accessing that caslib (directory). Without such protection, userA could use a host command to copy files from the caslibA directory to the directory for a caslib that userA can access from CAS.

Host Access in a Full Deployment

In a full deployment, host access from CAS is sometimes under individual identity and sometimes under a shared identity.

- For users who access CAS only from a programming interface such as SAS Studio, all host access from CAS is under each user's individual identity. For such users, you must mirror CAS layer access distinctions in the host layer.
- For users who access CAS only from a visual interface such as SAS Visual Analytics or SAS Environment Manager, all host access from CAS is under a shared identity. Such users need CAS layer access to data, but they do not need host access to data. Only the shared identity needs host access to data.

For such users, there is no reason to create host access controls that mirror your CAS access controls. Of course, you should always host protect your resources in accordance with your security requirements.

- For users who access CAS from both types of interface, host-layer access and experience vary depending on the type of interface that is used. For example, the personal caslib that they use from programming interfaces is not automatically accessible from the visual interfaces.

You can align access and experience for such users by assigning them to the `CASHostAccountRequired` group. For members of that group, host access from CAS is always under individual identity. Before you use this approach, review the associated limitations. See [“The `CASHostAccountRequired` Custom Group” on page 486](#).

Using CAS to Modify Host Access

You can use CAS to add host access controls to a new directory or file in the following circumstances:

- You add a caslib of the type `PATH` or `DNFS` and specify to create a directory.
- You save a table.

This functionality is provided by the permission parameter in the CAS actions `tables.addCaslib` and `tables.save`.

Here are the available fixed values:

Value	Octal*	Description
Private	700	Grants Read and Write access to only the owner.
GroupRead	750	Grants Read and Write access to the owner. Grants Read access to the owning group.
GroupWrite	770	Grants Read and Write access to the owner and the owning group.
GroupWritePublicRead	775	Grants Read and Write access to the owner and the owning group. Grants Read access to everyone.
PublicRead	755	Grants Read and Write access to the owner. Grants Read access to everyone.
PublicWrite	777	Grants Read and Write access to everyone.

* These octal values are for directories. For saved files (`SASHDAT` and `CSV`), the executable bit is not set.

Here are additional details:

- The owner is the host account that creates the directory or file. the owner always gets full access, regardless of whether you use the permission parameter. Use the permission parameter to further refine access.
- The owning group is the host group that is the primary group for the owner.
- You can specify an octal. You are not limited to the fixed values that are listed in the preceding table.

Protecting Files in the Public Caslib

Initially, all users have host-layer Write access to the directory for the `Public` caslib. If you want only the host account that creates a file in that directory to be able to remove that file, set the sticky bit on that directory.

Note: The sticky bit is a user ownership access flag. If the sticky bit is not set, any user with Write access to the directory can remove files from the directory.

Note: For information about the `Public` caslib, see [“Predefined Caslibs”](#).

General Authorization

General Authorization: Overview	109
General Authorization: How to (Authorization Window)	110
Introduction	110
Navigation	110
Examine Access	110
Provide Access	111
Limit Access	111
Add a Condition	112
Edit a Condition	112
Delete a Condition	112
Remove a Direct Setting	112
Identify the Source of Effective Access	113
General Authorization: How to (Rules Page)	114
Introduction	114
Navigation	114
Rules Page	114
Add a Rule	115
Edit a Rule	115
Copy a Rule	115
Delete a Rule	116
Edit a Condition	116
Delete a Condition	116
Locate a Particular Rule	116
Extend the Ability to Create Top-Level Folders	117
General Authorization: Concepts	118
Scope	118
Key Terms	118
Principals	118
Administrators	119
Inheritance	119
Permissions	120
Permissions by Task	120
Rule Targets	122
Rule Attributes	123
Rule Conditions	125
Authorization Decisions	127
Origins of Effective Access	128
General Authorization: Guidelines	129
General Authorization: Troubleshooting	130
Unexpected Outcomes	130

Unavailable Principals	130
Unrecognized Principals	130
Unintended Loss of Access	130
A Deleted Rule Reappears	131
General Authorization: Interfaces	132
Interfaces to General Authorization	132

General Authorization: Overview

To learn about the general authorization system, see “[General Authorization: Concepts](#)”.

To manage access, use the [interface](#) that best meets your needs. Here are suggestions:

- To adjust access to content (such as folders and reports), use the [Authorization window](#).
- To manipulate rules directly, use the [Rules page](#) or the [command-line interface](#).



Note: A programming-only deployment does not use the general authorization system.

General Authorization: How to (Authorization Window)

Introduction

These instructions explain how to set permissions on content (such as folders and reports) using [SAS Environment Manager](#).

Navigation

- 1 In the applications menu () , under **Administration**, select **Manage Environment**. In the navigation bar, click .





TIP Do not begin by clicking **Users**. Authorization information is not displayed in a user or group definition.

- 2 Locate and select the object.
- 3 Right-click, and select **View authorization** or **Edit authorization**.

Note: If the **Edit authorization** item is not enabled, you are not authorized to modify access to the object.

Examine Access

For each [principal](#) and [permission](#), the following icons depict effective (net) access to the current object:

	Authorized
	Not Authorized
	Conditional
	Unknown

Note: A diamond indicates that effective access comes from a permission that is directly assigned to the specified principal on the current object. If a direct setting exists but does not win (does not determine effective access), a diamond is not displayed.

The scope of the display is as follows:


- There is always a row for Authenticated Users.
- There is always a row for you, the currently connected user who is using the display.
- There is a row for each principal that is assigned to a rule that affects access to the current object. The exception is that internal service principals (for example, sasapp or sas.folders) are not displayed in the Authorization window.

- If you add an identity and do not give that identity at least one direct setting, that identity is automatically removed from the display.
- You cannot directly remove a row. If you remove all direct settings for an identity and there is no other reason for that identity to be displayed, that identity is automatically removed from the display.
- For a non-container object (such as a report), only the Read, Update, Delete, and Secure permissions are displayed. The Create permission is not applicable to an individual content object.
- For a container (such as a folder), two sets of permissions are displayed:
 - The first set of permissions affects access to the object, including the ability to add members to and remove members from the object. This set of permissions has no effect on the folder's members.
 - The second set of permissions affects the access that this object conveys to its child members. See ["Inheritance"](#).


TIP An effective access value of Not Authorized on the conveyed side of a folder's Authorization window does not guarantee that access to child members is not authorized. A direct setting on the child or another influencing rule might provide access to the child member. For example, when you create a top-level folder, the effective access values on the conveyed side for SAS Administrators are all Not Authorized. However, SAS Administrators does have effective access to a folder that you add below the top-level folder. That access comes from a predefined rule that gives SAS Administrators access to all folders.

Provide Access

- 1 Open the Edit Authorization window for the target object.
- 2 If the principal that you want to work with is not already listed, click **+**.

Note: If [guest access](#) is enabled, you must select either **Add Identities** or **Add Guest** after you click **+**.
- 3 Click an effective access icon (for example, .
- 4 In the pop-up window, select **Grant** as the direct setting.

Note: If you cannot change a direct setting, you do not have Secure permission for the current object.
- 5 In the Edit Authorization window, click **Preview**. Examine the impact of your unsaved change.

Note: If there is a relevant prohibit setting anywhere in the system, that setting has precedence over the direct grant that you added. In that case, the effective (net) result is **Not Authorized** () , and a diamond is not displayed.
- 6 Click **Save**.

Limit Access


Any access that is not granted is implicitly denied. The preferred approach is to grant selectively and to avoid use of prohibit settings.

If you must add a prohibit setting, make sure that you do not inadvertently block your own access, particularly for the Read and Secure permissions. If you do block your own access, see ["General Authorization: Troubleshooting"](#).


CAUTION! A prohibit setting has absolute precedence, even if a more specific grant setting exists.

Add a Condition


To provide access within a particular scope or set of circumstances, add a condition.

- 1 Open the Edit Authorization window for the target object.
- 2 If the principal that you want to work with is not already listed, click **+**.
Note: If [guest access](#) is enabled, you must select either **Add Identities** or **Add Guest** after you click **+**.
- 3 Click an effective access icon (for example, .
- 4 In the pop-up window, select **Conditional Grant**.
Note: A conditional prohibit setting does not provide access. A conditional prohibit setting blocks all access within its scope, regardless of any more specific grant settings. A conditional prohibit setting can limit access that is provided by a grant or conditional grant setting.
- 5 In the Condition window, create an expression that specifies the scope and circumstances in which access is granted. Your syntax is validated when you click **OK**. See [“Rule Conditions”](#).
- 6 In the Edit Authorization window, click **Preview**. Examine the impact of your unsaved change.
- 7 Click **Save**.

Edit a Condition

- 1 In the Edit Authorization window for an object, click the effective access icon for the direct conditional setting that you want to modify.
- 2 In the pop-up window, next to the **Conditional Grant** or **Conditional Prohibit** direct setting, click .
- 3 In the Condition window, edit the expression. Your syntax is validated when you click **OK**. See [“Rule Conditions”](#).
- 4 In the Edit Authorization window, click **Preview**. Examine the impact of your unsaved change.
- 5 Click **Save**.

Delete a Condition

- 1 In the Edit Authorization window for an object, click the effective access icon for the direct conditional setting that you want to delete.
- 2 In the pop-up window, next to the **Conditional Grant** or **Conditional Prohibit** direct setting, click .
- 3 In the Condition window, delete the expression. Click **OK**.
- 4 In the Edit Authorization window, click **Preview**. Examine the impact of your unsaved change.
- 5 Click **Save**.

Remove a Direct Setting

- 1 Open the Edit Authorization window for an object.

- 2 In the cell that has the direct setting that you want to remove, click the effective access icon. In the pop-up window, select **(none)** as the direct setting.

Note: If you cannot change the direct setting, you do not have Secure permission for the current object.

- 3 In the Edit Authorization window, click **Preview**. Examine the impact of your unsaved change.

Note: Any identities that are no longer principals are automatically removed.

- 4 Click **Save**.

Identify the Source of Effective Access

To determine which rules contribute to a particular effective access result, examine the origins information for that result.

- 1 Open the View Authorization window for the target content object.
- 2 Click the effective access icon for which you want origins information.
- 3 In the pop-up window, next to the **Effective Access** value, click ⓘ.

Note: If you have unsaved changes, the icon is disabled.



- 4 In the Origins window, review the displayed information.
 - The **Contributing Rules** table lists all applicable rules.
 - To view additional details, add columns to the table.
 - To directly modify a rule, use the [Rules page](#).

General Authorization: How to (Rules Page)

Introduction

These instructions explain how to directly manage general authorization rules using [SAS Environment Manager](#).

Navigation

In the applications menu () , under **Administration**, select **Manage Environment**. In the navigation bar, select .


The **Rules** page is an advanced interface. It is available to only SAS Administrators. You can use a [simpler interface](#) to set permissions on content such as folders and reports.

Rules Page



Use the **Rules** page to manage authorization rules directly. Here are examples:

- View and filter rules.
- Enable and disable rules.
- Replace the principal in a rule.
- View and edit a rule's description or reason.
- Use an existing rule as the basis for a new rule.
- Work with rules that affect access to functionality.
- View conditions for multiple rules at the same time.


Here are additional details:

- To ensure that all rules that should be visible to you are displayed, refresh the display and click **Reset all** in the **Rules Filter** pane.
- On the **Rules** page, you cannot see rules that are directly assigned to objects for which you lack the Secure permission.
- To add, remove, or reorder columns, click  , and select **Manage columns**.
- You cannot sort the values within a column.
- To view a subset of rules, use the **Rules Filter** pane. New and modified filters take effect after you click **Apply**.
- Filtering on the **Rules** page is case-sensitive.
- You can clear a filter by clicking its **Reset** link. You can clear all filters by clicking **Reset all** at the top of the pane.
- The **Guest** principal type is always listed, regardless of whether [guest access](#) is enabled.
- Display names for users and groups are not available on the **Rules** page.
- For details about rule attributes, see "[Rule Attributes](#)".


Add a Rule

- 1 On the **Rules** page, click .
- 2 In the New Rule window, provide values for at least the required [attributes](#). Here are tips:
 - In some fields, you can click  to browse instead of directly entering a value.
 - If a warning indicates that the **Principal** value cannot be validated, make sure the value is an ID, not a display name. If the principal is a service account (such as sasapp or sas.folders), you can ignore the warning.
 - To populate the list of permissions, use the **Clear All**, **Select All**, and **Choose** buttons.
- 3 Click **Save**.
- 4 On the **Rules** page, right-click the new rule, and select **Properties**. Verify that the attributes of the new rule are as you intended.
- 5 If the rule affects a content object (such as a folder or report), use the [Authorization window](#) to verify that the results are as you intended.


Edit a Rule

- 1 On the **Rules** page, select a rule, and then click .
- 2 In the Edit Rule window, modify [attributes](#) as needed. Here are tips:
 - If a warning indicates that the **Principal** value cannot be validated, make sure the value is an ID, not a display name. If the principal is a service account (such as sasapp or sas.folders), you can ignore the warning.
 - If the rule does not have a condition, an **Add Condition** button is present. If the rule has a condition, an **Edit Condition** button is displayed.
- 3 Click **Save**.
- 4 On the **Rules** page, right-click the rule, and select **Properties**. Verify that the attributes of the new rule are as you intended.
- 5 If the rule affects a content object (such as a folder or report), use the [Authorization window](#) to verify that the results are as you intended.


Copy a Rule

- 1 On the **Rules** page, select a rule, and then click .
- 2 In the New Rule window, modify [attributes](#) as needed.
- 3 Click **Save**.
- 4 On the **Rules** page, right-click the new rule, and select **Properties**. Verify that the attributes of the new rule are as you intended.
- 5 If the rule affects a content object (such as a folder or report), use the [Authorization window](#) to verify that the results are as you intended.

Delete a Rule


- 1 On the **Rules** page, select a rule, and then click .
- 2 In the confirmation window, click **Delete**.
- 3 If the rule affects a content object (such as a folder or report), use the [Authorization window](#) to verify that the results are as you intended.

Edit a Condition

- 1 On the **Rules** page, select a rule, and then click .
- 2 In the Edit Rule window, click **Edit Condition**.

Note: If a rule does not have a condition, an **Add Condition** button is present. If a rule has a condition, an **Edit Condition** button is displayed.
- 3 In the Edit Condition window, edit the expression. Your syntax is validated when you click **OK**.
- 4 Click **Save**.
- 5 If the rule affects a content object (such as a folder or report), use the [Authorization window](#) to verify that the results are as you intended.

Delete a Condition

- 1 On the **Rules** page, select a rule, and then click .
- 2 In the Edit Rule window, click **Edit Condition**.



Note: If a rule does not have a condition, an **Add Condition** button is present. If a rule has a condition, an **Edit Condition** button is displayed.
- 3 In the Edit Condition window, delete the expression.
- 4 Click **Save**.
- 5 On the **Rules** page, verify that the condition no longer exists. Right-click on the rule, select **Properties**, and verify that the **Condition** field is blank.
- 6 If the rule affects a content object (such as a folder or report), use the [Authorization window](#) to verify that the results are as you intended.

Locate a Particular Rule


Here are general tips:

- Text that you enter in the **Rules Filter** pane yields results that contain the specified text in matching case.
- Filter requirements are cumulative. For example, if you set two filters, only rules that meet both criteria are displayed.
- Remember to click **Apply** after you set or modify a filter.

Here are tips for locating a rule by date:

- In the **Rules Filter** pane, under **Date Modified**, click  to select a date or date range.
- Each rule's **Date Modified** value indicates when the rule was created or most recently modified.
- To add the **Date Modified** column to the display, click , and select **Manage columns**.

Here are tips for locating a rule by identity:

- The **Modified By** filter is based on who created or last updated a rule.
- To add the **Modified By** column to the display, click , and select **Manage columns**.
- The **Principal** filter is based on who a rule is assigned to.
- Both the **Modified By** and the **Principal** filters use ID values, not display name values. For example, to display only rules that were created by userA, specify `userA` in the **Modified By** filter. To display only rules that are assigned to the SAS Administrators group, specify **SASAdministrators** in the **Principal** filter.
- Rules that are predefined or generated can have a **Modified By** value that does not correspond to a user or group that is known to the identities service.

Here are tips for locating a rule by the URI of the rule's target:

- To view only those rules that target a specific content object, use the technique that is appropriate for the type of URI, as follows:
 - Browse content objects for the object URI for a rule target. In the drop-down list under **Object URI**, select **URI**.
 - Browse container objects for the container URI for a rule target. In the drop-down list under **Container URI**, select **URI**.
- To view only those rules that do not specify an object URI, select **(blank URI)** from the drop-down list under **Object URI**.
- To view rules that either specify `/**` as the object URI or do not specify an object URI, select **(global URI)** from the drop-down list under **Object URI**.
- To view only those rules that do not specify a container URI, select **(blank URI)** from the drop-down list under **Container URI**.

Extend the Ability to Create Top-Level Folders

Initially, only SAS Administrators can create top-level folders. Here are examples of how you can use the **Rules** page to extend that ability to other users:

- To enable all authenticated users to create top-level folders, locate the rule that targets the object URI `/folders/folders` and grants the **Add** and **Read** permissions to **Authenticated Users**. Edit that rule so that it also grants the **Create** permission.
- To enable a group that has the ID `groupA` to create top-level folders, add a new rule that targets the object URI `/folders/folders` and grants the **Create** permission to `groupA`.

General Authorization: Concepts

Scope

General authorization manages access to the following resources:

- content, such as folders and reports
- functionality, such as applications, features, and services

Key Terms

Rule	A composite of authorization elements. Example: A rule grants groupA the Read permission for folderA.
Target	The affected resource, such as an individual object, a set of objects, a service, or a service endpoint. Examples: folderA, reportA
Principal	The user, group, or construct to which a rule is assigned. Examples: UserA, GroupA, Authenticated Users
Permission	A type of access. Values: Add, Create, Delete, Read, Remove, Secure, Update
Setting	In a rule, the indication of whether (and to what extent) access is provided. Values: Grant, Conditional Grant, Prohibit, Conditional Prohibit
Condition	In a rule, the constraint expression. Most rules do not include a condition. Example: <code>currentUser () == #preferenceOwner</code>
Effective access	A context-neutral description of the net result of all relevant rules. Effective access does not incorporate evaluation of any conditions. Values: Authorized, Not Authorized, Conditional
Access outcome	In a context-aware access request, the authorization decision. Values: Authorized, Not Authorized

Principals

The principal in an authorization rule is the user, group, or construct to which the rule is assigned. The general authorization system supports the following principals:

- A user is either an individual authenticated user or a service account.
- A user group is either a custom group or a group in your authentication provider.
- Authenticated Users is the principal type that represents all authenticated users.

- Everyone is the principal type that represents all principals.
- Guest is the principal type that facilitates [guest access](#). Guest is not part of Authenticated Users, but is part of Everyone.

Note: When a principal is deleted, rules that are assigned to that principal are not automatically deleted. Such rules are reused if a new principal of the same type and ID is created. The general authorization system does not have an automated cleanup process for orphaned rules.

Administrators

The SAS Administrators group provides access throughout the general authorization system. A predefined rule grants all permissions throughout the general authorization system to the SAS Administrators group. However, the SAS Administrators group is not unrestricted or exempt from authorization requirements.

For more information about predefined groups, see [SAS Viya Administration: Identity Management](#).

Inheritance

Access flows through a hierarchy of containers. Each container conveys settings to its child members. Each child member inherits settings from its parent container. For example, a folder's child members might include reports and other folders.

Note: A reference member (such as a shortcut) does not inherit access from its parent folder.

You can manage access that a container conveys independently from access to the container. Here are examples of that separation:

- In a folder's Authorization window, the first set of permissions depicts access to the folder, and the second set of permissions depicts access that the folder conveys.
- On the **Rules** page, a rule that targets a folder can affect either or both types of access, depending on which fields (**Object URI**, **Container URI**, or both) are populated.

A rule that targets the object aspect of a container (the container's `objectUri` attribute) has different effects than a rule that targets the container aspect of a container (the container's `containerUri` attribute). Here are details, using `folderA` as an example container.

Rule Target	Potential Impact of the Rule
folderA (as an object)	<ul style="list-style-type: none"> Can affect the ability to read, update, or delete folderA. Can affect the ability to add or remove members for folderA. Settings are not conveyed to the objects within folderA.
folderA (as a container)	<ul style="list-style-type: none"> Settings are conveyed to folderA's child members.
folderA (as an object and as a container)	<ul style="list-style-type: none"> Can affect the ability to read, update, or delete folderA. Can affect the ability to add or remove members for folderA. Container settings are conveyed to folderA's child members.

Some interfaces enable you to create rules that target both aspects of access to a container. However, `containerUri` settings are never derived from or implicitly matched to `objectUri` settings. This separation enables you to provide Write access to the objects in a container without providing Write access to the container itself.

Permissions

Permission	Affected Activity
Create*	Create a new object.
Read	Read an object.
Update	Update or edit an object.
Delete	Delete an object.
Secure	Set permissions on an object (manipulate the object's direct rules).
Add**	Put an object into a container.
Remove***	Move an object out of a container.

* Applicable for a service, service endpoint, or media type.

** Applicable for a container, such as a folder.

*** Applicable for a container, such as a folder. In SAS Environment Manager, also affects the ability to delete a child member from a folder.

Table A.6 *Permission Settings*

Prohibit	Prevents access.
Conditional Prohibit	Prevents access in specified circumstances and scope.
Grant	Provides access, unless there is a relevant Prohibit or Conditional Prohibit setting.
Conditional Grant	Provides access in specified circumstances and scope, unless there is a relevant Prohibit or Conditional Prohibit setting.

Note: Setting is a compound of rule type and whether a condition is present. Setting is a client-layer convenience construct, not a service-layer rule attribute.

Permissions by Task

Introduction

To provide sufficient access to complete a task, you must consider both the availability of functionality and the availability of individual objects. For example, here are the primary requirements for creating and saving a new report:

- the ability to create reports (for example, the Create permission for the service endpoint that controls the ability to create reports). For information about managing access to functionality, see [SAS Viya Administration: Identity Management](#).
- the ability to add members to the target folder (the Add permission for the object aspect of the target folder).
- the ability to see and update the new report (for example, the Read and Update permissions for the target folder's containerUri). See ["Inheritance"](#).

Note: In addition to the requirements that are documented in this topic, most interfaces enable you to interact with only those resources for which you have Read access.

Details for Top-Level Folders

Here are the required permissions for managing top-level folders:

Task	Service URI	Top-Level Folder
Add a top-level folder*	Create	-
Delete a top-level folder	-	Delete
Rename a top-level folder	-	Update
Manage access to a top-level folder	-	Secure

* Initially, only members of the SAS Administrators group can add top-level folders. See [“Extend the Ability to Create Top-Level Folders” on page 117](#).

Initial access to a new top-level folder is as follows:

- The user who creates a new top-level folder has full access to that folder. Automatically generated direct grants provide that access.
- SAS Administrators has full access to every new top-level folder. The predefined rule that grants SAS Administrators permissions for all objects provides that access. See [“Examine Access” on page 110](#).

Details for Child Members

A child member is an object that is in a folder and is not a reference member. For example, folders (other than top-level folders) and reports are child members.

Here are the required permissions for managing child members:

Task	Service URI	Child Member	Current Parent Folder	New Parent Folder
Add	Create	(Read, Update)*	Add	-
Delete	-	Delete	Remove**	-
Update	-	Update	-	-
Rename	-	Update	-	-
Move	-	Update	Remove	Add
Manage access	-	Secure	-	-

* These permissions are required for only objects that are updated during their creation process. For example, the process of creating and saving a new report includes an internal update to the content of the new report. The necessary access is usually conveyed from the parent folder.

** This requirement applies only in SAS Environment Manager.

Initial access to a new child member is determined by inheritance and any other influencing rules, including any automatically generated direct settings.

Details for Reference Members

A reference member is a pointer to another resource. For example, an item in a list of favorite or recent objects is a reference member.

Access to a reference member is independent from access to the referenced resource. For example, having Read access to a favorite that points to ReportA does not equate to having Read access to ReportA. Conversely, having Read access to ReportA does not equate to having Read access to all reference members that point at ReportA.

Access to reference members is as follows:

- Anyone who has Read access to a folder can see all reference members in that folder.
- Anyone who has Remove access to a folder can delete all reference members in that folder.
- You cannot set permissions on a reference member.
- A reference member does not inherit permissions.

Details for Authorization Rules

Anyone who has the Secure permission for a resource can add, modify, and delete direct rules for that resource.

In the initial configuration, the SAS Administrators group can add, modify, and delete all authorization rules.

Details for Projects

Some applications create and manage projects, which are folders that have specialized, application-specific attributes and authorization models. For information about projects, see the documentation for the associated application. Here are examples:

- For information about projects in SAS Model Manager, see [SAS Model Manager: User's Guide](#).
- For information about projects in SAS Data Management, see [SAS Projects: User's Guide](#).

HTTP Mapping

Here are the standard mappings of HTTP verbs to permissions:

POST	Create
DELETE	Delete
GET, OPTIONS, HEAD	Read
PUT, PATCH	Update

Some actions override the default mappings and instead require a different permission.

Rule Targets

Introduction

Each rule affects a target resource (or set of resources), as identified in the rule's target-related attributes (objectUri, containerUri, and mediaType). A rule can specify any, all, or none of the three target-related attributes.

CAUTION! A rule that does not specify at least one target-related attribute affects access to all resources throughout the general authorization system.

A uniform resource identifier (URI) references a resource: a service, service endpoint, or particular object instance within a service. A URI does not correspond to a folder path or include a user-defined object name.

URI Suffix


An objectUri that includes the `/**` suffix affects access to all descendant functionality. Here are examples:

- In an objectUri that references a service, the `/**` suffix affects access to the service and all of its endpoints (sub-functions within the service).
- In an objectUri that references a particular object within a service, the `/**` suffix provides access to the associated service's endpoints in the context of that particular object.

In general, rules that specify the objectUri for a content object (such as a folder or report) should include the `/**` suffix. Inadvertently omitting the suffix narrows the effects of a rule and can yield unintended results due to insufficient access.

Note: The `/**` suffix references descendant *functionality* (service endpoints), not descendant *content objects* (such as reports within a folder). The `/**` suffix has nothing to do with container-based object inheritance. You cannot append the `/**` suffix to a containerUri.

A suffix is included as follows:

- The Authorization window appends the `/**` suffix to new rules that target objectUris. You cannot see or modify URIs in the Authorization window.
- The New Rule window appends the `/**` suffix when you populate the **Object URI** field by clicking  and selecting a content object in the Choose an Item window. You can see and modify URIs in the New Rule and Edit Rule windows.

In other interfaces and contexts, you must remember to include a suffix, when appropriate.

Note: A specialized rule might use the suffix `/*`, which affects access to only immediate descendant functionality.

objectUri

A rule that targets an objectUri affects access to the referenced resource. Here are examples:

- A rule that targets a folder's objectUri affects access to that folder.
- A rule that targets a report's objectUri affects access to that report.
- A rule that targets a service's objectUri (and does not target a specific object instance) affects access to functionality. See [SAS Viya Administration: Identity Management](#).

containerUri

A rule that targets a folder's container URI affects access that the folder **conveys** to its child members.

mediaType

A rule that targets a mediaType applies to all instances of that media type.

Rule Attributes

Target-related attributes:

<code>objectUri</code>	A relative URI that represents a resource such as a report, a folder, a service, or a service endpoint. Character limit: 500
<code>containerUri</code>	A relative URI that represents the container aspect of a container, such as a folder. Rules that specify a <code>containerUri</code> affect access that a container conveys to its child members. Character limit: 500
<code>mediaType</code>	A type of object, such as report. Rules that target a media type affect all object instances for that media type. Most rules do not specify a media type. Character limit: 100

Principal-related attributes:

<code>principalType</code>	Three of the values (Authenticated Users, Everyone, and Guest) represent classes of users. Everyone includes all authenticated users and any <code>guest</code> users. <ul style="list-style-type: none"> ■ Assign broad grant rules to Authenticated Users. ■ Do not assign prohibit rules to Everyone or to Authenticated Users. Such rules block access for all users, including yourself.
<code>principal</code>	The unique string that identifies a particular user or group by its ID. If <code>principalType</code> is <code>user</code> or <code>group</code> , you must specify a value for this attribute. Character limit: 100

Access-related attributes:

<code>type</code>	The indication of whether a rule blocks (prohibit) or attempts to provide (grant) access. Prohibit rules have absolute precedence.
<code>condition</code>	An expression that limits the scope or applicability of a rule. Character limit: 5120
<code>permissions</code>	A list of access types. At least one permission is required.
<code>enabled</code>	The indication of whether a rule is enabled. By default, rules are enabled. To temporarily prevent a rule from being enforced, disable the rule.

Documentation-related attributes:

<code>description</code>	Text that documents a rule for administrative purposes. Character limit: 1000
<code>reason</code>	Text that provides information for end users, where supported by a client. For example, a prohibit rule's reason could be displayed to an end user as part of an <code>access denied</code> message. Character limit: 1000

Rule Conditions

Overview

A condition is a Boolean expression that limits the scope of a rule.

- A rule that has no condition is always applied.
- A rule that has a condition that evaluates to `true` for a particular access request is applied in the authorization decision process for that access request.
- A rule that has a condition that evaluates to `false` for a particular access request is ignored in the authorization decision process for that access request.
- If a rule has an invalid condition, an error is logged and access is restricted as follows:
 - If a grant rule has an invalid condition, the rule is always ignored.
 - If a prohibit rule has an invalid condition, the rule is always applied.

You can specify a condition in inclusive or exclusive terms. Here are two examples:

- A rule grants the Read permission to GroupA for folderA, with a condition that the rule applies only on weekdays. A request from a member of GroupA to access folderA on Sunday is outside the condition. For that access request, the condition is false, so the rule is not applicable (it does not provide access).

Note: A conditional grant rule provides access in specified circumstances, but it does not prevent access outside of those circumstances.

- A rule prohibits the Read permission for GroupA for folderA, with a condition that the rule does not apply on weekends. A request from a member of GroupA to access the folder on Sunday is inside the condition. For that access request, the condition is true, so the rule is applicable (access is blocked).

Note: A conditional prohibit rule prevents access in specified circumstances, but it does not provide access outside of those circumstances.

Condition Syntax

- Conditions are written and stored in [Spring Expression Language \(SpEL\)](#).
- Boolean operators (AND, OR, and NOT) and parentheses are supported. For example, the following condition always evaluates to `true`:

```
(4 < 6) and (5 > 3)
```

- Built-in functions correspond to attributes of the requestor or the environment. You must append `()` to each built-in function (for example, `currentUser()`).

Note: You can use a constant instead of a function. However, functions are often more useful because they are dynamic. At request time, actual context-specific values are dynamically substituted into each function.

- Variables correspond to attributes of the target. You must prepend `#` to each variable (for example, `#userId`).

Note: The available condition variables for a particular type of object are designated in the service for that object type. For example, the preference service designates the `userId` attribute on preference objects as available for use as a condition variable.

Built-In Functions

Table A.7 Location-Based Functions

Function	Description	Type
locale()	Locale of the client that made the request (for example, <code>en_US</code>).	String
remoteHost()	Name of the client machine that made the request.	String
remoteIp()	IP address of the client machine that made the request.	String
serverIp()	IP address of the middle-tier server that received the request.	String
serverName()	Machine name of the middle-tier server that received the request.	String
serverPort()	Port of the middle-tier server that received the request.	int

Table A.8 Target-Based Functions

Function	Description	Type
contentType()	Content type of the target (for example, <code>application/vnd.sas.credential.domain+json</code>).	String
contentLength()	Length of the request.	long
uri()	URI of the target.	String

Table A.9 Time-Based Functions

Function	Description	Type
timestamp	Coordinated Universal Time (UTC) timestamp.	ZonedDateTime
timestamp(zoneId)	Timestamp of the request, based on a specified zoneId.*	ZonedDateTime
localTime(zoneId)	Time of the request, based on a specified zoneId.*	LocalTime
localDate(zoneId)	Date of the request, based on a specified zoneId.*	LocalDate
localDateTime(zoneId)	Date and time of the request, based on a specified zoneId.*	LocalDateTime

* A time zone ID that is valid for [java.time.ZoneId](#).

Table A.10 Other Functions

Function	Description	Type
currentUser()	Identifier for the currently connected user.	String

Function	Description	Type
method()	Method that the request invoked (for example, GET).	String
protocol()	Protocol of the request (for example, HTTP/1.1).	String
header(headerName)	Headers for a specified headerName.	List

Examples of Conditions

This condition makes its associated rule applicable only for weekday requests (in the US Eastern time zone):

```
localDate('US/Eastern').dayOfWeek != T(java.time.DayOfWeek).SUNDAY and
localDate('US/Eastern').dayOfWeek != T(java.time.DayOfWeek).SATURDAY
```

This condition makes its associated rule applicable only if the target's user ID is the same as the requesting user's ID:

```
#userId == currentUser()
```

Evaluation of Conditions

Here are key points about evaluation of conditions:

- If a request does not meet the criteria in a condition, the request is outside that condition. If a request meets the criteria in a condition, the request is inside that condition.
- In a description of effective access, there is no request context, so conditions are not evaluated. Even an atypical condition that is always true (1=1) or never true (1>2) yields an effective access result of Conditional in certain scenarios. A condition is evaluated only in the context of a specific request.
- In an actual request, there is a request context. Any relevant conditions are evaluated, and a definitive answer is provided (Authorized or Not Authorized).

Authorization Decisions

Precedence










In the general authorization system, precedence is extremely flat. The *only* factor that affects precedence is the type of rule (grant or prohibit). Prohibit rules have *absolute* precedence. If there is a relevant prohibit rule, effective access is always Not Authorized.

Neither object inheritance nor identity hierarchy has precedence implications. Here are examples:

- A grant setting that is assigned to you has *less* precedence than a prohibit setting that is assigned to Authenticated Users.
- A direct grant on a report has *less* precedence than a prohibit setting that the report inherits from its parent folder.

Cheat Sheet

In the following table, each row indicates the effective access answer for a separate, independent scenario. For example, if the only relevant rule is a Conditional Prohibit, the effective access answer is Not Authorized (because there is no relevant grant setting).

All Relevant Rules	Effective Access and Explanation
(none)	 Not Authorized (implicit). Any access that is not granted is implicitly denied.
Prohibit	 Not Authorized. A relevant prohibit setting blocks access.
Prohibit + (any other rules)	 Not Authorized. A relevant prohibit setting has absolute precedence.
Conditional Prohibits	 Not Authorized. No relevant grants, no access.
Grant	 Authorized. A relevant grant provides access, if there are no relevant prohibit settings.
Grant + Conditional Grants	 Authorized. Relevant grants provide cumulative access, if there are no relevant prohibit settings.
Grant + Conditional Prohibits	 Conditional. Authorized for requests that are outside all of the prohibit conditions. Prohibit wins, but only within its scope.
Conditional Grants	 Conditional. Authorized for requests that are inside any of the grant conditions.
Conditional Grants + Conditional Prohibits	 Conditional. Authorized for requests that are outside all of the prohibit conditions <i>and</i> inside at least one grant condition.

* This result is due to the lack of a grant, so you can override it by adding a grant.

** This result is due to a prohibit setting, so you cannot override it by adding grants. As long as the prohibit setting exists and is relevant, effective access is not authorized.

For details about conditional access, see [“Evaluation of Conditions”](#).

Origins of Effective Access

An explanation of effective access answers the question, Why does this identity have this effective access to this object? In general authorization, the explanation consists of a list of all relevant rules, including any rules that are applicable but not determinative.

To obtain origins information in the Authorization window, see [“Identify the Source of Effective Access”](#) on page 113.

General Authorization: Guidelines

The following basic guidelines contribute to simplicity and security.

- Minimize use of prohibit rules.
- Limit membership in administrative groups.
- Use groups, not individual users, as principals.
- Use folders, not individual objects, as targets.
- Use conditions only if you cannot efficiently express your authorization requirements another way.
- Perform a backup before and after you make significant changes to your system.

General Authorization: Troubleshooting

Unexpected Outcomes

Here are tips for troubleshooting an authorization outcome that differs from what you expect:

- Make sure all relevant rules are enabled. On the [Rules page](#), right-click a rule to view all of its properties. Or, add the **Enabled** column to the display.
- Make sure you understand the precedence model. See [“Authorization Decisions” on page 127](#).
- Examine the origins information for the unexpected outcome. See [“Identify the Source of Effective Access” on page 113](#).
- If the unexpected outcome relates to inheritance from a folder to objects in that folder, make sure you are using the second set of permissions in the folder’s Authorization window to convey access to the folder’s child members. See [“Inheritance” on page 119](#).
- If the unexpected outcome relates to your access, and you have changed your memberships in the current session, sign out and then sign back in.
- If the unexpected outcome relates to your access, and you are a member of the SAS Administrators group, sign out and then sign back in, indicating whether you want that membership to be in effect. See [“Impact of Assumable Memberships” on page 77](#).
- If the unexpected outcome is for access to a caslib or table, see [SAS Viya Administration: Cloud Analytic Services Authorization](#).

Unavailable Principals

To grant access to a principal that is not in the identities service, use the **Rules** page in SAS Environment Manager or use the command-line interface.

Unrecognized Principals

If the **Rules** page or the Authorization window displays a warning icon next to a principal’s name, that principal does not exist in the identities service.

- If the principal is a service account (for example, sas.folders or sasapp), you can ignore the warning icon.
- If you are using the New Rule or Edit Rule window, make sure that the correct value is selected in the **Principal type** field and the principal’s unique identifier (not display name) is specified.
- If you are using the Authorization window, make sure the identity still exists.

Note: Deletion of a custom group does not cause automatic deletion of rules in which that custom group is the principal.

Unintended Loss of Access

Reinstate Access: Instructions for Users

If you inadvertently block your own access to a resource, contact an administrator for assistance.

Note: Anyone who still has Secure access to the blocked resource can reinstate your access.

Reinstate Access: Instructions for Administrators

To reinstate access that is blocked by a prohibit rule, complete the following steps:

- 1 [Opt in](#) to your assumable membership in the SAS Administrators group.
- 2 Try to reinstate access by disabling, modifying, or deleting the prohibit rule. Here are some tips:
 - If the resource is a content object (such as a folder or report) and you cannot see the resource on the **Content** page, you lack Read access to the resource. Use the [Rules page](#).
 - If the resource is a content object and you cannot make changes in the resource's [Authorization window](#), you lack Secure access to the resource. Either delete the resource (if you have Delete access and the resource is not already in use) or proceed to the next step.
 - If the resource is not a content object, use the **Rules** page.
 - If you know who created (or last modified) a problematic rule or when a problematic rule was created (or last modified), use the **Modified By** or **Date Modified** filter on the **Rules** page to locate the problematic rule.

If you cannot reinstate access, proceed to the next step.

- 3 To temporarily prevent users other than yourself from using the deployment, close current sessions for users other than yourself, and disallow new sessions. See “[Disable Logins](#)” on page 41.
- 4 Temporarily disable self-enforcement of authorization requirements for the authorization service.
 - a In the configuration definition for the authorization service, add a supplemental property named `remote` with a value of `false`.

Note: To learn how to set configuration properties, see [SAS Viya Administration: Configuration Properties](#).
 - b Restart the authorization service.

Note: To learn how to restart services, see [SAS Viya Administration: General Servers and Services](#).
- 5 Disable, modify, or delete the problematic rule or rules.
- 6 Enable self-enforcement of authorization requirements for the authorization service.
 - a In the configuration definition for the authorization service, remove the supplemental property named `remote`.
 - b Restart the authorization service.
- 7 Verify that access is reinstated.
- 8 Make the deployment available again by allowing new user sessions.

If you cannot reinstate access, contact [SAS Technical Support](#) for assistance.

A Deleted Rule Reappears




Some of the predefined rules are bootstrapped by their associated service. If you delete one of those rules, it reappears the next time the service starts. Modifications that you make to such rules are preserved. If you are sure you do not want one of those rules to be in effect, disable that rule (instead of deleting it).

General Authorization: Interfaces

All general authorization requirements and constraints are always fully enforced. However, not all interfaces expose all general authorization features.

In the following table, the shaded part of each circle is an approximation of the amount of general authorization functionality that a particular interface exposes. The shading indicates relative coverage. The shading does not indicate alignment of coverage across interfaces.

Table A.11 Interfaces to General Authorization

 Rules page	The advanced enterprise graphical interface for managing rules directly.
 Authorization window	The basic enterprise graphical interface for managing access to content such as folders and reports.
 Command-line interface	A simple, scriptable interface for managing access to objects and resources.

Backup and Restore

Backup and Restore: Overview

This backup and restore documentation applies to a Linux installation.

The Backup and Restore service, when executed, automatically discovers information about your SAS Viya deployment and backs up critical configuration and user content from your SAS Viya deployment. To protect the integrity of content and configuration information stored in SAS Viya components, SAS recommends that you run the Backup service as part of a regularly scheduled backup process.

Important: The Backup service does not take the place of operating system or file system backups.

Backup and restore functions can be initiated only by SAS Administrators.

Backup and Restore: Terms and Concepts

Backup and Restore Terms

Here is a list of terms used in this document for the Backup and Restore service:

Backup Types

There are two types of backups that are provided by backup service:

- `default`—This is the default value for the backup type. When the backup type is default, `pg_dump` is used to back up the SAS Infrastructure Data server. `pg_dump` backs up the content in the database but does not include the PostgreSQL metadata such as user information, roles, and permissions

The default backup requires that the PostgreSQL metadata is not significantly changed. The default backup type can be fully restored using the SAS restore operation. The restore includes the Postgres data content, as well as the configuration data and other data sources.

- `binary`—When the backup type is binary, the `pg_basebackup` utility is used to back up the SAS Infrastructure Data Server. This takes the backup of all the binaries of the PostgreSQL including metadata such as user information, roles, and permissions. You can also back up other data sources by setting `'includeAllSourcesForBinaryBackup'` to `true` while initiating the backup.

The PostgreSQL portion of a binary backup must be restored manually. If the `includeAllSourcesForBinaryBackup` is set to `true` and the manual restore of PostgreSQL is completed, a SAS restore of the other portions of the binary backup could restore configuration data and whatever other data sources were included.

Deployment backup

a multi-tenant deployment in which only a *Provider Administrator* can initiate backup for multiple tenants in a single API call for those tenants. If an explicit list of tenants is not provided, the backup for all ONBOARDED tenants is triggered.

Deployment restore

a multi-tenant deployment in which only a Provider Administrator can initiate a restore of a Deployment Backup (with at least one successful tenant Backup) for multiple tenants in a single API for those tenants. The restore request accepts the list of tenants to be restored. If this list is not provided, the backup service creates this list with the tenants that are onboarded and are a provider. If all the tenants in the list are present in the backup, then the restore is triggered for each of the tenants on the list. If any of the tenants on the list are not present in the backup, then the restore does not proceed and is marked as failed.

Intra-tenant administrator

a SAS Administrator that is responsible for the administration of each tenant's internal resources. For example, assigning users to custom groups and managing access to SAS Viya content and CAS data are intra-tenant tasks.

Local Vault

A local file system path located on the same host as where a backup source resides. It is the location where the backup files for the data source are created and then moved to the shared vault.

Multi-tenant deployment

a SAS Viya deployment in which multiple tenants can access the same environment in isolated fashion such that data of tenants are not affected or impacted by other tenant's data or process. A multi-tenant deployment has the provider tenant by default.

Pre-restore validations

validations that are done before performing a restore using a given backup. It includes the following validation checks:

- Does the provided backup exist?
- Is the backup completed?
- Is the backup purged?
- If PostgreSQL is being restored using the default type of backup, does the list of databases in the backup match the list of databases currently present in Postgres?
- If the restore is using the binary type of backup, is the *includeAllSourcesForBinaryBackup* property set to `true`? If the *includeAllSourcesForBinaryBackup* is set to `false`, the restore fails. To restore the PostgreSQL portion of a binary backup, see [Restore an Unresponsive SAS Infrastructure Data Server](#) .
- In a multi-tenant environment where the tenant list is not provided, the backup service checks to see whether the onboarded tenants and the tenants in the backup match. If the tenant list is provided in the restore request, then the backup service checks to see whether all of the tenants specified in the restore request are onboarded.

Provider administrator

a SAS Administrator user who belongs to the 'provider' tenant in a multi-tenant deployment and is responsible for administering system-level operations for all the tenants like backup.

Provider tenant

the initial tenant (that is, tenant zero) created when a multi-tenant system is deployed. This tenant has full access to all applications in the deployment but is intended for provider administrator access only. Users in this tenant have access to information about the entire deployment, including other tenants.

Retention period

number of days that backups are stored before they are removed from the shared vault.

sas-viya-deploymentBackup-default

service used to schedule a default backup.

Shared Vault

any network location to preserve the backups from all tiers. The backup files are moved from local vault to shared vault. The install user of the SAS system should have access to this location. It is referred to as `sharedVault` in the SAS Environment Manager user interface.

Single-tenant deployment

a SAS Viya installation intended to be used only by a single tenant.

Slug

user-provided name for the backup or restore operation.

State

state of a backup or restore operation. Possible values for this attribute are as follows:

- `pending`—a backup or restore job has been created but the operation has not yet started.
- `running`—a backup or restore is in progress.
- `completedWithWarning`—at least one tenant backup or restore has failed or the CAS controller host is not reachable.
- `completed`—the backup or restore operation has completed successfully.
- `failed`—a backup or restore for one or more data sources failed.
- `Unknown`—indicates that either a backup agent is not installed on the source or the backup agent is not running. If it does not have a backup agent, contact your administrator. If it has a backup agent, then restart the backup agent service.

Tenant

one of the customers using a shared SAS Viya deployment.

In a multi-tenant system, a tenant is said to be onboarded when the SAS Viya infrastructure for that tenant is created. This includes the LDAP groups, the LDAP identities, the SAS Infrastructure server databases, the schemas, and the SAS Cloud Analytics Server instance.

Trigger

an event generated periodically by a scheduler that signals when a new instance of a job should be executed.

About the Backup and Restore Service

The Backup and Restore service is provided as a way to synchronize the backup and restoring of content and configuration information stored in the following components of a SAS Viya deployment:

- SAS Infrastructure Data Server
- SAS Configuration Server
- SAS Message Broker
- SAS Cloud Analytic Services (CAS Access Controls, Caslib information)

The Backup and Restore service has the following caveats:

- Backup and restore can be initiated only by SAS Administrators.
- The Backup and Restore service requires that the SAS Infrastructure Data Server is running.

What Is Backed Up

The backup service backs up the following components of your SAS deployment:

- **SAS Infrastructure Data Server (PostgreSQL)** – All of the PostgreSQL tables in the database managed by the SAS Infrastructure Data Server are backed up by the Backup and Restore service. If the SAS Infrastructure Data Server is clustered, only the tables on the primary PostgreSQL node are backed up. PostgreSQL metadata (system catalogs) such as user information, roles, and permissions that are stored in the SAS Infrastructure Data Server are backed up only when a binary backup is used.

In addition, any scheduled jobs such as automatic report distribution jobs and scheduled backup jobs can be properly restored only when a binary backup is used.

- **SAS Configuration Server (Consul)** – The service configuration registry that serves as a central repository for configuration data, service discovery, and health status. Only the service configuration that is registered with the configuration service and accessible to the tenant initiating the backup is backed up. This implies only a subset of the Consul Key Value store is backed up.
- **SAS Message Broker (RabbitMQ)** – RabbitMQ configuration such as the queue definitions.
- **SAS Cloud Analytic Services (CAS Access Controls)** – CAS configuration containing CAS Access Controls and Caslib Information.

These sources might reside on single machine or on different machines.


What Is Not Backed Up

The backup service does not back up the following:

- The SAS Viya deployment. Files that are included as part of your SAS Viya deployment are not backed up. The Backup and Restore service backs up content and configuration information but is not provided as a way to back up and restore a SAS Viya deployment.
- In a SAS Message Broker backup, messages or data from the queues are not backed up.
- In a clustered CAS environment, the backup action is invoked only on the primary controller.
- Data stored in data sources outside of the SAS Infrastructure Data Server are not backed up.
- Data that is loaded to CAS is not backed up.
- PostgreSQL metadata such as user information, roles, and permissions that are stored in the SAS Infrastructure Data Server are not backed up when the default backup type is used.
- Scheduled jobs such as automatic report distribution jobs and scheduled backup jobs can be properly restored only when a binary backup is used.
- The Backup and Restore service does not take the place of the operating system and file system backups.
- The SAS Message Broker is not backed up when the backup is initiated by an intra-tenant administrator. In a multi-tenant deployment, an intra-tenant administrator cannot take a binary type of backup.

Backup and Restore: Getting Started

A `DEFAULT_BACKUP_SCHEDULE` is created for you by the deployment process. The `DEFAULT_BACKUP_SCHEDULE` is set to run every Sunday at 1:00 a.m. There are some additional steps required before you can use the `DEFAULT_BACKUP_SCHEDULE`.

- 1 Confirm that the `DEFAULT_BACKUP_SCHEDULE` exists in the SAS Environment Manager. To do so, log on to the **SAS Environment Manager** and click . If the `DEFAULT_BACKUP_SCHEDULE` does not exist, you must restart the `sas-viya-deploymentBackup` service and check again.

There are a couple of reasons why the `DEFAULT_BACKUP_SCHEDULE` might not exist.

- a If a service that is starting fails to register itself with SASLogon, the service tries multiple times to do the client registration until the client registration is done. If this is the case, then the default backup is not created, since it is not able to generate client tokens to access other services such as scheduling or `jobExecution`. The following error message is displayed:



This service is not available which is required for scheduling default backup: "Access token denied." Cannot schedule backup since maximum retry attempt is reached and one of the dependent services is still not running.

- b While backup service is starting, other services have not yet started. For more information about what services should be running, see [How the Restore Process Works on page 146](#) .

In this scenario, the deploymentBackup service retries 25 times to schedule the default backup. If one or more of the services is not running when the 25 times is done, you see the following error message:

```
This service is not available which is required for scheduling default backup:
%name-of-service%. Cannot schedule backup since the maximum retry attempt is
reached and one of the dependent services is still not running.
```

Note: %name-of-service% is the service that did not start.

- 2 Complete the steps for [Backup Configuration Properties](#) to set the sharedDefault location. Ensure that the sharedDefault location has Read and Write permissions to any user that is executing the backup plus the cas user.
- 3 In the **SAS Environment Manager** click . In the **Jobs** list, right-click the DEFAULT_BACKUP_SCHEDULE.
- 4 Select **Run** from the pop-up menu to immediately run the backup. If you do not do this, the backup runs on Sunday at 1:00 a.m.
- 5 In the **SAS Environment Manager** click . Confirm that the DEFAULT_BACKUP_SCHEDULE is in the list.

SAS Infrastructure Data Server Binary Backup

Overview

The backup service takes the “binary” type for the backup of PostgreSQL using the [binary backup](#) and the “default” type for the backup using the [default backup](#). The PostgreSQL backup taken using the binary backup is useful to restore an unresponsive Postgres that occurs for reasons such as the corruption of Postgres system configuration data due to hard drive or memory issues. Without this type of backup, it is not possible to recover on unresponsive PostgreSQL server. The restore mechanism using binary backup is a manual process that is performed by the administrator. See [Restore an Unresponsive SAS Infrastructure Data Server](#) for the manual steps required to restore PostgreSQL.

When to Use the Binary Backup and When to Use the Default Backup

Binary Backup

The binary backup uses pg_basebackup, which makes a binary copy of the database cluster files, making sure that the system is put in and out of backup mode automatically. Backups are always taken of the entire database cluster, including the system catalogs, which contain the system configuration that is common for all the databases. The system configuration includes database users, their permissions, databases present in the server, their schemas, as well as other configuration data.

The default backup does not back up this common data, and it cannot restore this system information if it becomes corrupted. Therefore, it is recommended that the user create a new schedule for the binary backup. If there is any change in the system catalog, the binary backup should be run to back up those changes. For more information about system catalog, see <https://www.postgresql.org/docs/9.4/static/catalogs.html>.

Default Backup

The default backup uses `pg_dump`, which dumps a single database that is provided as an option to the `pg_dump` utility. It is not possible to restore individual databases or database objects from the backup taken by the binary backup. If a particular database is corrupted and needs to be restored without affecting other databases, the default backup should be used.

The default backup is essential in the following situations:

- In multi-tenancy, it can back up databases selectively, whereas the binary backs up all of the databases.
- Disaster Recovery on an alternate host is not possible with the backup taken by the binary backup. The binary backup overwrites the system catalog from the original environment to the destination environment. Therefore, a default backup is required.

How to Use the Binary Backup and the Default Backup

For an immediate backup, you can initiate a default backup through the SAS Environment Manager. You can manually schedule a binary backup. For more information see [Schedule a Backup](#) .

Other data sources such as the SAS Infrastructure Data Server, the SAS Cloud Analytic Services (CAS Access Controls, Caslib information), and the SAS Message Broker are not backed up by default with the binary backup. They are backed up with the default type of backup. You need to specifically request the backup of other data sources along with Postgres when using the binary backup. This is done by setting the `includeAllSourcesForBinaryBackup` parameter to true in the backup request body.

If data is corrupted, you first must try to recover using the default backup. To do this, check the history for last successful backup and then recover from that. If Postgres is not responsive, getting the backup history and initiating a regular restore is not possible. Even the `pgadmin` client cannot connect and access a corrupted database. If this is the case, you must recover Postgres manually using the binary backup. See [Restore an Unresponsive SAS Infrastructure Data Server](#) for the manual steps required to restore PostgreSQL. After manually restoring using the binary backup, Postgres is responsive. Then you can perform a regular restore from the latest successful default backup to restore the content and configuration to a point in time as late as possible.

Implementation Details

The backup service's POST `/backups/jobs` rest endpoint accepts the backup request payload that contains the parameters `backupType` and `includeAllSourcesForBinaryBackup`. These parameters indicate which type of backup (binary or default) is to be used for backing up Postgres.

The `backupType` parameter in the backup request can have one of the following values:

- *default*—indicates that Postgres is being backed up using the [default backup](#) . This is the default value if the parameter is missing in the request body.
- *binary*—indicates that Postgres is being backed up using the [binary backup](#) .

Note: The binary backup is made over a regular PostgreSQL connection, and uses the replication protocol. The connection must be made by a superuser or by a user with REPLICATION permissions. `pg_hba.conf` must explicitly permit the replication connection. For more information about prerequisites of binary backup, see <https://www.postgresql.org/docs/9.5/static/app-pgbasebackup.html>.

When the `backupType` is set to `binary`, you can use the `includeAllSourcesForBinaryBackup` parameter. This parameter, if provided in the backup request, has an option to also back up other sources, while taking a backup of Postgres using binary backup utility.

The `includeAllSourcesForBinaryBackup` parameter can have one of the following values:

- `false`—indicates only Postgres is backed up. This is the default value for this parameter.
- `true`—indicates that other sources are to be backed up along with the Postgres backup. In the case of a multi-tenant environment, all tenants are included in this backup by default.

The `backupType` and `includeAllSourcesForBinaryBackup` parameters are also visible in the details of the backup. This is helpful in identifying if the backup contains the backup taken using the binary backup or the default backup and if other data sources are present in this backup.

Note: While restoring a binary backup, Postgres should be restored before all other data sources. Otherwise, restoring the other data sources fails due to an unresponsive Postgres.

Implications on Current Backup and Multi-tenancy

- 1 The binary backup is an additional mechanism to back up the Postgres. This is **not** a replacement to the existing default backup. However, these two backups are mutually exclusive. That is, a backup that contains a Postgres backup using binary backup would not contain a backup using default backup and vice versa.
- 2 The binary backup does not support backing up individual databases. The binary backup is always a backup of Postgres for the entire deployment. To support the backup and restore of individual tenants, the default backup is used.
- 3 Two schedules are required for backup. By default, a backup schedule is present for the default type of backup. You must manually schedule the binary type of backup.

Note: In a multi-tenant environment, only a Provide Administrator User can execute an ad hoc binary backup **or** can create a schedule for binary backup.

For more information about scheduling a binary backup see [Schedule a Binary Backup](#) . The binary backup used to support the restore of Postgres using its binaries, should be scheduled monthly. Both of the schedules can be configured based on the customer's requirement. However, the binary backup schedule (monthly) can be less frequent compared to the default backup schedule (weekly).

Backup and Restore: Guidelines

Best Practices for Configuring Backups

- Always ensure the values of the `sharedVault` and `retentionPeriod` are set immediately after situations such as a fresh installation, an upgrade, or any modifications to your SAS deployment.
- Ensure that the `sharedVault` is accessible to the install user.

Note: The shared vault location must be different from the local vault location.

- Set the `retentionPeriod` value such that you always have at least the last three to four backups at any point in time. For example, if you are doing daily backups, the `retentionPeriod` must be 4 days. If you are doing weekly backups, the recommended `retentionPeriod` is 30 days.
- Scheduling is not supported out-of-the-box for binary backups. It is recommended that the administrator schedule a call to the backup REST API to perform scheduled backups. Backups should be scheduled at non-peak hours.

- If an upgrade has happened for this SAS Viya setup, log on to the SAS Environment Manager application. Navigate to the backup service configuration screen and make sure that the *sharedVault* property value is set to the intended location. For a custom property, add the key `restore.filter.sas.configuration.config.sas.deploymentbackup` and the value `'*'` (without quotation marks). This property should be added under 'supplimentalProperties' section using the **Add Property** option.
- Always initiate a binary backup after a tenant has been onboarded or offboarded.

Best Practices for Performing Backups

- Always use the Backup and Restore service to perform backups of the content and configuration that is stored in SAS Viya components. This is because the backup service automatically discovers what services are deployed and finds newly deployed services that should be included in the backup. The backup service also finds content and configuration data from your SAS Viya deployment. The backup service backs them up at the same point in time, which is required for a same point-in-time restore of content and configuration data.
- Configure the backup service immediately after situations such as a new installation, upgrading software, or after making any modifications to your SAS Viya deployment. See Backup and Restore: Service Configuration for more details.
- It is recommended that you schedule a binary backup so that it can be used to recover the SAS Infrastructure Data Server when it becomes unresponsive.
- Run the Backup service after making any modifications to your SAS Viya deployments. Examples of modifications include, but are not limited to, deploying SAS Viya, installing software updates, changes to topology, modifications to the SAS Viya configuration, including configuration properties.
- Old backups are purged after the retention period. If you do not want any of your backups to be deleted after the retention period, you must manually archive the backups to a safe location before they are purged.

Best Practices for Postgres on a Single-node When the SAS Infrastructure Data Server Is not Configured for High Availability

As previously described in the [About the Backup and Restore Service on page 135](#) section of this document, performing a restore with the Backup and Restore service requires that the SAS Infrastructure Data Server is running. In an event where the SAS Infrastructure Data Server does not start due to PostgreSQL corruption or for any other reason, you cannot perform a restore with the Backup and Restore service. SAS Viya deployments, which implement the SAS Infrastructure Data Server for HA, have a standby server (or node). In an event where the primary node is unresponsive, a standby node can be promoted to primary status. If promotion of the standby node to primary node is successful and the SAS Infrastructure Data Server can be started, you can then perform a restore.

If you have not configured the SAS Infrastructure Data Server for High-Availability (which is the default configuration), only a single PostgreSQL data server is configured. When only a single PostgreSQL data server is configured, it is recommended that you use the backup service to take a binary type of backup to create a binary copy of the data to be used as a basis for a point-in-time restore. It is recommended that you initiate a binary backup after any configuration or installation changes are made to the SAS Viya environment. For information about how to restore the SAS Infrastructure Data Server when the server is unresponsive due to PostgreSQL corruption or for any other reason, see [Restore an Unresponsive SAS Infrastructure Data Server](#) .

Best Practices for Performing Restores

- Always use the Deployment Backup and Restore service to perform restores to ensure a same point-in-time restore of content and configuration data.

- Always choose the most recent successful backup to perform a restore operation.
- Ensure minimum use of the systems. When performing a restore, only the following services should be running. For more information about what services are running, see [Machines](#) under the “SAS Environment Manager Functions” section.

```
sas-viya-consul-default
sas-viya-vault-default
```

```
sas-viya-sasdatasvrc-postgres-node0-ct-pg_hba
sas-viya-sasdatasvrc-postgres-node0-ct-postgresql
sas-viya-sasdatasvrc-postgres-pgpool0-ct-pcp
sas-viya-sasdatasvrc-postgres-pgpool0-ct-pgpool
sas-viya-sasdatasvrc-postgres-pgpool0-ct-pool_hba
sas-viya-sasdatasvrc-postgres-node0
```

```
sas-viya-cascontroller-default
sas-viya-httpproxy-default
sas-viya-rabbitmq-server-default
sas-viya-sasdatasvrc-postgres
sas-viya-authorization-default
sas-viya-cachelocator-default
sas-viya-configuration-default
sas-viya-identities-default
sas-viya-saslogon-default
avsas-viya-cas-management-default
sas-viya-tenant-default (only in the case of a multi-tenant deployment)
sas-viya-deploymentBackup-default
sas-viya-backup-agent-default
```

Note: The preceding step should be considered only during a restore on a single-tenant deployment or when performing a deployment restore in a multi-tenant environment. Do not stop any service while performing a tenant restore.

- The backup-agent service must be running on all the data sources that need to be restored.
- After performing a restore, stop and restart all services.
 - Stop all services:

```
sudo /etc/init.d/sas-viya-all-services stop
```

- Start all services:

```
sudo /etc/init.d/sas-viya-all-services start
```

Note: Starting and stopping the services is applicable only for a single-tenant restore or a deployment restore in multi-tenant deployment. This is not applicable for a tenant restore.

Backup and Restore: Service Configuration

Backup Configuration Properties



The backup service has its own configuration that is used to perform the backup and restore. The backup configuration includes configuration properties, which come with default values. The *sharedVault* property does not have a default value.

The following configuration properties can be changed:

- *retentionPeriod* —The number of days that backups are stored before they are removed from the shared vault. The backups cannot be recovered once they are deleted.
- *sharedVault* —A shared network location to keep a safe copy of the backup. This directory must exist, and be writeable by the install user (sas) and the user who executes the backup command. In a multi-tenant deployment, the install user must have Read and Write permissions on every machine. It is referred to as the Shared Vault in this document.

Modify the Backup Configuration Using the Environment Manager

The configuration properties *retentionPeriod* and *sharedVault* can be modified using SAS Environment Manager.

- 1 Log on to <protocol>://<host>:<port>/SASEnvironmentManager using administration credentials.
- 2 Click  on the left panel.
- 3 Select **All services** from the **View** drop-down menu.
- 4 Click **Backup service** in the left pane.
- 5 Scroll down until you see **sas.deploymentbackup**, and click  to the right of the service.
- 6 Change the properties as necessary or keep the defaults. However, you must change the *sharedVault* property as there is no default value.
- 7 Click **Save**.

Configure the Backup

After the software installation is complete, you must perform the following steps:

- 1 Set the *sharedVault* location to a network accessible location. This directory must have Read and Write permission to for the install user (sas). See [Modify the Backup Configuration Using the Environment Manager](#) to modify the *sharedVault* property.
- 2 Determine a schedule for the backup.
- 3 Backups can be scheduled using the scheduling service. Determine the frequency and type of the backup to be scheduled. See [Schedule a Backup](#) to create backup schedules as per your requirement.

Note: The local vault location is `/opt/sas/viya/config/backup`. The local vault location is located on machines that have SAS Configuration Server, the SAS Cloud Analytic Services (controller node), SAS Infrastructure Data Server, or SAS Message Broker. The shared vault location must be different from this location.
- 4 Set an appropriate retention period.

Set the *retentionPeriod* value such that you always have at least the last three to four backups at any point in time. For example, if you are doing daily backups, the recommended *retentionPeriod* is 4 days. If you are doing weekly backups, the recommended *retentionPeriod* is 30 days. See [Modify the Backup Configuration Using the Environment Manager](#) to set the *retentionPeriod*.
- 5 Check the groups for the users who are running SAS Viya services and CAS controllers. The backup service uses a 'sas' user to backup and restore data sources from the respective tiers that are included in the backup. For CAS backup, if you are using a different user who is not part of the 'sas' group, then you need to

change some settings before running backup and restore. For more information, see [Backup CAS Access Controls and Caslib Information](#) .

For example, if you have another user named 'cas' in the 'cas' group, make that 'cas' user a member of the 'sas' user group. Also make the 'sas' user a member of 'cas' user group. The following commands are examples of how to make users members of groups:

```
sudo usermod -a -G sas cas
sudo usermod -a -G cas sas
```

After running the commands, restart the following two services:

```
sudo /etc/init.d/sas-viya-backup-agent-default restart
sudo /etc/init.d/sas-viya-cascontroller-default restart
```

Note: During SAS Viya installation, the 'sas' user is created along with 'sas' group.

Make sure that the users running the cas process and sas services have the appropriate memberships. For more information see [Backup Configuration Information](#) .

Backup: How the Process Works

Single-Tenant Deployment

- 1 You invoke the `/backups/jobs` REST API with request of type `application/vnd.sas.backup.request+json`. A job is created in the global history file (`backuphistory.json`). The REST API then discovers the data sources that are to be backed up and creates a backup job with tasks for each of the data sources discovered. The sources are discovered using Consul Service Catalog and CAS Management Service.
- 2 A backup job is created. This object has a field 'ID', which is a unique identifier for the job and can be used to track the status of the backup using the rest endpoints. The service should be polled until the job status is either completed, failed, or completedWithWarning. The backup operation runs in the background.
- 3 Once the backup is complete, the backup files are stored locally on host where the data source resides. The data is stored inside a folder with a timestamp-based backupName (for example, `2016-08-04T05_29_55_910-0400`), tenant ID ('default' in the case of a single-tenant deployment), and data source name (for example, `2016-08-04T05_29_55_910-0400\default\consul\xxxx.dmp`).
- 4 After the backup operation is complete, the status of the backup operation is updated to the history file (`backuphistory_default.json`).
- 5 The backup service then finds the location of the shared vault from the configuration and transfers the locally stored backup files, including the history file, to the shared vault. The backups are stored within the folder named using the backupID. See [How Backups are Stored](#) for more details about the sharedVault directory structure.

Note: The shared vault location must be different from local vault location and should always be accessible to the install user of SAS Viya services.

- 6 After all of the files are transferred from the local vault to the shared vault, the transfer is complete. If backup of any of the data sources fails or the transfer of files fails, the entire backup is considered to have failed .
- 7 Once the transfer is complete, the status of the backup job is updated in the `backuphistory.json` file, and the backup data from the local vault is deleted.

Multi-Tenant Deployment

- 1 You invoke the `/backups/jobs` REST API with request of type `application/vnd.sas.backup.deployment.request+json`. (Optional) You can specify the list of tenants to be backed up. If the tenant list is provided in the backup request, then the backup service validates if each of the tenants received in the request are onboarded. However, if the tenant list is not provided in the backup request, the backup services retrieves all of the onboarded tenants in the system. Once the list of tenants are determined, the backup service creates a job in the global history file (`backuphistory.json`) and invokes the `/backups/jobs` REST API with request `application/vnd.sas.backup.request` for each tenant in the determined list with a common backupID.

Note: A binary type of backup includes all of the onboarded tenants by default. The list of tenants to be backed up is not accepted by the backup service in a binary type of backup.

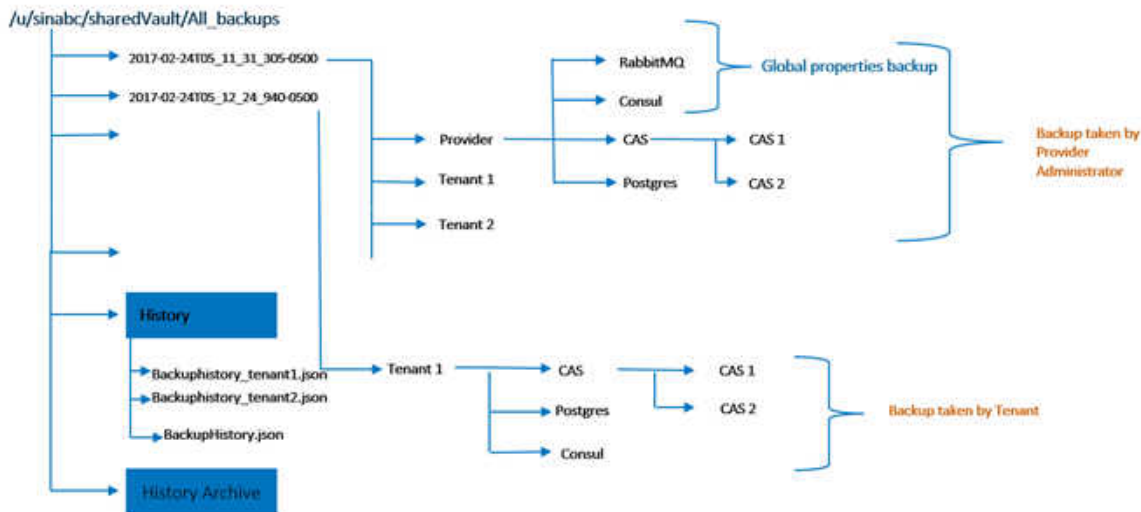
- 2 For each tenant, the backup service discovers the data sources that are to be backed up. The backup service creates a backup job with tasks for each of the data sources discovered. The sources are discovered using Consul Service Catalog and CAS Management Service.
- 3 A backup job is created. This object has the backupID, which was generated during the deployment backup operation. The backupID can be used to track the status of the backup using the rest endpoints. The service should be polled until the job status is either completed, failed, or completedWithWarning. The backup operation runs in the background.
- 4 Once the backup is complete, the backup files are stored locally on the host where the data source resides. The data is stored inside a folder with a timestamp-based backupName (for example, `2016-08-04T05_29_55_910-0400`) and data source name (for example, `2016-08-04T05_29_55_910-0400\acme\consul\xxxx.dmp`).
- 5 After the backup operation is complete, the status of the backup operation is updated to the tenant history file (`backuphistory_<tenantId>.json`).
- 6 The backup service then finds the location of the shared vault from the configuration and transfers the locally stored backup files, including the tenant history file, to the shared vault. The backups are stored within the folder named using the backupID. See [How Backups are Stored](#) for more details about the sharedVault directory structure.

Note: The shared vault location must be different from local vault location and should always be accessible to the install user of SAS Viya services.

- 7 After all of the files are transferred from the local vault to the shared vault, the transfer is complete. If backup of any of the data sources fails or the transfer of files fails, the entire backup is considered to have failed.
- 8 Once the backup operation is complete, an event is triggered. The backup service listens to this event and updates the status of the backup operation for that tenant in the global history file (`backuphistory.json`).
- 9 When the backup operation completion event is received for all the tenants, the deployment job status is updated. If the backup of all tenants is successful, the deployment backup job is marked complete. If at least one tenant backup fails, the deployment backup job is marked completedWithWarning. If all the tenant backups fail, the deployment backup job is marked as failed.

Backups: How They Are Stored

The following diagram explains the directory structure of the shared vault and how the backups are stored within this structure.



In the diagram above, the path `/u/sinabc/shareVault/All_backups` is the path to the shared vault. This directory contains the folders for the backups taken. The folders are named using the date and time at which the backup was performed. Each backup folder contains folders for the tenants included in the backup. In a single-tenant environment, there is only one folder named `__default__`. In a multi-tenant environment, a folder named `tenantId` is available. Within each tenant folder, there are folders for each of the data sources to which the tenant has access. Within each data source folder, you can find the backup files for that data source.

Within the shared vault, there are also folders named `History` and `HistoryArchive`. The `History` folder stores the history files, which includes the global history file (`backuphistory.json`) and tenant history files (`backuphistory_<tenantId>.json`). The `History archive` folder contains the backup of the `History` folder taken after each successful backup or restore operation.

Backup and Restore: Purging

The Backup service retains backups for a period of time that is set by an administrator. The default value for `retentionPeriod` is 30 days. The `retentionPeriod` can be modified through the Backup Configuration using SAS Environment Manager.

The Backup service retains the last successful backup of each type regardless of `retentionPeriod`. If a binary and default type backup is taken for particular deployment, then the last successful backup for type 'default' and the last successful backup for type 'binary' is retained. In a multi-tenant deployment, if a backup was explicitly taken by a tenant after a successful deployment backup (default or binary), those backups are also retained.

Old backups are purged after the retention period and are deleted from the file system. Previous backups for onboarded tenants are purged in the next purge cycle if the backup has passed the retention period. You should not manually delete the data from the shared vault for an offboarded tenant. The restore from this backup would fail because the backup data is not present. In the case of a multi-tenant environment, purging of the backups happens only for tenants with the state "ONBOARDED".

Note: Purging is not performed by the Backup Service for backups of offboarded tenants.

Restore: Perform a Restore

Overview

The restore operation automatically restores content and configuration information to the SAS Configuration Server, the SAS Message Broker, and the SAS Infrastructure Data Server. However, you cannot use the backup service to automatically restore the CAS configuration and to restore the SAS Infrastructure Data server if it is unresponsive. For instructions on manually restoring CAS configurations, see [Restore CAS Server Access Controls and Caslib Information](#) . For instructions on how to restore an unresponsive SAS Infrastructure Data server, see in section . [Restore an Unresponsive SAS Infrastructure Data Server on page 149](#)

The Backup and Restore service has the following caveats:

- Cannot be used to restore a CAS configuration. You must do this manually. See [Restore CAS Server Access Controls and Caslib Information](#) .
- Cannot resolve a problem where the SAS Infrastructure Data server cannot start by redeploying PostgreSQL and then attempting to restore from a backup to that new deployment.
- Cannot perform a restore if the SAS Infrastructure Data Server does not start due to PostgreSQL corruption or for any other reason.
 - If the SAS Infrastructure Data Server does not start and you have configured the SAS Infrastructure Data Server for HA, you must first promote a standby server to primary status before performing a restore with the Backup and Restore service. For more information see [Restore an Unresponsive SAS Infrastructure Data Server](#) .
 - If you have not configured the SAS Infrastructure Data Server for HA, it is recommended that you read and perform the steps provided in the section [Best Practices for Postgres on a Single-node When the SAS Infrastructure Data Server is Not Configured for HA](#) . Then use the binary backup in addition to the Backup and Recovery service.

Prerequisites

To perform a restore, the SAS Infrastructure Data Server must be running and responding to the requests. If you have not configured the SAS Infrastructure Data Server for HA and the server becomes unresponsive and cannot be started because of corruption or for any other reason, you must first use the binary backup utility to restore. For more information see [Best Practices for Postgres on a Single-node When the SAS Infrastructure Data Server is Not Configured for HA](#) .

How the Restore Process Works

Tenant in a Multi-Tenant Deployment and Single-Tenant Deployment

In the case of a tenant in a multi-tenant deployment or restoring a single-tenant deployment:

Note: Only in the case of a single-tenant deployment when restoring using the binary backup, ensure that you restore the SAS Infrastructure Data server manually. For more information, see [Restore an Unresponsive SAS Infrastructure Data Server](#) . If there are other data sources included in the binary backup, start the restore using backup service after manual restore of SAS Infrastructure Data server to restore those data sources.

- 1 Before starting the restore, you need to check whether the shared vault location is pointing to the location where the backup resides.

2 During the restore, the following services must be running:

```

sas-viya-consul-default
sas-viya-vault-default
sas-viya-sasdatasvrc-postgres-node0-ct-pg_hba
sas-viya-sasdatasvrc-postgres-node0-ct-postgresql
sas-viya-sasdatasvrc-postgres-pgpool0-ct-pcp
sas-viya-sasdatasvrc-postgres-pgpool0-ct-pgpool
sas-viya-sasdatasvrc-postgres-pgpool0-ct-pool_hba
sas-viya-sasdatasvrc-postgres-node0
sas-viya-cascontroller-default
sas-viya-httpproxy-default
sas-viya-rabbitmq-server-default
sas-viya-sasdatasvrc-postgres
sas-viya-authorization-default
sas-viya-cachelocator-default
sas-viya-configuration-default
sas-viya-identities-default
sas-viya-saslogon-default
sas-viya-cas-management-default
sas-viya-tenant-default (only in the case of a multi-tenant deployment)
sas-viya-deploymentBackup-default
sas-viya-backup-agent-default

```

Note: The backup-agent service must be running on all the data sources that need to be restored.

Stop all SAS services except those mentioned above and ensure that the processes have been stopped before initiating a restore. Manually stop any SAS processes that are still running.

- 3 While initiating the restore, invoke the /restores/jobs endpoint along with the name of the backup (value in the backupName field) in the restore request of type 'application/vnd.sas.restore.request'. The backup service receives the request and creates a job in the global history file.
- 4 When a restore is initiated, the backup service performs the pre-restore validations. For more information about the pre-restore validations see [Backup and Restore Terms](#) in the "Backup and Restore: Terms and Concepts" section.
If all the checks pass the test, the restore is initiated.
- 5 The backup service sends a message to each of the data sources to download the backup files from the shared vault to local vault for the restore.
- 6 Once files are downloaded to the local vault, the restore is started.
- 7 Once the restore is completed, all of the data sources send the message of completion back to back up service.
- 8 The backup service then updates the status of the restore job depending on the status of the restore of data sources.
- 9 Once the restore has successfully completed, stop all SAS services and ensure that the processes have been stopped. Manually stop any SAS processes that are still running.
- 10 Restore the CAS server manually. For more information, see [Restore CAS Server Access Controls and Caslib Information](#) .
- 11 Restart all services.

Note: When restoring a tenant in a multi-tenant deployment, the steps are the same. However, the SAS services mentioned in step two should not be stopped and no service is required to be restarted once the restore is completed. When a tenant is being restored, only the configuration for that tenant is restored and Postgres is

restored only if the backup type is default. The CAS server should be restored manually only for the tenant being restored.

Multi-Tenant Deployment by Provider Administrator

In the case of a multi-tenant deployment by a Provider administrator where all tenants or selected tenants are backed up:

Note: When restoring using the binary backup, ensure that you restore the SAS Infrastructure Data server manually. For more information see [Restore an Unresponsive SAS Infrastructure Data Server](#) . If there are other data sources included in the binary backup, start the restore using the backup service after manual restore of SAS Infrastructure Data server to restore those data sources.

- 1 Before starting the restore, ensure that the shared vault location is pointing to the location where the backup resides.
- 2 During the restore, the following services must be running:

```
sas-viya-consul-default
sas-viya-vault-default
sas-viya-sasdatasvrc-postgres-node0-ct-pg_hba
sas-viya-sasdatasvrc-postgres-node0-ct-postgresql
sas-viya-sasdatasvrc-postgres-pgpool0-ct-pcp
sas-viya-sasdatasvrc-postgres-pgpool0-ct-pgpool
sas-viya-sasdatasvrc-postgres-pgpool0-ct-pool_hba
sas-viya-sasdatasvrc-postgres-node0
sas-viya-cascontroller-default
sas-viya-httpproxy-default
sas-viya-rabbitmq-server-default
sas-viya-sasdatasvrc-postgres
sas-viya-authorization-default
sas-viya-cachelocator-default
sas-viya-configuration-default
sas-viya-identities-default
sas-viya-saslogon-default
sas-viya-cas-management-default
sas-viya-tenant-default
sas-viya-deploymentBackup-default
sas-viya-backup-agent-default
```

Note: The backup-agent service must be running on all the data sources that need to be restored.

Stop all SAS services except those mentioned above, and ensure that the processes have been stopped before initiating a restore. Manually stop any SAS processes that are still running.

- 3 Invoke `/restores/jobs` with the backupName to be used for the restore in the request of type 'application/vnd.sas.restore.deployment.request+json'. Optional, one can specify the list of tenants to be restored and those tenants should be part of specified backup.
- 4 The restore request accepts the list of tenants to be restored. If this list is not provided, the backup service creates this list with the tenants that are onboarded along with the Provided Administrator. If any of the tenants on the list are not present in the specified backup, then the restore operation does not proceed and is marked as failed. However, if all the tenants on the list are present in the specified backup, then the restore job is created in the global history file. The `/restores/jobs` API is invoked for each requested tenant with the request 'application/vnd.sas.restore.request+json'.
- 5 The backup service receives the restore request and pre-restore validations and checks to see whether the provided backupName is valid. For more information about the pre-restore validations, see [Backup and Restore Terms](#) in the "Backup and Restore: Terms and Concepts" section.

- 6 If the backupName is valid, the backup service retrieves the data sources from the backup and initiates the restore tasks for each of the data sources and sends it to the backup agent.
- 7 The backup agent performs the restore and sends the appropriate result back to the backup service.
- 8 The backup service consolidates the results of the restore for all the data sources and marks the restore job for that tenant appropriately. The backup service sends an event about the restore status for this tenant.
- 9 When the service receives the responses for each of the tenants, it marks the status of the restore job in the global history file. If all tenants send a success event, the job is marked completed. If any tenant restore is unsuccessful, the job is marked as failed.
- 10 Once the restore has completed successfully, stop all SAS services and ensure that the processes have been stopped. Manually stop any SAS processes that are still running.
- 11 Manually restore the CAS server for each tenant. For more information, see [Restore CAS Server Access Controls and Caslib Information](#).
- 12 Restart all services.

Restore an Unresponsive SAS Infrastructure Data Server

If you have not configured the data server for HA, and the data server does not start because of PostgreSQL corruption or for any other reason, perform the following steps:

- 1 Stop all services including the database service.

```
sudo /etc/init.d/sas-viya-all-services stop
```

- 2 Archive or rename the existing node0 directory.

```
cd /opt/sas/viya/config/data/sasdatasvrc/postgres
mv node0 node0_original
```

- 3 Create a node0 directory with permissions and ownership similar to the old node0 directory.

```
mkdir node0
chmod 700 node0
cd node0
```

- 4 Extract the contents of the base.tar.gz file into the node0 directory.

- In a multi-tenant environment:

```
tar -xvf ../sharedVault/<backup-Id>/provider/postgres/base.tar.gz
```

- In a single-tenant environment:

```
tar -xvf ../sharedVault/<backup-Id>/__default__/postgres/base.tar.gz
```

Once the files are extracted, delete the base.tar.gz file from the node0 directory.

- 5 Ensure that the hot_standby property is set to off in the postgresql.conf file. The postgresql.conf file can be found in the `/opt/sas/viya/config/data/sasdatasvrc/postgres/node0` directory.

```
echo "hot_standby = off" >> postgresql.conf
```

- 6 Remove the recovery.conf and recovery.done files located at `/opt/sas/viya/config/data/sasdatasvrc/postgres/node0`.

- 7 Start the Consul service and then start the database service.

```
sudo /etc/init.d/sas-viya-vault-default start
```

```
sudo /etc/init.d/sas-viya-sasdatasvrc-postgres start
```

- 8 Make sure that the database service started successfully without any issues.

```
sudo/etc/init.d/sas-viya-sasdatasvrc-postgres status
```

Verify that the primary data server has started without any issues and has a status of “up”. Here is an example:

```
sudo /etc/init.d/sas-viya-sasdatasvrc-postgres status
Checking status of sas-viya-sasdatasvrc-postgres...
```

```
PGPool is running with PID=4733
```

```
PGPool is running with PID=32413
Checking Postgresql nodes status...
 node_id | hostname | port | status | lb_weight | role | select_cnt | load_balance_node |
 replication_delay
-----+-----+-----+-----+-----+-----+-----+-----+-----
 0 | myhost.domain.com | 5432 | up | 1.000000 | primary | 0 | true | 0
(1 row)
```

- 9 Set the `hot_standby` property in the `psql.conf` file back to `on`.

```
echo "hot_standby = on" >> postgresql.conf
```

- 10 Restart all services.

Backup and Restore for Programming-Only Deployments

How To

Back up CAS Access Controls and Caslib Information

CAUTION! It is strongly recommended that you back up each CAS server’s stored access control and caslib information after adding global-scope caslibs or setting access controls. Backups are particularly important after you modify access controls or add, delete, or modify global caslibs.

You can perform a backup programmatically by using the `createBackup` and `completeBackup` actions.

Note: You need to be an administrator to perform a backup.

- 1 To perform the backup, run the following code in SAS Studio replacing the `path` location to your location:

```
cas casauto host="cloud.example.com" port=5570;

proc cas;
accessControl.assumeRole /
    adminRole="SuperUser";
accessControl.createBackup /
    path="/my/backup/location";
accessControl.completeBackup;
accessControl.dropRole / adminRole="SuperUser";
quit;
```

- 2 Copy the backup location directory to a location where it can be saved. The cas user needs Write access to the location provided. If the location does not exist and the cas user has Write access, the location is created.

If you do not specify `path=" "`, the backup location is the directory named `backup`. This directory is in the `permstore` option location. It is under the directory named for the fully qualified DNS name of the machine that runs the main controller. The cas user must have Read and Write access to both the `permstore` and `backup` directories. For more information about `permstore`, see [Configuration File Options Reference](#) in the “SAS Cloud Analytic Services: Reference” section of the *SAS Viya Administration: SAS Cloud Analytic Services*.

Back up Configuration Information

Here are the located hosts that have been listed in the `[sas-casserver-primary]` host group in the inventory file: machine:

```
/opt/sas/viya/config/etc/cas/default/casconfig.lua
/opt/sas/viya/config/etc/cas/default/cas.hosts
```

Here are the located hosts that have been listed in the `[programming]` host group in the inventory file:

```
/opt/sas/viya/config/etc/sasstudio/default/init_usermods.properties
/opt/sas/viya/config/etc/sasstudio/default/appserver_usermods.sh
/opt/sas/viya/config/etc/spawner/default/spawner_usermods.sh
/opt/sas/viya/config/etc/workspaceserver/default/autoexec_usermods.sas
/opt/sas/viya/config/etc/workspaceserver/default/sasv9_usermods.cfg
/opt/sas/viya/config/etc/workspaceserver/default/workspaceserver_usermods.sh
```

If your site has created global folder shortcuts for SAS Studio, you should back up the directory that contains the shortcuts. By default, the shortcuts are stored in the following directory:

```
/opt/sas/viya/home/SASFoundation/GlobalStudioSettings
```

Note: Your site might have configured a different directory for the shortcuts. For details, see [Configuring Global Folder Shortcuts](#) in the Configuration Properties: How to Configure SAS Studio topic in the *SAS Viya Administration: Configuration Properties*.

Restore CAS Server Access Controls and Caslib Information

- 1 Stop the server. See [Start and Stop Cloud Analytic Services](#) in *SAS Viya Administration: Servers*.
- 2 The `permstore` directory and its contents at `/opt/sas/viya/config/etc/cas/default` should be replaced with the `permstore` directory that is located in the `sharedVault` that you specified when configuring your backup. The `permstore` directory is under the `<backupId>/cas-shared-default` folder. The cas user needs Read and Write access to the `permstore` directory.

If you specified a location in the `path=" "` option when creating a backup, then that is the content that you should restore.

- 3 Restart the server. See [Start and Stop Cloud Analytic Services](#) in *SAS Viya Administration: Servers*.

Restore the Most Recent Permstore in the Event of a Failover

If no global-scope caslibs were added or changes to access controls were made after the failover, then you can restore from a backup as described in [Restore CAS Server Access Controls and Caslib Information](#)

If changes were made or you are unsure, then perform the following steps to restore the latest permstore from the backup controller.

CAUTION! The backup controller removes its permission store at start-up in order to begin in a synchronized state with the primary controller. It is vital that you preserve the permstore from the backup controller before restarting the server in order to have access to the most up-to-date permission store.

- 1 Stop the server. See [Start and Stop Cloud Analytic Services](#) in *SAS Viya Administration: Servers*.
- 2 Find the location of the primary controller's permission store. The default location is: `/opt/sas/viya/config/etc/cas/default/permstore/<fully-qualified-domain-name>`.
- 3 Remove the contents of the directory.
- 4 On the backup controller host, find the location of the backup controller's permission store. The default location is: `/opt/sas/viya/config/etc/cas/default/permstore/<fully-qualified-domain-name>`.
- 5 On the backup controller host, make a copy of the permstore. Make sure that you preserve the file ownership and permissions. One way to do this is with the tar command.

```
sudo tar cf /root/backup_controller_permstore.tar
```

- 6 Copy the archive to the host for the primary controller.
- 7 On the primary controller host, change directory to the location of the now empty permstore.

```
sudo tar xf /path/to/backup_controller_permstore.tar
```

- 8 Start the server.
 - If the primary controller host is stopped, then boot the machine. When the machine starts, the server is automatically started.
 - If the primary controller host is running, then start the server with the following command:

```
sudo /etc/init.d/sas-viya-cascontroller-default start
```

For more information about stopping and starting CAS see [Operate](#) in the SAS Cloud Analytics Services: How To (Scripts) topic.

Backup and Restore: Backup Manager


What Is the Backup Manager

The Backup Manager is a graphical interface in which you can manage the backup and restore processes for your systems. The SAS Backup Manager is available as a plug-in within the SAS Environment Manager. You can use the SAS Backup Manager for the following tasks:

- viewing the backup and restore history
- viewing details about backup and restore
- viewing the backup configuration
- running an immediate (ad hoc) backup
- restoring a backup

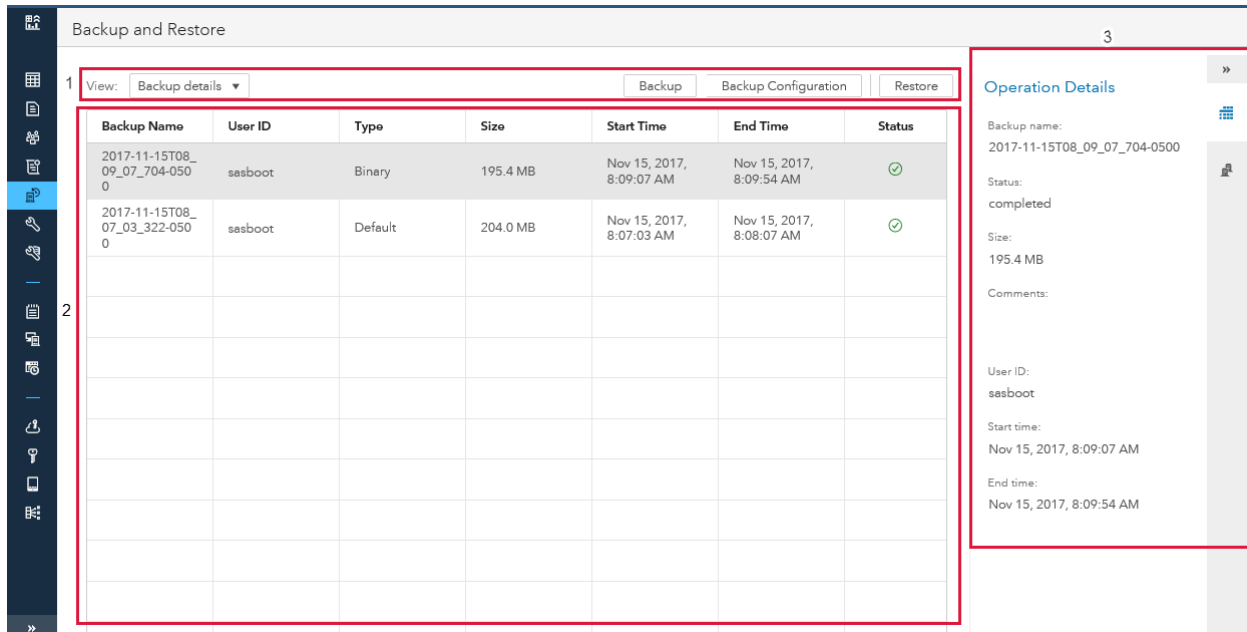
To view the SAS Backup Manager from the Environment Manager:

- 1 Log on to the SAS Environment Manager.

2 In the left menu panel, click .

First Look at the Backup Manager

The Backup Manager provides a way to back up and restore your system using a graphical interface. The features of the SAS Backup Manager window are as follows:



Backup Name	User ID	Type	Size	Start Time	End Time	Status
2017-11-15T08_09_07_704-0500	sasboot	Binary	195.4 MB	Nov 15, 2017, 8:09:07 AM	Nov 15, 2017, 8:09:54 AM	✓
2017-11-15T08_07_03_322-0500	sasboot	Default	204.0 MB	Nov 15, 2017, 8:07:03 AM	Nov 15, 2017, 8:08:07 AM	✓

- 1 The toolbar enables you to view backup or restore details, perform an ad hoc backup or restore, and view the backup configuration
- 2 The backup and restore history view displays all of the completed backups or restores. Highlighting a row displays details about that backup or restore.
- 3 The backup and restore details view enables you to view operational and data source details about the backup or restore in the **Operation Details** pane.

View Backup and Restore History

To view a list of backups or restores that are complete, are currently running, or are waiting to run:

- From the **View** drop-down list on the toolbar, select **Backup details** or **Restore details**.

The list includes all backups or restores recorded in the backup history. This includes backups that have been purged due to the retention policy. It also includes backups or restores currently running or that are waiting to run. By default, they are listed in descending order by **Start Time**. The following information is listed in each table:

Backup Name or Restore ID

the unique identifier of the backup or restore, based on the date and time that the backup or restore started (for example, 2017-10-28T05_33_47_326-0400).

User ID

the user ID of the user that ran the backup or restore or the identity name of the service that initiated the backup or restore.

Type

the type of backup – Binary or Default.

Size

the total size of the files that were backed up (not available for a provider in a multi-tenant environment). This column is not displayed when you select **Restore details**.

Start Time








the date and time that the backup or restore started running.

End Time

the date and time that the backup or restore stopped running.

Status

contains one of the following icons, indicating the status of the backup or restore operation:

Icon	Description
	The backup or restore has not yet started.
	The backup or restore is currently running (in progress).
	The backup or restore completed without errors or warnings.
	The backup or restore completed with warnings.
	The backup or restore completed with errors.
	The backup was purged.
	The status of the backup or restore cannot be determined.


Backup Name	User ID	Type	Size	Start Time	End Time	Status
2017-11-15T08_09_07_704-050_0	sasboot	Binary	195.4 MB	Nov 15, 2017, 8:09:07 AM	Nov 15, 2017, 8:09:54 AM	
2017-11-15T08_07_03_322-050_0	sasboot	Default	204.0 MB	Nov 15, 2017, 8:07:03 AM	Nov 15, 2017, 8:08:07 AM	

TIP You can refresh your browser to see the latest status.

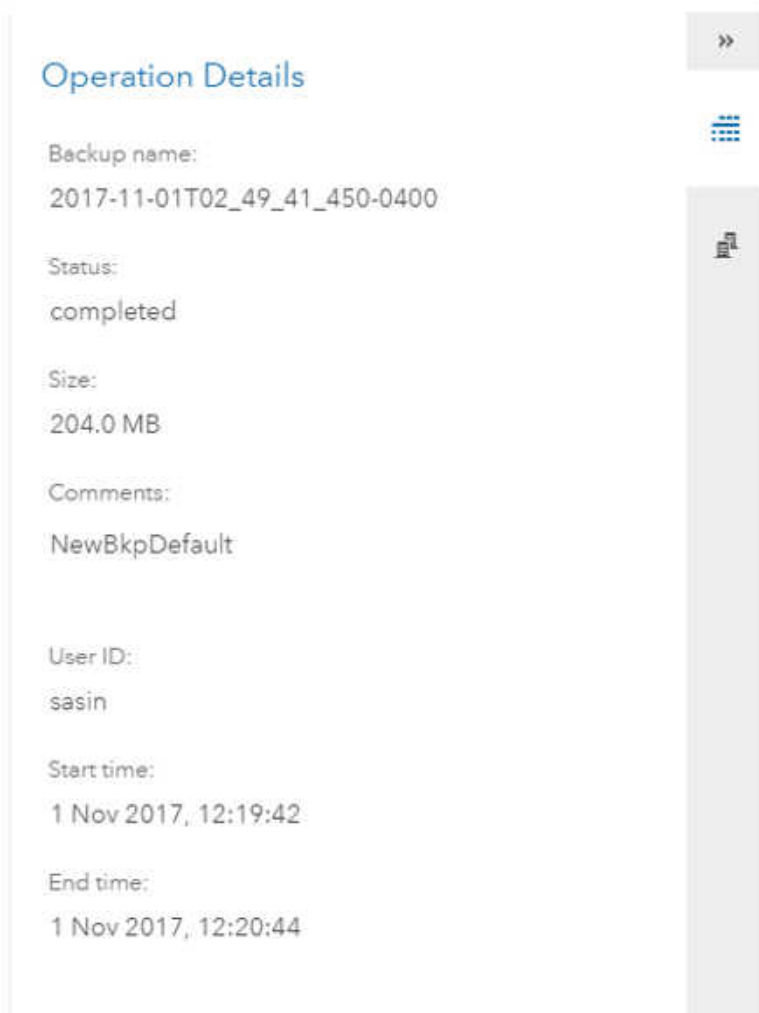
To use CLI commands to view the backup and restore history, see [Backup and Restore: Command Line Interface on page 161](#)

View Backup or Restore Details in a Single-tenant Setup or Tenant in a Multi-tenant Setup

View Backup or Restore Operation Details

To view details for a particular backup or restore operation, click  in the right pane. A panel slides out that displays all of the details for that operation. The following details are displayed:

- Backup Name or Restore ID.
- Status of the backup or restore job.
- Total size of the files that were backed up. This information does not appear for restores.
- Any comments that were specified when the backup or restore was run.
- User ID of the user that ran the backup or restore or the identity name of the service that initiated the backup or restore.
- Start and end date and time for the backup or restore.



Operation Details

Backup name:
2017-11-01T02_49_41_450-0400

Status:
completed

Size:
204.0 MB

Comments:
NewBkpDefault


User ID:
sasin

Start time:
1 Nov 2017, 12:19:42

End time:
1 Nov 2017, 12:20:44

View Backup or Restore Data Sources

View the List of Data Sources


The data sources for the currently selected backup or restore are listed in the right pane. To view the list of data sources, click . If you are viewing details for a restore, only the data sources that were restored are listed.

By default, the backup data sources include the following:


- SAS Message Broker (not available to a tenant in a multi-tenant setup)
- SAS Configuration Server
- SAS Cloud Analytic Services
- SAS Infrastructure Data Server

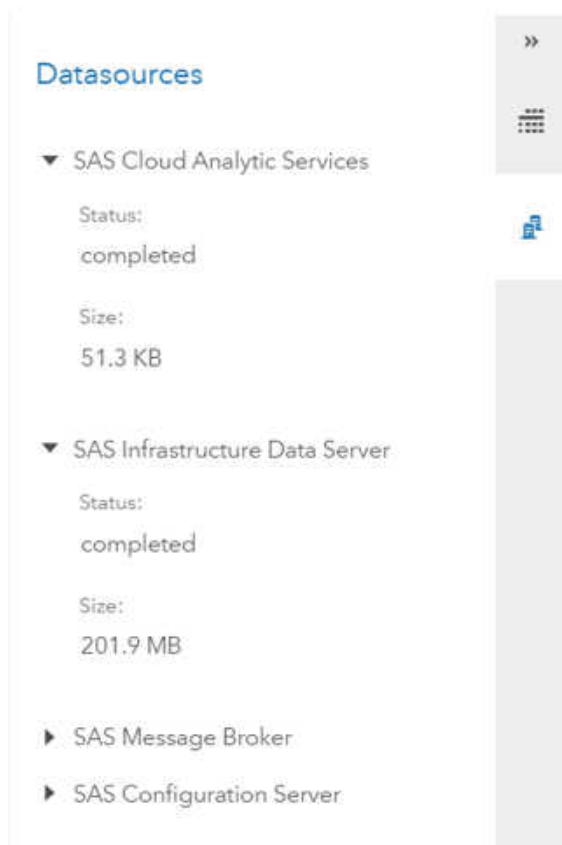
For more information about the data sources, see [About the Backup and Restore Service](#).

View Details about a Data Source

To view details about a particular backup or restore data source, click  to the left of the source name. The following details are displayed:

- the status of the data source's backup or restore.
- the total size of the backup files for this data source. This information does not appear for restores.

To collapse the data source so that the details are not displayed, click  to the left of the data source name.



Datasources

- ▼ SAS Cloud Analytic Services
 - Status: completed
 - Size: 51.3 KB
- ▼ SAS Infrastructure Data Server
 - Status: completed
 - Size: 201.9 MB
- ▶ SAS Message Broker
- ▶ SAS Configuration Server

View Backup or Restore Details for a Provider in a Multi-tenant Setup

To view details for a particular backup or restore operation, select the operation by double-clicking any of its columns or right-clicking and selecting **Tenants** from the pop-up menu. A listing of the tenants that were backed up or restored is displayed. The following details appear for each tenant:

- the tenant name.
- the start and end date and time for the backup or restore for the tenant.
- the status of the backup or restore for the tenant.

View the Backup Configuration

To view the current backup configuration, click **Backup Configuration**. The Backup Configuration dialog box displays the following information:

- **Retention period**—number of days that backups are stored before they are removed from the shared vault.
- **Shared vault**—location where all backups are stored. In a multi-machine deployment, the install user must have Read and Write permissions on every machine.
- **Additional databases**—additional Postgres databases, which need to be backed up.

Backup Configuration

Retention period:

30

The number of days that backups are stored before they are removed from the backup vault

Shared Vault:

/opt/sas/viys/sharedVault

The location where all backups are stored. In a multi-machine deployment, the install user must have Read and Write permissions on every machine.

Additional databases:


(not set)

Additional Postgres databases which need to be backed up.

For more information see [Configuring the Backup](#)

Run an Unscheduled Backup

To start an immediate backup:

- 1 Log on to the SAS Environment Manager.
- 2 In the left menu panel, click .

- 3 Click **Backup** in the upper right corner of the Backup and Restore window.
- 4 In the Backup dialog box, enter the following:
 - **Comments**—optional free-form comment describing the backup. The comment is recorded in backup history and is displayed in the backup's **Operation Details**.
 - **Backup type**—select a backup type. There are two types of backups:
 - Default
 - Binary

For more information about the backup types, see [Backup and Restore Terms](#)

The **Default** type is selected by default.

For the **Binary** type, select the **Include all sources** check box, if other sources also need to be backed up.

Backup

Comments:

Backup type:

- Binary
 - Include all sources
- Default

- Tenants**—If the user is a provider in a multi-tenant setup, the user can select the tenants to be backed up from the onboarded tenant list.

Backup

Comments:

Backup type:

Binary

Include all sources

Default

Select All

- provider
- acme
- cyberdyne

Note: A binary backup includes all of the onboarded tenants by default.

If the user is an intra-tenant administrator in a multi-tenant environment or an administrator in a single-tenant environment, the tenant list is not displayed.

For an intra-tenant administrator, the backup type is not displayed since the intra-tenant administrator is allowed to perform only a default backup.

- 5 Click **Backup** to start the backup.

A new row is added in the table with the status running.

The page refreshes until the backup process is complete.



After the process is complete, a message is displayed letting you know whether the backup was completed, completed with warnings, or failed.

Note: To see the status of the backup on the main page, refresh your browser.

Restore a Backup

- 1 Select **Backup details** from the **Views** drop-down list.
- 2 Select the backup to be restored.

If the backup type is **Binary**, ensure that you restore the [SAS Infrastructure Data Server](#) before initiating the restore using the user interface

- In the case of a multi-tenant environment where the Provider Administrator is initiating the restore operation, the status of the backup must either be Completed () or Completed with Warnings ().

If the backup has a status of 'Completed with Warnings', the Provider Administrator needs to select those tenants only with a status of Completed from specified backup and initiate restore operation.

- In the case of a multi-tenant environment, if the intra-tenant Administrator is initiating a restore operation, the status of the backup must be Completed.
- In the case of a single-tenant environment, the backup status must be Completed before a restore operation can be initiated.

If a Binary backup had other sources included, only those sources can be restored using the user interface. The SAS Cloud Analytic Services must be manually restored. For more information, see [Restore CAS Server Access Controls and Caslib Information on page 151](#).

- 3 The **Restore** dialog box displays the selected Backup name.
- 4 The following details can be entered in the **Restore** dialog box:
 - **Comments**—optional free-form comment describing the backup. The comment is recorded in restore history and is displayed in the restore's **Operation Details**.
 - **Force restore if some databases don't match**—Select this check box if you want to force a restore if the databases do not match.

Restore

Backup ID:

2017-11-15T08_07_03_322-0500

Comments:

Force restore if some databases don't match

- **Tenants**—if the user is a provider in a multi-tenant setup, the user can select the tenants to be restored from the tenant list. The tenant list displays the onboarded tenants present in the specified backup whose backups are "completed" or "completedWithWarnings".

Restore

Backup ID:

2017-10-30T03_00_02_851-0400

Comments:

Force restore if some databases don't match

Select All

- provider
- cyberdyne
- acme

Note: If the user is an intra-tenant administrator in a multi-tenant environment or an administrator in a single-tenant environment, the tenant list is not displayed.

- 5 Click **Restore** to start the restore process. The UI displays a message `Restore In Progress`.

It is recommended not to use the UI while a restore is in progress. As the datasources are being refreshed, the UI might not display all results.

- 6 After the process is complete, a message is displayed letting you know whether the restore was completed, completed with warnings, or failed.

For any failed backup or restore operation, the Provider Administrator can check the logs at `/opt/sas/viya/config/var/log/deploymentbackup/default` and `/opt/sas/viya/config/var/log/backup-agent/default`. If a failure occurs in any operation initiated by an intra-tenant Administrator, the administrator should contact the Provided Administrator to get a reason for the failure.

- 7 To view the restore, select **Restore details** in the **Views** drop-down list. The latest restore operation is now the first row in the table.

Backup and Restore: Command-Line Interface

Overview

The Backup microservice has two command-line interface (CLI) utilities that are available in SAS Viya. These CLIs are developed using the SAS Viya CLI framework and can be used as independent utilities or as plug-ins to the `sas-admin` CLI. For details about the CLI framework, see the SAS CLI Framework documentation. For more information about CLI, see [Overview of the Command-Line Interface](#)

Note: The CLI does not support tenant-specific backup or restore operations.

The following sections show how to use these CLIs to perform backup and restore operations.

SAS Admin CLI

The CLI commands are located at `/opt/sas/viya/home/bin`. Navigate to that directory and initialize the profile and authenticate against the host.

- 1 **Create a profile:** Specify the URL of the host where the backup service is hosted. Also, specify an output type.

```
Command: sas-admin --profile "hostname.mycompany.com " profile init
Enter configuration options:
```

```
Service Endpoint> http://hostname.mycompany.com
```

```
Output type (text|json|fulljson)> text
```

```
Enable ANSI colored output (y/n)?> y
```

```
Saved 'hostname.mycompany.com ' profile to
```

```
/r/sanyo.unx.sas.com/vol/vol101/u101/sretest/.sas/config.json.
```

- 2 **Authenticate:** Specify user name and password.

```
Command: sas-admin --profile "hostname.mycompany.com" auth login
```

```
Enter credentials for http://hostname.mycompany.com :
```

```
Userid> sasboot
```

```
Password>
```

```
Login succeeded. Token saved
```

On a multi-tenant setup, the provider credentials should be provided for deployment level environment operations.

On a multi-tenant setup, tenant credentials should be provided for a tenant level environment operation.

On a single-tenant system, the administrator credentials should be provided.

Once the profile is created and the user is authenticated, that user can use other commands.

3 Set a default profile: sets "hostname.mycompany.com" as default profile.

```
Command export SAS_CLI_PROFILE="hostname.mycompany.com"
```

SAS Backup CLI

Overview

There are three backup CLI commands specific to the backup service:

- start
- list
- show

There are many other commands, which come with the CLI framework, and those can also be used as required. For example, the `-help` command can be used with any plug-in or command to get related help.

```
Command: -help
```

Example:

```
sas-admin backup -help
```

Output:

```
NAME:
  sas-backup - SAS Backup Command Line Interface

USAGE:
  sas-admin backup command [command options] [arguments...]

VERSION:
  1.2.3

COMMANDS:
  authenticate, auth, authn    Handle authentication process
  help, h                      Show a list of commands or help for one command
  list                         Lists backups
  show                          Shows backup details
  start                         Start backup
```

Backup CLI Commands

Start Backup Help

Show backup CLI start command help:

```
start
```

Example:

```
sas-admin backup start -help
```

Output:

NAME:

```
sas-admin backup start
```

USAGE:

```
sas-admin backup start [command options] [arguments...]
```

OPTIONS:

<code>--backup-type, -t "default"</code>	Specifies the type of backup to be taken. The default value is: "default". The valid values for the "backup-type" option are as follows: "default", "binary"
<code>--comments, -c "default comment"</code>	Specifies free text comments that are associated with this backup operation. The default value is: "default comment"
<code>--configuration-id, -n "default"</code>	Specifies the ID of a backup configuration to be used for a backup operation. Only the default configuration is supported. The default value is: "default"
<code>--include-all-sources-for-binary-backup, -i</code>	Specifies whether to include other sources in the backup when the "backup-type" option is specified as "binary". If the "true" value is specified, then all sources are included. The default value is: "false"
<code>--slug, -s "default slug"</code>	Specifies the name that is given to a backup operation. The default value is as follows: "default slug"
<code>--version, -v "1"</code>	Specifies the version of the media type. The default value is 1.

Start Backup**Command:**

```
start
```

Example:

```
sas-admin backup start --comments="default comments" --configuration-id="default"
--slug="default slug" --version=1
```

Output for a multi-tenant system:

```
Start Deployment backup
JobId          2017-10-28T05_56_24_380-0400
Version        1
BackupType     default
State          running
Owner          sasboot
StartTimeStamp 2017-10-28T09:56:24.385Z
EndTimeStamp
tenants        [intech provider acme cyberdyne]
```

Output for a single-tenant system or when the backup is initiated by an intra-tenant administrator on a multi-tenant system:

```
sas-backup-cli start
Version        1
JobId          2017-10-28T11_06_51_561-0400
BackupType     default
IncludeAllSourcesForBinaryBackup false
Slug           default slug
Comments       default comment
State          pending
Owner          sasin
StartTimeStamp 2017-10-28T15:06:51.561Z
EndTimeStamp
```

```
Purged                false
```

To start a default backup with comment:

```
sas-admin backup start -c="sample comment"
```

OR

```
sas-admin backup start --backup-type=default -c="sample comment"
```

To start a binary backup to include only the Postgres data source, with a comment:

```
sas-admin backup start --backup-type=binary -c="sample comment"
```

OR

```
sas-admin backup start --backup-type=binary -c="sample --includeAllSourcesForBinaryBackup=false"
```

To start a binary backup to include all other sources in addition to the Postgres data source, with a comment:

```
sas-admin backup start --backup-type=binary
--includeAllSourcesForBinaryBackup=true -c="sample comment"
```

On a multi-tenant system when the provider is logged in, this command takes a backup of all on boarded tenants.

On a multi-tenant system when the intra-tenant administrator is logged in, this command takes a backup of that tenant.

On a single-tenant system, this command takes a backup of that system.

Get Backup List Help

Command:

```
list --help
```

Output:

NAME:

```
sas-admin backup list - Lists backups.
```

USAGE:

```
sas-admin backup list [command options] [arguments...]
```

OPTIONS:

```
--limit, -l "10"    Specifies the maximum number of backup jobs to return. The default value is 10.
--start, -s "0"     Specifies the 0-based offset of the first backup to return. The default value
                    is 0 for the first one.
```

Show Backup List

Example:

```
sas-admin backup list
```

Output for a multi-tenant system. This is a sample and not all output columns are represented.

BackupId	Version	BackupType	State	Owner	StartTimeStamp
2017-10-28T05_56_24_380-0400	1	default	running	sasboot	2017-10-28T09:56:24.385Z
2017-10-28T05_47_56_781-0400	1	default	completedWithWarning	sasboot	2017-10-28T09:47:56.796Z
2017-10-24T05_35_31_252-0400	1	default	completedWithWarning	sasboot	2017-10-24T09:35:31.259Z
2017-10-24T04_50_51_569-0400	1	default	failed	sasboot	2017-10-24T08:50:51.575Z
2017-10-23T06_37_01_017-0400	1	binary	completed	sasboot	2017-10-23T10:37:01.020Z
2017-10-23T06_30_23_033-0400	1	default	completed	sasboot	2017-10-23T10:30:23.037Z
2017-10-23T06_09_39_355-0400	1	default	completed	sasboot	2017-10-23T10:09:39.361Z
2017-10-23T03_40_16_240-0400	1	default	completedWithWarning	sasboot	2017-10-23T07:40:16.249Z

```
2017-10-23T00_53_09_691-0400 1 default completedWithWarning sasboot 2017-10-23T04:53:09.697Z
2017-10-17T09_13_04_082-0400 1 default completed sasboot 2017-10-17T13:13:04.087Z
```

For a single-tenant setup or when initiated by an intra-tenant administrator on a multi-tenant setup. This is a sample and not all output columns are represented.

Version	BackupId	BackupType	IncludeAllSourcesForBinaryBackup	Slug	Comments
1	2017-10-28T11_06_51_561-0400	default	false	default slug	default comment
1	2017-10-28T09_04_37_140-0400	default	false	new-Slug	new backup
1	2017-10-27T14_38_26_579-0400	default	false	default slug	default comment
1	2017-10-27T14_29_00_225-0400	default	false	default slug	default comment
1	2017-10-27T13_44_03_524-0400	default	false	default slug	default comment
1	2017-10-27T13_42_21_700-0400	default	false	default slug	default comment
1	2017-10-27T13_36_15_236-0400	default	false	default slug	default comment
1	2017-10-26T02_14_03_814-0400	default	false	default slug	default comment
1	2017-10-25T07_45_52_386-0400	default	false	default slug	default comment

Show all backups sorted in time order, newest to oldest.

On a multi-tenant system, the provider administrator sees all deployment backups. The intra-tenant administrator sees all available tenant backups for that tenant.

On a single-tenant system, the administrator sees all available backups on that system.

Example: See the most recent 10 records from the backup history:

```
sas-admin backup list (Uses defaults)
```

OR

```
sas-admin backup list -s=0 -l=10
```

OR

```
sas-admin backup list --start=0 -- limit=10
```

Show Backup Details Help

Command:

```
show -help
```

Example:

```
sas-admin backup show -help
```

Output:

NAME:

```
sas-admin backup show - Shows backup details.
```

USAGE:

```
sas-admin backup show [command options] [arguments...]
```

OPTIONS:

```
--id, -i Specifies the unique ID of the backup operation for which details have been requested.
```

Show Backup Details

Example:

```
sas-admin backup show -i=2017-10-28T05_56_24_380-0400
```

Output for a multi-tenant system. This is a sample and not all output columns are represented.

```
Version 1
```

```

BackupId      2017-10-28T05_56_24_380-0400
BackupType    default
Slug
Name
State         running
Owner         sasboot
StartTimeStamp 2017-10-28T09:56:24.385Z
EndTimeStamp

```

Tenant Backups:

BackupId	Tenant	StartTimeStamp	EndTimeStamp	State
2017-10-28T05_56_24_380-0400	intech	2017-10-28T09:56:25.416Z	2017-10-28T09:56:48.074Z	completed
2017-10-28T05_56_24_380-0400	provider	2017-10-28T09:56:24.385Z		running
2017-10-28T05_56_24_380-0400	acme	2017-10-28T09:56:25.047Z	2017-10-28T09:56:47.043Z	completed
2017-10-28T05_56_24_380-0400	cyberdyne	2017-10-28T09:56:24.723Z	2017-10-28T09:56:45.956Z	completed

Output for a single-tenant system or when initiated by an intra-tenant administrator on a multi-tenant system. This is a sample and not all output columns are represented.

```

show --id=2017-10-28T11_06_51_561-0400
Version      1
BackupId     2017-10-28T11_06_51_561-0400
BackupType   default
IncludeAllSourcesForBinaryBackup false
Slug         default slug
Comments     default comment
Name         2017-10-28T11_06_51_561-0400
State        completed
Size         1.58721695e+08
Owner        sasin
StartTimeStamp 2017-10-28T15:06:52.360Z
EndTimeStamp 2017-10-28T15:08:30.229Z
Purged       false

```

Sources:

SourceId	Name	Address	State
rabbitmq	SAS Message Broker	rdcesx14016.race.sas.com	completed
consul	SAS Configuration Server	rdcesx14016.race.sas.com	completed
cas-shared-default	SAS Cloud Analytic Services	rdcesx14016.race.sas.com	completed
postgres	SAS Infrastructure Data Server	rdcesx14016.race.sas.com	completed

On a multi-tenant system, the provider sees the administration details of the specified deployment backup. The intra-tenant administrator sees details of the specified tenant backup.

On a single-tenant system, the administrator sees the details of the specified backup on that system.

SAS Restore CLI

Restore CLI Help

Example:

```

sas-admin restore -help

```

Start Restore Operation Help

Command:


```
StartTimeStamp          2017-10-28T15:33:24.330Z
EndTimeStamp
```

Start a restore for a default backup or binary backup, which has `includeAllSourcesForBinaryBackup=true`.

```
sas-admin restore start --backup-name=2017-10-25T05_28_24_244-0400
```

Start a restore for a specified backup and forces the backup to proceed even if the validation (database list mismatch) fails:

```
sas-admin restore start --backup-name=2017-10-25T05_28_24_244-0400 force=true
```

On a multi-tenant system, when the restore is initiated by the provider administrator, this restores all onboarded tenants available in the backup. When the restore is initiated by an intra-tenant administrator, this restores that tenant.

On a single-tenant system, the restore process restores that system from the specified backup.

Show History of Restore Operations Help

Command:

```
list --help
```

Example:

```
sas-admin restore list help
```

Output:

NAME:

```
sas-admin restore list - Shows the history of restore operations.
```

USAGE:

```
sas-admin restore list [command options] [arguments...]
```

OPTIONS:

```
--limit, -l "10"    Specifies the maximum number of restore jobs to return. The default is 10.
--start, -s "0"     Specifies the 0-based offset of the first restore job to return.
                    The default is 0 for the first one.
```

Show Restore Operations List

Command:

```
list
```

Example:

```
sas-admin restore list
```

Output in a multi-tenant system. This is a sample and not all output columns are represented.

Version	RestoreId	BackupName	BackupType	State
1	2017-10-28T07_23_44_594-0400	2017-10-28T07_16_40_223-0400	default	completed
1	2017-10-23T11_24_44_602-0400	2017-10-23T06_09_39_355-0400	default	completed
1	2017-10-23T06_42_51_525-0400	2017-10-23T06_37_01_017-0400	binary	completed
1	2017-10-23T06_33_11_357-0400	2017-10-23T06_30_23_033-0400	default	completed
1	2017-10-17T09_17_41_209-0400	2017-10-17T09_13_04_082-0400	default	completed

Output in a single-tenant system or when the backup is initiated by an intra-tenant administrator on multi-tenant system. This is a sample and not all output columns are represented.

Version	RestoreId	BackupName	BackupType	Slug
---------	-----------	------------	------------	------

1	2017-10-28T11_33_24_330-0400	2017-10-28T11_06_51_561-0400	default	slug
1	2017-10-28T10_14_09_371-0400	2017-10-28T09_04_37_140-0400	default	new-restore-slug
1	2017-10-27T03_21_26_623-0400	2017-10-26T02_14_03_814-0400	default	new-restore-slug
1	2017-10-26T02_19_59_068-0400	2017-10-25T07_45_52_386-0400	default	new-restore-slug
1	2017-10-26T02_15_38_600-0400	2017-10-26T02_14_03_814-0400	default	new-restore-slug
1	2017-10-25T07_43_34_590-0400	2017-10-25T07_39_27_192-0400	default	new-restore-slug
1	2017-10-25T05_35_16_499-0400	2017-10-25T05_28_24_244-0400	default	new-restore-slug
1	2017-10-25T03_01_25_128-0400	2017-10-25T02_51_31_855-0400	binary	restore
1	2017-10-24T06_42_56_211-0400	2017-10-16T03_33_38_720-0400	default	slug
1	2017-10-18T06_05_20_455-0400	2017-10-18T06_03_35_463-0400	binary	new-restore-slug

This list shows all the restores that are available to the logged in user, sorted in time order, newest to oldest.

On a multi-tenant system, the provider administrator sees all deployment restores. The intra-tenant administrator sees all available tenant restores for that tenant.

On a single-tenant system, the administrator sees all available restores on that system.

Show the most recent 10 restores from the restore history:

```
sas-admin restore list
```

OR

```
sas-admin restore list -s=0 -l=10
```

OR

```
sas-admin restore list --start=0 --limit=10
```

Show Restore Job Details Help

Command:

```
show -help
```

Example:

```
sas-admin restore --help
```

Output:

NAME:

```
sas-admin restore show - Shows the details of a restore operation.
```

USAGE:

```
sas-admin restore show [command options] [arguments...]
```

OPTIONS:

```
--id, -i Specifies the unique ID of the restore operation for which details have been requested.
```

Show Restore Job Details

Command:

```
show
```

Example:

```
sas-admin restore show --id=2017-10-28T07_23_44_594-0400
```

Output for a multi-tenant system:

Version	1
RestoreId	2017-10-28T07_23_44_594-0400
BackupName	2017-10-28T07_16_40_223-0400

```

BackupType          default
Slug
Comments
State              completed
Owner              sasboot
OwnerTenantId      provider
IncludeAllSourcesForBinaryBackup false
StartTimeStamp     2017-10-28T11:23:44.606Z
EndTimeStamp       2017-10-28T11:25:18.357Z

```

Output for tenant restores. This is a sample and not all output columns are represented.

RestoreId	Tenant	StartTimeStamp	EndTimeStamp	State
2017-10-28T07_23_44_594-0400	cyberdyne	2017-10-28T11:23:45.220Z	2017-10-28T11:23:53.921Z	completed
2017-10-28T07_23_44_594-0400	intech	2017-10-28T11:23:46.108Z	2017-10-28T11:24:11.570Z	completed
2017-10-28T07_23_44_594-0400	provider	2017-10-28T11:23:46.564Z	2017-10-28T11:25:18.357Z	completed
2017-10-28T07_23_44_594-0400	acme	2017-10-28T11:23:45.674Z	2017-10-28T11:24:03.541Z	completed

Output for a single-tenant system or when initiated by an intra-tenant administrator on a multi-tenant system. This is a sample and not all output columns are represented.

```

Version          1
RestoreId        2017-10-28T11_33_24_330-0400
BackupName       2017-10-28T11_06_51_561-0400
BackupType       default
IncludeAllSourcesForBinaryBackup false
Slug             slug
Comments
Name            2017-10-28T11_33_24_330-0400
State           completed
Owner           sasin
StartTimeStamp  2017-10-28T15:33:25.223Z
EndTimeStamp    2017-10-28T15:34:21.233Z

```

Sources:

Version	SourceId	Name	Address	State
1	rabbitmq	SAS Message Broker	rdcesx14016.race.sas.com	completed
1	postgres	SAS Infrastructure Data Server	rdcesx14016.race.sas.com	completed
1	consul	SAS Configuration Server	rdcesx14016.race.sas.com	completed

Backup and Restore: Schedule a Backup

Default Scheduling in Backup

The default schedule backup for a default backup is to run every Sunday at 1:00 a.m.. You can see the following log message in the Backup Service log: Default schedule created for BackupService to run backup job every Sunday 1AM.

For scheduling, the following services need to be started for the backup schedule to work:

- sas-viya-identities-default
- sas-viya-scheduler-default
- sas-viya-jobdefinitions-default
- sas-viya-jobexecution-default

■ sas-viya-restexecutionprovider-default



If one of the services is not running during Backup Service bootstrap, then the Backup Service retries every 5 minutes for 25 tries to schedule default backup. If after 25 tries the backup is still not scheduled or one of the dependent services is still not running, then the following error message is displayed: Cannot schedule backup since maximum retry attempt is reached and one of the dependent services is still not running. Check and ensure that all the required services are started, and then restart the Backup Service (sas-viya-deploymentBackup-default). This schedules the default backup.

Once the backup schedule has been created, it is displayed in the SAS Environment Manager.

The screenshot shows the SAS Environment Manager interface. The 'Scheduling' section is active. On the left, there is a 'Jobs Filter' panel with a 'Date Created' dropdown and a 'Start date' input field. The main area displays 'Jobs (7)' with a search bar and a table. The table has two columns: 'Name' and 'Scheduled'. The first row is 'DEFAULT_BACKUP_SCHEDULE' with a green checkmark in the 'Scheduled' column, and it is circled in red. The second row is 'BINARY_BACKUP_SCHEDULE' with a green checkmark.


Name	Scheduled
DEFAULT_BACKUP_SCHEDULE	✓
BINARY_BACKUP_SCHEDULE	✓

You can add one or more triggers to the schedule. To add a trigger to the schedule:




- 1 Click  in the left pane to display the Scheduling window.
- 2 Select the name of the schedule that you created. In the figure above, the schedule is named DEFAULT_BACKUP_SCHEDULE. Click . The Edit Schedule dialog box is displayed.

Edit Schedule


Name:

Run as: 

Trigger type:

Available triggers   

<input type="checkbox"/>	Name	Enabled
<input type="checkbox"/>	Default_Backup_Schedule_Trigger	<input checked="" type="checkbox"/>

- 3 Click  to add a trigger to the schedule. The New Trigger window is displayed.

New Trigger

Name: *

Frequency: Interval: days

Time: *

Time zone:

Start date:

End:

- 4 Update the fields as appropriate. You must complete the **Name** and **Time** fields.
- 5 Click **Save** to save the new trigger.
If you want to return to the default settings, click **Reset**.

Edit Schedule

Name:

Trigger type:

Available triggers

<input type="checkbox"/>	Name	Enabled
<input type="checkbox"/>	Default_Backup_Schedule_Trigger	<input checked="" type="checkbox"/>
<input type="checkbox"/>	New trigger for scheduling backup	<input checked="" type="checkbox"/>

- 6 Click **Save** to save the new job schedule.

Schedule a Binary Backup

In SAS Viya, a default backup is scheduled through the Backup Service bootstrap. If you want to schedule a binary backup, then manual steps needs to be executed. When using curl commands, the user needs to get the Authorization token for the administrator user.

For information about how to obtain a token, see [Obtain an Access Token Using Password Credentials](#) .

To create a schedule for a binary backup:

- 1 You need a jobDefinition ID. This jobDefinition ID is already created when default backup is scheduled.
- 2 Using the jobDefinition ID, you need to create a jobExecution Request.
- 3 The jobExecution response to the jobExecution request gives a link to submit the job to be scheduled.
- 4 Once the job is submitted, it is visible through SAS Environment Manager and you can schedule your job.

The following steps provide the details to schedule a binary backup:

- 1 Execute the jobDefinitions REST API to get the jobDefinition ID. This ID was created while scheduling Default backup. The jobDefinition can be used to create schedule for binary backup. To get the jobDefinition ID for an existing jobDefinition, run the following curl command:

```
curl --verbose --header "Accept: application/json" --header "Authorization: bearer <token>" --request "GET"
  "<protocol>://<host>:<port>/jobDefinitions/
  definitions?limit=20&filter=in(name,"DEFAULT_BACKUP_SCHEDULE")"
```

This will return a jobDefinition object defined for default backup schedule. From the response select the "id" parameter.

Sample Response:

```
items": [
  {
    "creationTimeStamp": "2017-10-28T13:40:55.566Z",
    "modifiedTimeStamp": "2017-10-28T13:40:55.566Z",
    "createdBy": "sas.deploymentBackup",
    "modifiedBy": "sas.deploymentBackup",
    "version": 1,
    "id": "7c0f326b-280f-42ef-a404-39c9c775f325",
    "name": "DEFAULT_BACKUP_SCHEDULE",
    "type": "REST"
  }
]
```

The jobDefinition ID is highlighted in the code above.

- 2 Create a job request once the jobDefinition ID is known. The following parameters need to be populated to create a jobRequest object.
 - **name**—Name of the job request to be created. This name is used as the label in the SAS Environment Manager.
 - **description**—One-line description of jobRequest to be created.
 - **jobDefinitionURI**—This is the jobDefinition ID that was obtained when the curl command was run.
 - **contentType**—This is the content type of the DeploymentBackup Request.

If multi-tenant: contentType = "application/vnd.sas.backup.deployment.request+json"

If single-tenant: contentType = "application/vnd.sas.backup.request+json"

If you are running a single-tenant environment, the following is a sample curl command for creating a jobRequest for the job execution service:

```
curl --verbose --header "Accept: application/vnd.sas.job.execution.job.request+json"
  --header "Authorization: bearer <token>"
  --header "Content-Type: application/vnd.sas.job.execution.job.request+json" --data '{
  "name": "BINARY_BACKUP_SCHEDULE1",
  "description": "This jobRequest is to execute binary backup.",
  "jobDefinitionUri": "/jobDefinitions/definitions/7c0f326b-280f-42ef-a404-39c9c775f325",
  "arguments": {
    "contentType": "application/vnd.sas.backup.request+json",
    "backupType": "binary"
  }
}
```

```
}' --request "POST" "<protocol>://<host>:<port>/jobExecution/jobRequests"
```

If you are running a multi-tenant environment, the following is a sample curl command for creating a jobRequest for the job execution service:

```
curl --verbose --header "Accept: application/vnd.sas.job.execution.job.request+json"
  --header "Authorization: bearer <token> "
  --header "Content-Type: application/vnd.sas.job.execution.job.request+json" --data '{
  "name": "BINARY_BACKUP_SCHEDULE1",
  "description": "This jobRequest is to execute binary backup.",
  "jobDefinitionUri": "/jobDefinitions/definitions/7c0f326b-280f-42ef-a404-39c9c775f325",
  "arguments": {
    "contentType": "application/ vnd.sas.backup.deployment.request+json ",
    "backupType": "binary"
  }
}' --request "POST" "<protocol>://<host>:<port>/jobExecution/jobRequests"
```

Note: If you are creating a schedule for a binary backup, the `includeAllSourcesForBinaryBackup` property is not provided in the job request. In this case, only the Postgres data source is backed up. To back up other sources along with the Postgres binary backup, you need to run ad hoc backup.

- 3 Once the jobRequest is created, there is a link with “rel” as “submitJob” in the response. This link is used to submit the job. The following code is a sample response from the jobRequest command:


```
{
  "creationTimeStamp": "2017-10-30T10:43:11.203Z",
  "modifiedTimeStamp": "2017-10-30T10:43:11.203Z",
  "createdBy": "sasin",
  "modifiedBy": "sasin",
  "version": 2,
  "id": "9b7bd81d-451d-43be-8a72-0a14f3a288f6",
  "name": "BINARY_BACKUP_SCHEDULE1",
  "description": "This jobRequest is to execute binary backup.",
  "jobDefinitionUri": "/jobDefinitions/definitions/7c0f326b-280f-42ef-a404-39c9c775f325",
  "arguments": {
    "contentType": "application/vnd.sas.backup.request+json",
    "backupType": "binary"
  },
  "properties": [],
  "links": [
    {
      "method": "GET",
      "rel": "self",
      "href": "/jobExecution/jobRequests/9b7bd81d-451d-43be-8a72-0a14f3a288f6",
      "uri": "/jobExecution/jobRequests/9b7bd81d-451d-43be-8a72-0a14f3a288f6",
      "type": "application/vnd.sas.job.execution.job.request"
    },
    {
      "method": "POST",
      "rel": "submitJob",
      "href": "/jobExecution/jobRequests/9b7bd81d-451d-43be-8a72-0a14f3a288f6/jobs",
      "uri": "/jobExecution/jobRequests/9b7bd81d-451d-43be-8a72-0a14f3a288f6/jobs",
      "responseType": "application/vnd.sas.job.execution.job"
    }
  ]
}
```

The “rel” and the corresponding link are highlighted in the code above.

- 4 To submit the job, execute the following curl command:



```
curl --verbose --header "Accept: application/vnd.sas.job.execution.job+json"
--header "Authorization: bearer <token>"
--request "POST" "<protocol>://<host>:<port>/jobExecution/jobRequests/
9b7bd81d-451d-43be-8a72-0a14f3a288f6/jobs"
```

The job is created for the given jobRequest and the job is displayed in the SAS Environment Manager.




- 5 Once the job is created, you can schedule the job by adding a schedule trigger in the SAS Environment Manager.
 - a Select the job name in the SAS Environment Manager and click . The Edit Schedule dialog box is displayed.

Edit Schedule


Name:

Run as:  

Trigger type:

Available triggers   

Name	Enabled

- b In the **Run as** field, select the user from the Selected Identities window who should run the job. This user should be the administrator user.
- c Click  to add a trigger to the schedule. The New Trigger window is displayed.

New Trigger

Name: *

Frequency: Interval: days

Time: *

Time zone:

Start date:

End:

- d Update the fields as appropriate. You must complete the **Name** and **Time** fields.
 - e Click **Save** to save the new trigger.
- If you want to return to the default settings, click **Reset**.

- 6 Click **Save** to save the new job schedule.

Backup and Restore: Disaster Recovery

Overview

The following terminology is used in this section of the document:

Source machine

Machine where the backups were taken and these backups need to be restored on another machine.

Target machine

Alternate host having the same SAS Viya software installed on it as the source machine. The recovery operation should be executed on this alternate host from the backup taken on the source machine.

Source shared vault

Shared vault location for the source machine.

Target shared vault

Shared vault location for the target machine.

Note: Disaster recovery is not supported using the binary backup.

For a multi-tenant setup, the disaster recovery needs to be performed by a provider administrator. For a single-tenant setup, an administrator can perform the disaster recovery process.

Assumptions

The following assumptions are made for a disaster recovery:

- The content present on the source machine sharedVault should be copied to some other network location. This is referred to as the target shared vault.
- The SAS Viya configuration on the source machine and target machine is the same.
- All the services started successfully on the target machine.

Perform a Disaster Recovery

- 1 Copy the contents of the source shared vault to some location. This is also used as the target shared vault.
- 2 Make sure the target shared vault location has the same access rights as the source shared vault.
- 3 Go to the SAS Environment Manager on the target machine and navigate to the backup service configuration. Change the *sharedVault* property value to the location for the target shared vault.

Note: The content needs to be available under the target shared vault location before doing this. If you change the shared vault location and then copy the content, you risk losing data on the target machine.

- 4 As soon as the shared vault is changed, if there is any request issued for the backup or restore endpoints or if the service is restarted, then the system evaluates whether it needs to synchronize the local history with the Postgres tables. This is done by checking that the last successful backup from the global history is available in the Postgres table. If it is not available, the local history files are synchronized with the Postgres tables.
- 5 Go to the target machine command prompt, or the machine where the CLI utilities are installed. For more details about the CLI utilities, see [Backup and Restore: Command Line Interface](#).

Use the `sas-admin backup list` command to see the backup history list. You should be able to see all the backups that are available on the source machine listing output.

- 6 Select a backup ID from the list to use in the restore operation.
- 7 Using the command line interface, enter the following command:

```
sas-admin restore start --backup-name=<backup-id>
```

- 8 Make a note of the JobId printed on the console.
- 9 To see the details of restore operation, enter the following command:

```
sas-admin restore show -id-=JobId
```

- 10 While the restore is running and Postgres is being restored, all of the services might not respond. The CLI command might not give the expected response and give a 403 error. You need to keep running the `sas-admin restore show -id-=JobId` until you see the proper output and the status is either completed or failed.
- 11 If the restore job details show the status as completed, the restore is successful.
- 12 Restart all of the services.
- 13 After you restart all of the services, you need to restore CAS. To restore CAS, see [Restore CAS Server Access Controls and Caslib Information on page 151](#).

Once all of the steps are done, the disaster recovery is complete.

If a restore job's status is failed, look in the logs to see what has failed and then take the necessary actions.

You can also use the [Backup Manager on page 152](#) or the [Backup and Restore: Command Line Interface on page 161](#) to perform the restore steps.

Backup and Restore: Troubleshooting

Backup and Restore: Logs

The backup and restore facility generates the following logs that can be used in troubleshooting:

- On the host where the backup service deployed, service logs are created under the path `/var/log/sas/viya/deploymentBackup/default/xxxxx.log`. The name of the log files is based on the time at which the backup service was started.
- On each of the data sources, backup logs are created under the path `/var/log/sas/viya/backup-agent/default/xxxxx.log`. The name of the log files is based on the time at which the backup agent service was started.
- If a restore of the SAS Infrastructure Data Server fails, the log files for this restore operation backup log files are dumped to `/opt/sas/viya/config/backup/<backup_id>/<tenant_id>/postgres` on the host where a data source resides. This is the location of the local vault.

Backup and Restore: Error and Warning Messages

A Backup or Restore Is Already in Progress

This message indicates that there is already a backup or restore operation in progress. You cannot initiate multiple backup and restore operations at the same time.

Database List in the Backup and Postgres Database Do Not Match

Databases in backup: <source list in backup> and Databases in Postgres: <source list in database> do not match. Set force option in restore request to true to force restore:

This indicates that the databases present in Postgres at the time of the backup and the databases currently in Postgres do not match. The service would not restore a database missing in the current system in this case.

TIP If this is acceptable and a user still wants to restore the remaining databases, then you need to set 'force' field sent in body of restore request to 'true' to force restore. When force restore is set to true, the restore operation restores only the databases that are currently in Postgres and which are also available in backup.

```
curl -X POST -H 'Authorization: bearer <token>' -H 'Accept: application/vnd.sas.restore.job+json'
-H 'Content-Type: application/vnd.sas.restore.request+json' -d '{ "slug": "<slug name>",
"comments": "<comments>", "backupName": "<name>", "force": true
}' '<protocol>://<host>:<port>/deploymentBackup/restores/jobs'
```

Configuration with the ID "{0}" Was Not Found

This indicates that the configurationId provided in the backup request is not available or supported.

TIP Currently only `default` is the value supported for configurationId. Modify the backup request body to set the configurationId to `default` and try again.

Error Code 403 While Retrieving Information Related to Restore Operation

While a Postgres restore is running, it might take a while for all services to respond. In this case, the user might see 403 Forbidden error.

Request Contains Invalid Values for the Start or Limit Parameters

The request contains invalid values for the start (`{0}`) or limit (`{1}`) parameters. The values received for the start parameter and the limit parameter are invalid. Use positive integers as values for the start and limit parameters.

Resubmit the request with valid values.

Specified Backup Does Not Have a Directory in the Vault or the History File

The specified backup is not found in the shared vault or the history file. Try using another backup to initiate the restore.

Invalid Shared Vault Location

The shared vault location is invalid. Set the valid shared vault location from the SAS Environment Manager. See [Modify the Backup Configuration Using the Environment Manager on page 142](#).

List Can Only Contain Onboarded Tenants

The list of tenants should contain only tenants with a state of "onboarded". The following tenants are either not onboarded or are not valid: `{0}`.

Only onboarded tenants can be provided in the tenant list. Remove the tenants listed in the error from the tenant list.

backupType Value of Binary Contains a List of Tenants

The list of tenants should not be provided for backupType value of 'binary'. The Deployment Backup of backupType value 'binary' can be triggered only by a Provider Administrator user in multi-tenant deployment.

A binary backup cannot be taken for a subset of tenants in a multi-tenant environment. Remove the list of tenants from the request body if a binary backup is to be taken. You can also change the backup type to default for taking the backup for the tenants provided in the tenant list.

Also, use the Provider Administrator credentials to initiate the backup in a multi-tenant environment.

Multi-Tenant Deployment Can Be Triggered Only by a Provider Administrator

Use Provider Administrator credentials to initiate the deployment backup.

Shared Vault Is Not Accessible

The shared vault "`{0}`" is not accessible. Ensure that the shared vault is accessible before starting the job.

Check to see whether the shared vault location is accessible to the 'sas' user in 'sas' group.

Specified Backup ID Is Incompatible with the Target System

The specified backup ID is incompatible with the target system. Use a different backup ID to restore to the target environment.

The database mode of the environment from which the backup was taken is different from the target system. Ensure you are using the appropriate backup from the appropriate environment.

Command Line Interface

Command-Line Interface: Overview	182
Introduction	182
Inventory	182
Key Points	184
About the Examples	184
Command-Line Interface: Preliminary Instructions	185
Set the SSL_CERT_FILE Environment Variable	185
Create at Least One Profile	185
Use a Profile to Sign In	186
Command-Line Interface: Syntax	187
Structure	187
Integrated Help	187
Output Type	190
Global Command: Profile	190
Global Command: Authenticate	192
Command-Line Interface: Troubleshooting	192
CLI Examples: Audit	192
Examples	192
Command-Line Examples: CAS Authorization	193
Getting Access Information	193
Managing Access Controls	193
Details and Tips	195
Command-Line Examples: General Authorization	196
Getting Access Information	196
Managing Rules	197
Details and Tips	198
CLI Examples: Backup	199
Examples	199
CLI Examples: Restore	199
Examples	199
CLI Examples: CAS Administration	199
Facilitate Guest Access	200
Manage CAS Role Memberships	200
Manage SAS Sessions	200
Manage SAS Formats	200
Manage Tables	201
Manage Caslibs	202
Details	202

CLI Examples: Configuration	203
Examples	203
Details	203
CLI Examples: Compute	203
Examples	203
CLI Examples: Folders	204
Examples	204
Details	204
CLI Examples: Fonts	205
Examples	205
Details	205
CLI Examples: Device Management	206
Examples	206
CLI Examples: Identities	207
Examples	207
Details	207
CLI Examples: Licensing	208
Examples	208
CLI Examples: Job	208
Examples	208
Details	209
CLI Examples: Reports	209
Examples	209
Details	209
CLI Examples: Tenant Administration	210
Examples	210
Details	210
CLI Examples: Transfer	210
Examples	210
Details	211

Command-Line Interface: Overview

Introduction

SAS Viya contains administrative command-line interfaces (CLIs). In SAS Viya, a CLI is a user interface to the SAS Viya REST services where you enter commands on a command line and receive a response back from the system. You can use a CLI to interact directly with SAS Viya programmatically without a GUI.

Inventory

The following administrative CLIs are available in SAS Viya:

Name	Scope and Examples
admin	Hosts other CLIs that run as plug-ins to this one. The top-level administrative command-line interface that is used to initialize, authenticate, and execute other plug-ins.
audit	Gets SAS audit information. See “CLI Examples: Audit” .
authorization	Gets general authorization information and manages rules. See “Command-Line Examples: General Authorization” on page 196.
backup	Manages backups. See “CLI Examples: Backup” on page 199.
restore	Manages restore operations. See “CLI Examples: Restore” .
cas	Manages CAS administration and authorization. See “CLI Examples: CAS Administration” , and “Command-Line Examples: CAS Authorization” on page 193. For information about cas environment variables, see CAS Administration: Details on page 202.
configuration	Manages the operations of the configuration service. See “CLI Examples: Configuration” on page 203.
compute	Manages the operations of the compute service. See “CLI Examples: Compute” on page 203.
folders	Gets and manages SAS folders. See “CLI Examples: Folders” on page 204.
fonts	Manages fonts that are provided by SAS as well as custom fonts that are registered in SAS Visual Analytics 8.2. See “CLI Examples: Fonts” on page 205.
devices	Manages mobile device blacklist and whitelist actions and information. See “CLI Examples: Device Management” .
identities	Gets identity information, and manages custom groups. See “CLI Examples: Identities” on page 207.
licenses	Manages SAS product license status and information. See “CLI Examples: Licensing” .
job	Manages the operations of the job flow scheduling service. See “CLI Examples: Job” .
reports	Manages SAS Visual Analytics 8.2 reports. See “CLI Examples: Reports” .

Name	Scope and Examples
tenants	Manages tenants in a multi-tenant deployment. See “CLI Examples: Tenant Administration” on page 210 .
transfer	Promotes SAS content. See “CLI Examples: Transfer” on page 210 .

Key Points

Here are key points for using the CLIs:

- To prepare to use the admin CLI plug-ins, see [“Command-Line Interface: Preliminary Instructions” on page 185](#).
- Commands and subcommands are case-sensitive.
- You must precede the options of the commands with `--`.

Note: Some command options support a shortcut notation in which a single hyphen precedes the option. For example, you can use `--help` or `-help` for the help global option.

- Use the Help from within each CLI for information about the available commands, subcommands, and options. For the admin CLI plug-ins, see [“Command-Line Interface: Syntax” on page 187](#).

Note: This document reflects sas-admin CLI functionality in the initial release of SAS Viya 3.3. The integrated Help supersedes this documentation and provides the most current information about any expanded or enhanced functionality.

About the Examples

The following points apply to all of the examples in this document:

- The examples assume that you have signed in to SAS Viya using the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).
- The examples explicitly specify all necessary options. In practice, you might find it more efficient and concise to use environment variables where available. Remember to clear values for any environment variables when appropriate.
- The examples include line breaks within commands for presentation purposes only. Do not include line breaks when you submit a command.
- The examples generally include single quotation marks when quotation marks are required. Use the quotation marks that are appropriate for your platform.
- The examples generally assume that you are using the default profile, rather than a named profile.

Note: If you logged in using a named profile, you must do one of the following to use that profile:

- set the `SAS_CLI_PROFILE` environment variable to the name of the profile. This will remain in affect until you log off from the environment.

For example, suppose that you have set the `SAS_CLI_PROFILE` environment variable to `Target1` as follows: `export SAS_CLI_PROFILE=Target1`

Then, you can log on to the environment with the “Target1” profile with this command: `sas-admin auth login`

- include the `profile` global option on each CLI command as follows: `sas-admin --profile profile-name CLI-name CLI-commands`

Then, you can log on to the environment with the “Target1” profile with this command: `sas-admin --profile Target1 auth login`

Command-Line Interface: Preliminary Instructions

Complete the following required preliminary tasks before you use a CLI.

Set the `SSL_CERT_FILE` Environment Variable

If your environment is enabled for Transport Layer Security (TLS), then you must set the `SSL_CERT_FILE` environment variable to the path location of the `trustedcerts.pem` file.

If the `SSL_CERT_FILE` environment variable is not already set, complete these steps:

- 1 In a command window on the SAS Viya machine, navigate to the following directory: `/opt/sas/viya/home/bin`.
- 2 Set the environment variable as follows: `export SSL_CERT_FILE=/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts/trustedcerts.pem`.

Create at Least One Profile

If you have not already created a profile for the environment that you want to use, complete the following steps:

- 1 In a command window on the SAS Viya machine, navigate to the following directory: `/opt/sas/viya/home/bin`.
- 2 At the command prompt, enter a command to initialize a new profile. Here are examples:

To create a default (unnamed) profile, enter: `sas-admin profile init`

To create a profile called `prod`, enter: `sas-admin --profile prod profile init`

You can use a named profile to access different environments using the same set of CLIs. See [“Default Profile and Named Profile” on page 190](#) for information about why you might want to use a named profile..

Note: Running the `profile` global command creates a `config.json` file in this directory: `home-directory/.sas`. For more information, see [“Overview” on page 190](#).

- 3 Respond to the subsequent prompts as follows:

Service Endpoint	Specify the URL for the SAS Viya environment. Use the following format: <i>communications-protocol://web-server-host-name:web-server-port</i> For example: <code>https://host.example.com:443</code>
------------------	--

Output type	Specify your preferred format for CLI output (<code>text</code> , <code>json</code> , or <code>fulljson</code>). Note: For more information about the output types, see “Output Type” on page 190 .
Enable ANSI colored output	Specify whether to enable colored output (<code>y</code> or <code>n</code>).

- Repeat steps 2 and 3 for any additional profiles that you want to create.

Use a Profile to Sign In

- In a command window on the SAS Viya machine, navigate to the following directory: `/opt/sas/viya/home/bin`.
- At the command prompt, enter a command to initiate the sign-in process. Here are examples:

To use your default (unnamed) profile (assuming that the <code>SAS_CLI_PROFILE</code> environment variable is not set), enter:	<code>sas-admin auth login</code>
--	-----------------------------------

To use a profile called <code>prod</code> , enter:	<code>sas-admin --profile prod auth login</code>
--	--

- At the subsequent prompts, enter your user ID and password.

By default, your authentication remains active for 12 hours. You can use the `auth logout` command to sign out.

Note: When you run the `auth login` global command, a bearer token is written to the `credentials.json` file in this directory: `home-directory/.sas`. For more information, see [“Overview” on page 190](#).

Note: If you logged in using a named profile, you must do one of the following to use that profile:

- set the `SAS_CLI_PROFILE` environment variable to the name of the profile. This will remain in affect until you log off from the environment.

For example, suppose that you have set the `SAS_CLI_PROFILE` environment variable to `Target1` as follows: `export SAS_CLI_PROFILE=Target1`

Then, you can log on to the environment with the “Target1” profile with this command: `sas-admin auth login`

- include the `profile` global option on each CLI command as follows: `sas-admin --profile profile-name CLI-name CLI-commands`

Then, you can log on to the environment with the “Target1” profile with this command: `sas-admin --profile Target1 auth login`

See Also

[“Command-Line Interface: Overview” on page 182](#)

Command-Line Interface: Syntax

Structure

The basic structure of a command-line interface (CLI) command is:

```
sas-admin interface-name [global options] command [command options] [subcommand] [subcommand options] [arguments]
```

sas-admin

specifies the sas-admin CLI.

interface name

specifies the CLI plug-in.

[global options]

specifies options that are applicable to all CLIs.

command

specifies a command that is specific for the CLI that you are using.

[command options]

specifies options for the CLI-specific command that you are using.

[subcommand]

specifies a subcommand for the CLI command that you are using.

[subcommand options]

specifies options for the subcommand.

[arguments]

specifies arguments for options.

Here are some basic examples of issuing CLI commands:

Example: Change the output type that is used with the audit CLI to JSON.

```
sas-admin audit --output json
```

Example: Show more detailed information for the mobile device CLI `blacklist list` command.

```
sas-admin --verbose devices blacklist list
```

For information about the CLIs that are available, see [“Inventory” on page 182](#).

Integrated Help

Global Commands

Use the integrated help within the CLI to learn about the available global commands, plug-ins, and global options. The global options apply to each CLI plug-in.

Example: List all the global commands, plug-ins, and global options for the admin CLI.

```
sas-admin help
```

Here is the output from this command in a Linux environment:

```
NAME:
  sas-admin - SAS Administrative Command Line Interface
```

USAGE:

```
sas-admin [global options] command [command options] [arguments...]
```

VERSION:

```
1.0.9
```

COMMANDS:

```
authenticate, auth, authn    Handles authentication to the target environment.
help, h                      Shows a list of commands or help for one command.
plugins                      Manages plugins.
profile, prof                Shows and updates options.
```

PLUGINS:

```
audit
authorization
backup
cas
compute
configuration
devices
folders
fonts
identities
job
licenses
reports
restore
tenant
transfer
```

GLOBAL OPTIONS:

```
--colors-enabled            Enables or disables ANSI colored output. [$SAS_CLI_COLOR]
--help, -h                  Shows help.
--insecure, -k              Allows connections to TLS sites without validating the server
certificates.
--locale "en"               Specifies a locale to use. [$LC_ALL, $LANG]
--output                    Specifies output format - text, json, fulljson. [$SAS_OUTPUT]
--profile, -p "Default"     Specifies a named profile to use. [$SAS_CLI_PROFILE]
--quiet, -q                 Quiets spurious output, data only.
--sas-endpoint              Sets the URL to the SAS services. [$SAS_SERVICES_ENDPOINT]
--verbose                   Shows detailed processing information and output.
--version, -v               Prints the version.
```

COPYRIGHT:

```
(c) 2016-2017 SAS Institute Inc. All Rights Reserved.
```

Example: Show the version of the CLI.

```
sas-admin --version
```

Here is the output from this command in a Linux environment:

```
sas-admin version 1.0.9
```

CLI Plug-in

Use the integrated help within the CLI to learn about the available commands, subcommands, and options for each CLI plug-in. Use the same syntax for each CLI, substituting the CLI plug-in name or command that you are getting help for.

Example: List all the commands for the devices plug-in to the admin CLI.

```
sas-admin devices --help
```

Here is the output from this command in a Linux environment:

NAME:

```
sas-devices
```

USAGE:

```
sas-admin devices command [command options] [arguments...]
```

COMMANDS:

```
authorized-devices    Manages the authorization of devices.
blacklist             Manages the list of blacklisted devices.
enforcement          Manages the policy enforcement of mobile devices.
help, h              Shows a list of commands or help for one command.
last-access          Manages the set of history records of the devices that use the mobile application.
whitelist            Manages the list of whitelisted devices.
```

Example: List the subcommands of the `blacklist` command that is a part of the devices plug-in to the admin CLI.

```
sas-admin devices help blacklist
```

Here is the output from this command in a Linux environment:

NAME:

```
sas-admin devices blacklist - Manages the list of blacklisted devices.
```

USAGE:

```
sas-admin devices blacklist [arguments...]
```

COMMANDS:

```
add                Adds a device to the blacklist.
delete            Deletes a device from the blacklist.
list              Lists the devices in the blacklist.
```

Example: List the options of the `list` subcommand of the `blacklist` command that is a part of the devices plug-in to the admin CLI.

```
sas-admin devices blacklist help list
```

Here is the output from this command in a Linux environment:

NAME:

```
sas-devices blacklist list - Lists the devices in the blacklist.
```

USAGE:

```
sas-devices blacklist list [command options] [arguments...]
```

OPTIONS:

```
--all            Returns all of the devices in the blacklist.
--limit "10"     Specifies the maximum number of devices to return. The default value is 10.
```

`--start "0"` Specifies the 0-based offset of the first device to return. The default value is 0.

Output Type

You must specify an output type for your CLI when you create your profile. The output types for CLIs are as follows:

- `text`
Specifies that the output from the CLI is in text format. This is the default format.
- `JSON`
Specifies that the output from the CLI is in JSON format.
- `fulljson`
Specifies that the output from the CLI is the entire JSON response. This option is useful when writing scripts in which you need access to the entire response in order to complete a task.

Global Command: Profile

Overview

Use the `profile` global command to create the connection profile that defines your SAS Viya deployment. This process creates the following two files in the directory `home-directory/.sas`:

- `config.json`
Contains information about your SAS Viya deployment, including the name of the connection profile, the service endpoint, and the output type (`text`, `json`, or `fulljson`).
- `credentials.json`
Will contain the authentication tokens for your session that are created after you issue the `auth login` global command.

Default Profile and Named Profile

You can create a default (unnamed) profile or a named profile. If you do not specify a named profile in the `sas-admin auth login` command, and the `SAS_CLI_PROFILE` environment variable is not set, then the default (unnamed) profile is used.

You can use a named profile if you need to work with two or more environments simultaneously from the same machine using the same set of CLIs. When you log on to an environment with a named profile, the associated token is stored for that specific profile. You can work in different environments at the same time by specifying different profile names in the commands. For example, suppose that you have different development, test, and production environments. You can create a separate, named profile for each environment to help distinguish the environment that you are connecting to.

You can also use a named profile to eliminate the requirement to create a profile with the correct settings every time you log on. Suppose that you know that you want to use the JSON output type and a certain endpoint. You can create a named profile with these settings. Then when you want to log on, you can specify this profile name and eliminate the need to create a new profile with these settings

Note: If you logged in using a named profile, you must do one of the following to use that profile:

- set the `SAS_CLI_PROFILE` environment variable to the name of the profile. This will remain in affect until you log off from the environment.

For example, suppose that you have set the `SAS_CLI_PROFILE` environment variable to `Target1` as follows: `export SAS_CLI_PROFILE=Target1`

Then, you can log on to the environment with the “Target1” profile with this command: `sas-admin auth login`

- include the `profile` global option on each CLI command as follows: `sas-admin --profile profile-name CLI-name CLI-commands`

Then, you can log on to the environment with the “Target1” profile with this command: `sas-admin --profile Target1 auth login`

Examples

Here are typical examples of creating default (unnamed) and named profiles:

Example: Create a default connection profile, specify the `text` output type, and specify the path to your SAS Viya deployment as the service endpoint.

Note: The `SAS_CLI_PROFILE` environment variable must not be set in order for this example to work.

```
sas-admin profile init
```

- At the prompt for **Service Endpoint**, enter the URL for the SAS Viya environment as follows: `https://endpoint URL`
- At the prompt for **Output type**, enter `text`.
- At the prompt for **Enable ANSI colored output**, enter `y` or `n`.

Here is the `config.json` file that was created. “**Default**” indicates that a default profile was created.

```
{
  "Default": {
    "ansi-colors-enabled": "false",
    "output": "text",
    "sas-endpoint": "https://endpoint URL"
  }
}
```

Example: In the same environment, create a connection profile that is named `Target1`, specify the `json` output type, and specify the path to your SAS Viya deployment as the service endpoint.

```
sas-admin --profile Target1 profile init
```

Here is the `config.json` file that was created. Notice that there are now two profiles: “Default” and “Target1”. Notice that the output type for the “Target1” profile is JSON.

```
{
  "Default": {
    "ansi-colors-enabled": "false",
    "output": "text",
    "sas-endpoint": "https://endpoint URL"
  },
  "Target1": {
    "ansi-colors-enabled": "false",
    "output": "json",
    "sas-endpoint": "https://endpoint URL"
  }
}
```

See “[Command-Line Interface: Preliminary Instructions](#)” on page 185 for more information about creating profiles.

Example: In the same environment, log on using the `Target1` profile that you created in the previous example.

```
sas-admin --profile Target1 auth login
```

Global Command: Authenticate

Use the `authenticate` global command to log on or log off from the environment. This process stores a token in the `credentials.json` file.

Note: The token expires after 12 hours, so you might need to re-execute the command at a later time to reconnect to the environment.

To log on, issue the `auth login` command. For syntax information, see [“Structure” on page 187](#). Here are typical examples:

- Use this command to log on with a default (unnamed) profile: `sas-admin auth login`. The `SAS_CLI_PROFILE` environment variable must not be set in order for this example to work.
- Use this command to log on with the profile named “Target1”: `sas-admin --profile Target1 auth login`.

If the `SAS_CLI_PROFILE` environment variable is set to `Target1`, then you can log on to the “Target1” environment as follows: `sas-admin auth login`.

To log off, issue the `auth logout` command. For syntax information, see [“Structure” on page 187](#). Here are typical examples:

- Use this command to log off from the SAS Viya environment that is specified in your default (unnamed) profile: `sas-admin auth logout`. The `SAS_CLI_PROFILE` environment variable must not be set in order for this example to work.
- Use this command to log off from a SAS Viya environment that is specified in the profile named “Target1”: `sas-admin --profile Target1 auth logout`.

If the `SAS_CLI_PROFILE` environment variable is set to `Target1`, then you can log off from the “Target1” environment as follows: `sas-admin auth logout`.

Command-Line Interface: Troubleshooting

Message: `token expired and refresh token is not set`

Explanation: You are not currently authenticated to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Message: `flag provided but not defined`

Explanation: You might have specified a global option in the wrong location. See [“Command-Line Interface: Syntax” on page 187](#).

CLI Examples: Audit

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Examples

Example: List the records of audit entries for reports.

```
sas-admin audit list --application reports
```


Example: List the records of audit entries of type security and sort by user.

```
sas-admin audit list --sort-by user --type security
```

Example: List the records of audit entries of type security and state of success, and then write the results as CSV to an output file:

```
sas-admin audit list --state success --type security --csv /tmp/outputfile.text
```

See Also

- [“Command-Line Interface: Overview” on page 182](#)
- [SAS Administration: Auditing](#)

Command-Line Examples: CAS Authorization

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Getting Access Information

Example: List tableA’s direct access controls.

```
sas-admin cas tables list-controls --server serverA --caslib caslibA --table tableA
```

Example: List tableA’s direct access controls and inherited settings.

```
sas-admin cas tables list-controls --server serverA --caslib caslibA --table tableA --list-type all
```

Example: Show effective (net) access to tableA for userA.

```
sas-admin cas tables list-controls --server serverA --caslib caslibA --table tableA
--control-type effective --user userA
```

Example: Show effective access to tableA for groupA and groupB.

```
sas-admin cas tables list-controls --server serverA --caslib caslibA --table tableA
--control-type effective --group 'groupA|groupB'
```

Example: Show the source of userA’s access to tableA.

```
sas-admin cas tables list-controls --server serverA --caslib caslibA --table tableA
--control-type origin --user userA
```

Managing Access Controls

Example: Remove all direct access controls from tableA.

```
sas-admin cas tables clear-controls --server serverA --caslib caslibA --table tableA
```

Example: Set a simple row-level access control on the CARS table so that members of groupA can see only those rows where the value in the Make column is **Ford**.

```
sas-admin cas tables add-control --server serverA --caslib caslibA --table CARS --group groupA
--grant Select --where "make='Ford'"
```

Example: Set an identity-based, row-level access control on the Salary table. The reason is so that each authenticated user can see only those rows where the value in the User column is his or her own user ID.

```
sas-admin cas tables add-control --server serverA --caslib caslibA --table salary --group "*"
--grant Select --where "User='SUB::SAS.Userid'"
```

Example: Enable guests to read data in the Public caslib.

```
1 sas-admin cas caslibs add-control --server serverA --caslib Public --guest --grant ReadInfo
  --superuser
2 sas-admin cas caslibs add-control --server serverA --caslib Public --guest --grant Select
  --superuser
```

- 1 This example is applicable to only a deployment where [guest access](#) is enabled.

In the standard configuration, because only a privileged user can modify access to the Public caslib, the superuser option is specified here. Only a member of the Superuser role for the specified CAS server can obtain elevated privileges by specifying the superuser option.

- 2 The grants in this example support reading of data, but do not support just-in-time loading of data. Instead of granting LimitedPromote to guest, consider using a different technique for ensuring that data is loaded.

Because this example does not create and reuse a dedicated superuser session, you must specify the superuser option in each command where elevated privileges are needed.

Example: Enable groupA to read data (and perform just-in-time data loading) in a new caslib.

```
1 sas-admin cas caslibs add-control --server serverA --caslib caslibA --group groupA --grant ReadInfo
2 sas-admin cas caslibs add-control --server serverA --caslib caslibA --group groupA --grant Select
3 sas-admin cas caslibs add-control --server serverA --caslib caslibA --group groupA
  --grant LimitedPromote
```

Example: Make the same changes as in the preceding example, but use an access control transaction so that you can review your changes before you commit them to the server.

```
1 sas-admin cas sessions create --name mysess --server serverA --superuser
2 sas-admin cas transactions checkout --session-id XYZ --server serverA --caslib caslibA
3 sas-admin cas caslibs add-control --session-id XYZ --server serverA --caslib caslibA
  --group groupA --grant ReadInfo
4 sas-admin cas caslibs add-control --session-id XYZ --server serverA --caslib caslibA
  --group groupA --grant Select
5 sas-admin cas caslibs add-control --session-id XYZ --server serverA --caslib caslibA
  --group groupA --grant LimitedPromote
6 sas-admin cas caslibs list-controls --session-id XYZ --server serverA --caslib caslibA
7 sas-admin cas transactions commit --session-id XYZ --server serverA
8 sas-admin cas sessions delete --session-id XYZ --server serverA
```

- 1 Start a session. If you are a member of the Superuser role for the associated CAS server, give the session Superuser status.
- 2 Check out the caslib into the session that you just started. Use the session ID that is returned from the preceding command. The session-id value `xyz` is used here for simplicity. Checking out an object automatically starts a transaction.
- 3 Grant access within the transaction.
- 4 Grant access within the transaction.
- 5 Grant access within the transaction.
- 6 Review the results. Because you supply the session ID, the output reflects the uncommitted changes in your session.
- 7 After you review the output from the list-controls command, commit your changes.

8 If you are finished, it is a good practice to delete your session.

Example: Replace any direct access controls on tableA with access controls from an external JSON file. In this example, the replacement access controls are derived from tableB and are then applied to tableA.

```
1 sas-admin cas tables list-controls --server serverA --caslib caslibA --table tableB > ac.json
```

```
2 sas-admin cas tables replace-controls --server serverA --caslib caslibA --table tableA
  --source-file ac.json
```

```
3 sas-admin cas tables list-controls --server serverA --caslib caslibA --table tableA
```

1 This example writes the direct access controls for tableB to the file `ac.json` in the directory from which you are running your CLI.

Note: This example assumes that the profile that you are using specifies `json` as your default output type. Otherwise, you must use the global option `--output` to specify `json` as the output type for this command. That option must immediately follow `sas-admin`.

Note: You can reference an absolute path or a relative path.

2 Delete any direct access controls on tableA, and replace them with the access controls that you wrote to the `ac.json` file.

3 Review the new set of direct access controls on tableA.

Note: You can modify the output from tableB before you use it to replace direct access controls on tableA.

Details and Tips

Basics

- Throughout this topic, the term *access control* refers to an access control in the CAS authorization system. To manage access to content objects and functionality, see [“Command-Line Examples: General Authorization” on page 196](#).
- You can add, delete, and replace only direct access controls.
- A request to delete a direct access control that does not exist does not generate an error.
- To modify access that a table inherits, set direct access controls on the parent caslib.
- You cannot modify access that a caslib inherits.
- Use of [access control transactions](#) is optional. You do not have to check out an object in order to modify its access controls.
- In the list-controls command, use the control-type and list-type options as follows:
 - Use the control-type option only if you want to obtain net access information (`effective`) or source information (`origins`).
 - The list-type options are not relevant if the control-type is `effective` or `origin`.
 - Use the list-type option only if you want to obtain inherited settings, in addition to direct access controls (`all`) or instead of direct access controls (`inherited`).
- You can obtain [origins on page 100](#) information for only one identity at a time.
- The value `serverA` is used in the examples for simplicity. A more typical server name is `cas-shared-default`.
- See [“Details” on page 202](#) for information about using environment variables with the CAS commands.

Principals

- To specify a particular identity (where supported), you must provide a user ID or a group ID rather than a name.

CAUTION! The user ID and the group ID that you provide are not validated. Make sure the IDs that you provide are accurate.

- To specify multiple users or multiple groups (where supported), use the pipe character (|) as a delimiter and enclose the string in single quotation marks (for example: `--user 'userA|userB'`). You cannot specify both users and groups in a single request.
- The group `*` corresponds to Authenticated Users. To specify that principal, enter `--group '*'`.
- The Guest principal represents all users who connect as guests. To specify that principal, enter the option `--guest` and do not specify a value.

Permissions

- To specify a particular permission (where supported), use one of the following case-insensitive values: ReadInfo, Select, LimitedPromote, Promote, CreateTable, DropTable, DeleteSource, Insert, Update, Delete, AlterTable, AlterCaslib, or ManageAccess. You cannot specify multiple permissions in a single request.
- For information about the scope and purpose of each permission, see [“CAS Authorization: Concepts” on page 91](#).

Fine-Grained Controls

- Row-level grants are always for the Select permission on a table. The syntax for [row-level permission filters](#) is the same as in other CAS authorization interfaces.
- You cannot set [column-level permissions](#) using this interface.

See Also

- [“Command-Line Interface: Overview” on page 182](#)
- [SAS Viya Administration: Cloud Analytic Services Authorization](#)

Command-Line Examples: General Authorization

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Getting Access Information

Example: Show detailed properties of a specified rule.

```
sas-admin authorization show-rule --id d85144aa-79dc-4852-b949-645cc5ff8ffc --details
```

Example: Show effective access for a specified object URI. To return information about contributing rules, you must specify fulljson output in your profile. Specifying the `output` global option in the CLI command inline is insufficient.

```
sas-admin authorization explain --target-uri /SASHome/**
```

Example: List all the rules in the deployment.

```
sas-admin authorization list-rules
```

Managing Rules

Example: Give groupA Read access to reportA.

```
sas-admin authorization authorize --permissions Read --group groupA
--object-uri /reports/reports/33db163a-716e-4980-a5bc-6c42a0278c40
```

Example: Provide [guest access](#) to reportA.

```
sas-admin authorization authorize --permissions Read --guest
--object-uri /reports/reports/33db163a-716e-4980-a5bc-6c42a0278c40
```

Example: Grant Authenticated Users Read access to folderA and its child members.

```
sas-admin authorization authorize --permissions Read --authenticated-users
--object-uri /folders/folders/2414f911-d276-4357-8550-fcf03753c9e7/**
--container-uri /folders/folders/2414f911-d276-4357-8550-fcf03753c9e7
```

Example: Delete a rule.

```
1 sas-admin authorization show-rule --id d85144aa-79dc-4852-b949-645cc5ff8ffc --details
2 sas-admin authorization remove-rule --id d85144aa-79dc-4852-b949-645cc5ff8ffc
```

- 1 Review the rule's properties so that you are certain you are deleting the correct rule.
- 2 Delete the rule.

Example: Change the principal in an existing rule so that the rule is assigned to Group B, which has `groupB` as its ID.

```
1 sas-admin authorization show-rule --id cd75a376-c5d4-4951-9e57-cf441610628c --details
2 sas-admin authorization update-rule --id cd75a376-c5d4-4951-9e57-cf441610628c --group groupB
```

- 1 Review the rule's properties so that you are certain you are modifying the correct rule.
- 2 Modify the rule.

Example: Include the Update and Delete permissions in an existing rule that already grants the Read permission.

```
1 sas-admin authorization show-rule --id cd75a376-c5d4-4951-9e57-cf441610628c --details
2 sas-admin authorization update-rule --id cd75a376-c5d4-4951-9e57-cf441610628c
--grant --permissions Read,Update,Delete
```

- 1 Review the rule's properties so that you are certain you are modifying the correct rule.
- 2 Modify the rule.

Example: Edit the description in an existing rule.

```
1 sas-admin authorization show-rule --id 0e8a6ce7-e51a-40cc-aeda-ee2a5efb53ca --details
2 sas-admin authorization update-rule --id 0e8a6ce7-e51a-40cc-aeda-ee2a5efb53ca
--description 'This is a revised description.'
```

- 1 Review the rule's properties so that you are certain you are modifying the correct rule.
- 2 Modify the rule.

Details and Tips

- Throughout this topic, the term *rule* refers to an authorization rule in the general authorization system. To manage access to CAS objects (such as caslibs and tables), see [“Command-Line Examples: CAS Authorization” on page 193](#).

- To assign a rule to a principal type, use one of the following options:

Guest	--guest
Authenticated Users	--authenticated-users
Everyone	--everyone

- To assign a rule to a particular identity, you must provide a user ID or a group ID, not a name. For example, to assign a rule to the SAS Administrators custom group, specify: `--group SASAdministrators`
- CAUTION!** The user ID and the group ID that you provide are not validated. Make sure the IDs that you provide are accurate.
- You can obtain the ID for a user or group from the **Users** page in SAS Environment Manager.
 - You can obtain the objectURI for a content object (such as a report) from the **Content** page in SAS Environment Manager. Select the object in the navigation pane. On the right, the **URI** field in the **Basic Properties** section contains the object URI. To target the object URI for a content object (such as a report) or a container (such as a folder), append a suffix. See [“Rule Targets” on page 122](#).
 - You can obtain the ID for a rule from the **Rules** page in SAS Environment Manager. Right-click a rule and select **Properties**. The last field in the Properties window contains the rule’s ID.
 - When you use the show-rule command, always specify that you want details to be returned. Some of the fields that can be essential to interpreting a rule are excluded from the default response. For example, a condition is not included in the default response.
 - When you use the update-rule command, specify only the options for the rule properties that you want to modify. For any option that you specify in the update-rule command, provide the complete replacement value or values.
 - When you use the explain command, the returned information indicates the effective (net) access of each relevant principal for all permissions.

Note: In the output from the explain command, the `grant` and `prohibit` values indicate effective (net) access, not direct settings. For example, a `prohibit` value in the output from the explain command is usually caused by the lack of any relevant grant, rather than by the existence of a relevant Prohibit rule. See [“Authorization Decisions” on page 98](#).
 - Enabling or disabling guest access involves more than running the `enable-guest-access` or `disable-guest-access` command. See the [guest access](#) documentation.

See Also

- [“Command-Line Interface: Overview” on page 182](#)
- [SAS Viya Administration: General Authorization](#)

CLI Examples: Backup

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Examples

Example: List the first 50 backup jobs.

```
sas-admin backup list --limit 50
```

Example: Start a binary backup named backupA.

```
sas-admin backup start --slug backupA
```

See Also

- [“Backup and Restore: Overview” on page 133](#)
- [“Command-Line Interface: Overview” on page 182](#)

CLI Examples: Restore

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Examples

Example: Show the history of restore operations.

```
sas-admin restore list
```

Example: Start a restore operation of a specified backup, and specify that the name of the restore operation is restoreA.

```
sas-admin restore start --backup-name backup-ID --slug restoreA
```

See Also

- [“Backup and Restore: Overview” on page 133](#)
- [“Command-Line Interface: Overview” on page 182](#)

CLI Examples: CAS Administration

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Facilitate Guest Access

Example: To facilitate guest access on the specified server, modify the direct access controls for the predefined caslibs on the server. To do this, use the controls that are defined in the specified source file. Perform this action with elevated privileges if the user is a member of the Superuser role. This is one of several required steps to enable guest access. For a complete set of instructions on how to enable guest access, see [“Authentication: Guest Access” on page 57](#).

```
sas-admin cas facilitate-guest --source-file path-to-controls-file --server serverA --superuser
```

The default access control file is located on the SAS Viya machine at this location: `/opt/sas/viya/home/share/guest/facilitate-guest-controls.txt`.

Manage CAS Role Memberships

Example: List the administrative users on the specified CAS server.

```
sas-admin cas admin-users list --server serverA
```

Example: Add the user user1 to the Superuser role on the specified CAS server.

```
sas-admin cas admin-users add --user user1 --server serverA
```

Example: Delete the group with the ID group1 and name group1_name from the Superuser role on the specified CAS server. The action is performed without prompting the user for confirmation since the `force` option is used.

```
sas-admin cas admin-users delete --group group1 --server serverA --name group1_name --force
```

TIP To delete a user or a group from the administrative users, you must specify the name of the user or the group as well as the identity of the user or the group. Use the `name` option to specify the name. Use the `user` option to specify the ID for a user. Use the `group` option to specify the ID for a group. In order to obtain the ID, use the `admin-users list` command.

Manage SAS Sessions.

Example: List the sessions of which you are the owner on the specified CAS server.

```
sas-admin cas sessions list --server serverA
```

Example: Using elevated privileges, list 50 sessions for which the owner is user1 on the specified CAS server, and sort by `state`. You must be a member of the Superuser role to run the command with elevated privileges.

```
sas-admin cas sessions list --server serverA --superuser --limit 50 --owner user1 --sort-by state
```

Example: Using elevated privileges, list all sessions for which the name contains the string `dataExplorer` on the specified CAS server. You must be a member of the Superuser role to run the command with elevated privileges.

```
sas-admin cas sessions list --server serverA --superuser --all --name-contains dataExplorer
```

Example: Using elevated privileges, show additional information about the session with ID 12345 on the specified CAS server. You must be a member of the Superuser role to run the command with elevated privileges.

```
sas-admin cas sessions show-info --superuser --session-id 12345 --server serverA
```

Manage SAS Formats

Example: Display the values for the specified SAS format (in this example, `$fruit`).


```
sas-admin cas formats show-info --server serverA --format '$fruit' --format-library userformats3
```

Example: Display all SAS format libraries for serverA.

```
sas-admin cas formats list --format-library=* --server serverA
```

Example: Display the search order for SAS format libraries for serverA.

```
sas-admin cas formats search-order --server serverA
```

Note: Format names that are used in the commands must be enclosed in single quotation marks.

If you run the show-info command for a format, the results are displayed according to these rules:

- If you request information about a non-locale format (such as `$charfmt`) and you do not specify the `ignore-locale` option, the default locale format will be returned. If the default system locale is `en_US` and the `ignore-locale` option was not specified, then the returned format for `$charfmt` is `en_US-$charfmt`.
- If you request information about a non-locale format (such as `$charfmt`), you must specify the `ignore-locale` option in order to return the actual format with no locale. If the `ignore-locale` option was specified, the returned format for `$charfmt` is `$charfmt`.
- If you request information about a format that has a non-existent locale, the format for the default system locale is returned. Suppose that you request information about `en_bogus-$charfmt`, which is a format that does not exist. If the default system locale is `en_US`, the returned format is `en_US-$charfmt`.
- If you request information about an invalid format, an error is returned. Suppose that you request information about `en-US-$charfmt`. The hyphen in the locale name (`en-US`) should have been an underscore (`en_US`). No format is returned, and an error is displayed.

See Also

See [“Data Administration: Reference” on page 298](#) for more information about user-defined formats.

Manage Tables

Note: These examples explicitly specify the `server` and `caslib` required options. However, using environment variables might be more efficient for these options. For more information about the environment variables, see [“Details” on page 202](#).

Example: For the specified caslib and CAS server, list all tables with names that contain the string `visual`, sort by `state`, and return a maximum number of 50 tables.

```
1 sas-admin cas help tables
2 sas-admin cas tables help list
3 sas-admin cas tables list --caslib caslibA --server serverA --name-contains visual --sort-by state --limit 50
```

- 1 Review the Help for the `cas tables` command.
- 2 Review the Help for the `list` subcommand of the `cas tables` command.
- 3 Issue the command to list the tables for the specified caslib and CAS server with the appropriate subcommand and options.

Example: Load the given table for the specified caslib and CAS server.

```
sas-admin cas tables load --table airlines --server serverA --caslib caslibA
```

Example: Unload the given table for the specified caslib and CAS server.

```
sas-admin cas tables unload --table airlines --server serverA --caslib caslibA
```

Example: Show information about the given table for the specified caslib and CAS server.

```
sas-admin cas tables show-info --table airlines --server serverA --caslib caslibA
```

Manage Caslibs

Note: These examples explicitly specify the `server` required option. However, using an environment variable might be more efficient for this option. For more information about the environment variables, see “Details” on page 202.

Example: On the specified server, create a caslib that is based on a file path.

```
sas-admin cas caslibs create path --name caslibA --path /tmp/dept --server serverA
```

Example: On the specified server, list the first 20 global caslibs. You must be a member of the Superuser role to run the command with elevated privileges.

```
sas-admin cas caslibs list --server serverA --scope global --limit 20 --superuser
```

Example: On the specified server, list the global caslibs starting at caslib number 21. You must be a member of the Superuser role to run the command with elevated privileges.

```
sas-admin cas caslibs list --server serverA --scope global --start 21 --superuser
```

Example: Delete the given caslib from the specified server.

```
sas-admin cas caslibs delete --server serverA --name caslibA
```

Details

The CAS CLI supports the following environment variables:

- SAS_CLI_DEFAULT_CAS_SERVER
- SAS_CLI_DEFAULT_CASLIB
- SAS_CLI_DEFAULT_CAS_SESSION

You can assign values to the environment variables that you want to remain in effect throughout your session. If the cas CLI command requires the `server`, `caslib`, or `session-id` options, and the environment variables are set, then you can omit the required options from the CAS CLI command.

For example, suppose the following:

- **SAS_CLI_DEFAULT_CAS_SERVER** is set to `serverA`.
- **SAS_CLI_DEFAULT_CASLIB** is set to `caslibA`.

You can then run this command without specifying the required `server` and `caslib` options: `./sas-admin cas tables show-info --table airlines`.

Note: Some commands do not support the use of some of the environment variables. For example, the CAS CLI ignores the cas environment variables for the following commands:

- `caslib remove-control`
- `caslib delete`
- `tables remove-control`
- `sessions delete`

You must explicitly specify all required options when using these commands.

See Also

- “Command-Line Interface: Overview” on page 182
- *SAS Viya Administration: SAS Cloud Analytic Services*

- [SAS Viya Administration: Identity Management](#)
- [SAS Viya Administration: Data](#)

CLI Examples: Configuration

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions”](#) on page 185.

Examples

Example: List all the configurations that exist in the Configuration service.

```
sas-admin configuration configurations list
```

Example: Download the configurations with the specified definition name (`spring` in this example) from the SASLogon service, and write the output to a file.

```
sas-admin configuration configurations download --definition-name spring --service SASLogon  
--target path-to-output-file
```

Example: List the expectation objects in the Configuration service.

```
sas-admin configuration expectations list
```

Example: Show the expectation with the specified ID.

```
sas-admin configuration expectations show --id 185a046e-4e88-4e29-86cf-61d04b9abd07
```

Details

- You can delete a configuration with the configuration CLI.

CAUTION! Do not use the `delete` subcommand of the `configurations` command unless you are sure that you want to delete your configuration.

See Also

[“Command-Line Interface: Overview”](#) on page 182

CLI Examples: Compute

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions”](#) on page 185.

Examples

Example: List the compute contexts.

```
sas-admin compute contexts list
```

Example: Validate the compute context session with the specified ID.

```
sas-admin compute contexts validate --id 389fee7a-e164-4e45-b836-a301638e9945
```

Example: Delete the compute context session with the specified name.

```
sas-admin compute contexts delete --name "SAS Job Execution compute context"
```

Example: List the launcher contexts.

```
sas-admin compute launchers list
```

Example: Delete the launcher context with the specified ID.

```
sas-admin compute launchers delete --id 8fbdd5f8-a2ee-42a5-a228-8737a0cf778f
```

Example: List the compute sessions.

```
sas-admin compute sessions list
```

See Also

[“Command-Line Interface: Overview” on page 182](#)

CLI Examples: Folders

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Examples

Example: Create folderA. Add subfolderA as a child folder to folderA.

```
1 sas-admin folders create --name folderA
```

```
2 sas-admin folders create --name subfolderA --parent-id parent-folder-ID
```

1 Create folderA and note the ID.

2 Issue the command to create subfolderA and specify the ID of folderA as the parent folder.

Example: List the members of folderA.

```
sas-admin folders list-members --id folderA-ID
```

Example: Update the name of folderA to departmentA.

```
sas-admin folders update --id folderA-ID --name departmentA
```

Example: Delete departmentA.

```
sas-admin folders delete --id departmentA-ID
```

Details

Many of the folders commands require the ID of a folder as an argument. The ID of a folder is displayed when you create the folder. The ID of folders is also displayed when you list folders.

See Also

- [“Command-Line Interface: Overview” on page 182](#)
- [“Content Management: Overview” on page 257](#)

CLI Examples: Fonts

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Examples

Example: Add and register the open-source OpenSans-Bold.ttf Google font.

```
sas-admin fonts add --uri https://server:port/fonts/OpenSans-Bold.ttf
```

Example: List the currently registered fonts in the system.

```
sas-admin fonts list
```

Example: Show information about the Arial Symbol font.

```
1 sas-admin fonts list --name "Arial Symbol"
```

```
2 sas-admin fonts show-info --id font-ID-of-Arial-Symbol-font
```

- 1 Identify the ID of the Arial Symbol font.
- 2 Show information about the Arial Symbol font using the specified font ID.

Example: Delete fontA from the system.

```
1 sas-admin fonts list --name fontA
```

```
2 sas-admin fonts delete --file-id file-id of fontA
```

- 1 Identify the file ID of fontA.
- 2 Delete fontA from the system using the specified file ID.

Details

- The fonts CLI can be used to add Web Open Font Format (WOFF), TrueType (TTF), and TrueType Collection (TTC) fonts to the Fonts service. Once added, the fonts are available for the following purposes:
 - To render content in web browsers with Web Open Font Format (WOFF) and TrueType (TTF) fonts.

TrueType Collection (TTC) fonts are not displayed in web browsers. Therefore, users cannot select text that uses TrueType Collection (TTC) fonts if they are included in a SAS web application such as SAS Visual Analytics.
 - To render contents for printing with TrueType (TTF) and TrueType Collection (TTC) fonts.

Web Open Font Format (WOFF) fonts are not supported for printing PDF or SVG output. Therefore, WOFF fonts must be paired with a matching TTF or TTC font for print support.

Note: The following SAS system fonts are WOFF only, and are not supported for printing:

- Noto Sans
- Noto Sans JP
- Noto Sans KR
- Noto Sans SC
- Noto Sans TC

- Noto Sans Thai

Given the TTC and WOFF font limitations, you might need to upload a combination of font formats in order to satisfy both of the preceding purposes.

- A font might not be available from a web server that is accessible from the middle tier without authentication. If so, you can upload the font to the SAS Viya server and reference it with the file:/// URI scheme.
- In multi-tenant environments, the fonts are shared across all the tenants. Maintenance is supported only on the provider tenant.
- You are responsible for obtaining licensing of any fonts that are registered with the system. In a multi-tenancy configuration, this includes licensing the fonts for use by all tenants.

See Also

[“Command-Line Interface: Overview” on page 182](#)

CLI Examples: Device Management

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Examples

Example: Determine whether the device with ID device1 is authorized for use in the environment.

```
sas-admin devices authorized-devices validate --device-id device1
```

Example: Show device enforcement status.

```
sas-admin devices enforcement status
```

Example: Add the device with ID device1 to the whitelist.

```
sas-admin devices whitelist add --device-id device1
```

Example: List the devices that are enabled in the whitelist.

```
sas-admin devices whitelist list
```

Example: Add the device with ID device1 to the blacklist.

```
sas-admin devices blacklist add --device-id device1
```

Example: Remove the device with ID device1 from the blacklist.

```
sas-admin devices blacklist delete --device-id device1
```

Example: From a list of the devices of type iPhone that have connected or attempted to connect to the server, add a specific iPhone to the blacklist.

```
1 sas-admin devices last-access list --device-type iPhone
```

```
2 sas-admin devices blacklist add --device-id device-id
```

1 List the last-access attempts of all devices of type iPhone.

2 Add a device to the blacklist using the device ID that was identified in the previous step.

Example: List in fulljson output the last access attempts to the server for all devices.

```
sas-admin --output fulljson devices last-access list
```

See Also

- [“Command-Line Interface: Overview” on page 182](#)
- [SAS Viya Administration: Mobile](#)

CLI Examples: Identities

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Examples

Example: Add user1 to the group that has the ID 4444.

```
sas-admin identities add-member --user-member-id user1 --group-id 4444
```

Example: Create a group with the name Salesgroup, the group ID 8888, and the description of “Custom sales group”.

```
sas-admin identities create-group --id 8888 --name Salesgroup --description "Custom sales group"
```

Example: Remove user1 from the sales and marketing groups.

```
1 sas-admin identities list-memberships --user-id user1
2 sas-admin identities remove-member --group salesgroup --user-member-id user1
3 sas-admin identities remove-member --group marketinggroup --user-member-id user1
```

- 1 Verify the groups that user1 belongs to.
- 2 Remove user1 from the group that has the group ID sales-group.
- 3 Remove user1 from the group that has the group ID marketing-group.

Example: Show details about the group that has the group ID ABC.

```
sas-admin identities show-group --id ABC
```

Details

- If a group is created with no name, the specified ID will be used for the name.
- The following identities commands list only 50 items at a time by default:
 - list-groups
 - list-members
 - list-memberships

To list more than 50 items, you can use the `--limit` option.

Example:

```
sas-admin identities list-members --group-id ABCD --limit 100
```

See Also

- [“Command-Line Interface: Overview” on page 182](#)
- [“Identity Management Overview” on page 475](#)

CLI Examples: Licensing

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Examples

Example: List site information, which is the site name, the site number, the operating system name, the release number, the server date, the grace period, and the warning period.

```
sas-admin licenses site-info list
```

Example: List all products in the system whose name contains “Visual Analytics”.

```
sas-admin licenses products list --name-contains "Visual Analytics"
```

Example: List all products in the system that are expired.

```
sas-admin licenses products list --expired
```

Example: List the products in the system that have these identifiers: 827, 921, and 985.

```
sas-admin licenses products list --product-ids 827,921,985
```

Example: List the number of products with a current license that are deployed.

```
sas-admin licenses count --current
```

Example: List the Data-Connector products whose licenses are covered within the grace period.

```
sas-admin licenses data-connectors list --grace
```

See Also

- [“Command-Line Interface: Overview” on page 182](#)
- [SAS Viya Administration: Licensing](#)

CLI Examples: Job

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Examples

Example: List the job flows.

```
sas-admin job flows list
```

Example: Generate a template for a flow and write the output to a file.

```
sas-admin job flows generate-template --file-out /tmp/output.txt
```

Example: List job flow scheduling service objects.

```
sas-admin job schedulers list
```


Details

Here is an example of a template file that is generated from a flow that you created:

```
{
  "name": "Replace with name of the flow",
  "description": "(Optional) Replace with description of the flow",
  "triggerType": "Replace with trigger type, select one of: runnow, manual, event",
  "triggerCondition": "Replace with either any or all",
  "flowProperties": {},
  "defaultJobProperties": {}
}
```

See Also

[“Command-Line Interface: Overview” on page 182](#)

CLI Examples: Reports

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Examples

Example: Show information about the report that has the ID a85235e7-fad1-4f8a-9ad9-ea0d576619e1.

```
sas-admin reports show-info --id a85235e7-fad1-4f8a-9ad9-ea0d576619e1
```

Example: List the detailed output of all reports that were created after 2017-05-23.

```
sas-admin reports list --created-after 2017-05-23 --details
```

Example: List the reports in the system that were modified by user1.

```
sas-admin reports list --modified-by user1
```

Example: List a maximum of 50 reports that are sorted by ID and in descending order.

```
sas-admin reports list --details --sort-by ~id --limit 50
```

Example: Delete the report that has the ID a85235e7-fad1-4f8a-9ad9-ea0d576619e1.

```
sas-admin reports delete --id a85235e7-fad1-4f8a-9ad9-ea0d576619e1
```

Details

- The reports CLI lists only 20 reports at a time by default. To list more than 20 reports, you can use the `--limit` option. To list 50 reports, enter `--limit 50`.
- To specify what report number to start the list with, you can use the `--start` option. Suppose that you have listed the first 20 reports, and you want to list the next 20 reports, starting with report number 21, enter `--start 21`.

See Also

[“Command-Line Interface: Overview” on page 182](#)

CLI Examples: Tenant Administration

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Examples

Example: Create the tenant with the ID `companya` and assign a description.

```
sas-admin tenant create --id companya --description "description of companya"
```

Example: Delete the tenant that has the ID `companya`.

```
sas-admin tenant delete --id companya
```

Example: Offboard the tenant that has the ID `companya`.

```
sas-admin tenant offboard --id companya
```

Example: Enable the tenant that has the ID `companya`.

```
sas-admin tenant enable --id companya
```

Example: Disable the tenant that has the ID `companya`.

```
sas-admin tenant disable --id companya
```

Details

- The Help for the ID of a tenant states that the string must match this pattern: `^[a-z]+[a-z0-9]*`
This pattern means that the ID of a tenant must start with a lowercase letter, followed by any number of lowercase letters. Use of numbers is optional.
- When you enable a tenant so that users can sign in to it, the access policy of the tenant is changed from `providerTenantUsersOnly` to `allUsers`.

When you disable a tenant so that users can no longer sign in to it, the access policy of the tenant is changed from `allUsers` to `providerTenantUsersOnly`.

For more information about the access policy of a tenant, see [“Edit the Properties of a Tenant” on page 574](#).

See Also

- [“Command-Line Interface: Overview” on page 182](#)
- [SAS Viya Administration: Multi-tenancy](#)

CLI Examples: Transfer

The following examples assume that you have already signed in to SAS Viya at the command line. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

Examples

Example: See the commands and subcommands that are available for the transfer import command.

```
sas-admin transfer import --help
```

Example: Export a new transfer package from an object that has the resource URI `/reports/reports/faa7f5f2-0822-4ca0-9f92-23bda3e02738`, and name the package “Export Report”.

```
sas-admin transfer export --name "Export Report"
--resource-uri "/reports/reports/faa7f5f2-0822-4ca0-9f92-23bda3e02738"
```

Example: Get the mapping information for the transfer package that is named “Export Report” from the previous example.

```
1 sas-admin transfer list --name "Export Report"
2 sas-admin transfer get-mapping --id transfer-package-ID
```

- 1 Locate the ID for the transfer package that you want to export.
- 2 Issue the command to retrieve the mapping information for the “Export Report” transfer package that has the specified ID.

Example: Upload a package source environment to the target environment, and write the mapping file to the specified location.

```
1 sas-admin --profile source transfer download --id transfer-package-ID --file /tmp/MyPackage.json
2 sas-admin --profile target transfer upload --file /tmp/MyPackage.json --mapping /tmp/map.txt
```

- 1 Download the package to your local machine and store it in a package file that is named `MyPackage.json`.
- 2 Upload the `MyPackage.json` file to the target environment, and write the mapping file to the file `map.txt`.

Details

You can specify information about the export or import operation that you want to perform using the Transfer service REST API standards, as follows:

■ export

You can specify information about the export operation that you want to perform using the `request` option of the `transfer export` command. The option accepts JSON input of type `ExportRequest`. The content of the input can be contained in a quoted string or in a file. The filename must begin with the at sign (`@`). The filename can be specified in either of two forms:

```
@filename.txt
```

```
@/path/.filename.txt
```

The content of this option makes up the POST request through which the export package is sent.

Table A.12 HTTP POST Export Request Members

Name	Type	Description
version	integer	The schema version number of the JSON media type. This is version 1.
name	string	The name of the export job that is used to export objects from a source system to a transfer package. This is also the name of the transfer package that is being created.
description	string	A short description of the export job.
items	list	The list of URIs to include in the transfer package.

Here is a sample JSON file of type ExportRequest. The name of the file is export.json.

```
{
  "version": 1,
  "name": "My reports",
  "description": "Export of all my reports",
  "items": [
    "/reports/reports/4d083692-3c9a-4f2c-945f-e96fad972036",
    "/folders/folders/d4f1533a-229d-4d45-a5c9-b8a21fbc1e39"
  ]
}
```

You can use either of the following syntax options of the command to export the information:

□ `sas-admin transfer export --request @/path-to-file/export.json`

Note: When you include the information in a file, the first character following the request option must be the at sign (@). Therefore, if a pathname is used, it must start with the at sign (@).

□ `sas-admin transfer export --request '{ "version": 1, "name": "My reports", "description": "Export of all my reports", "items": ["/reports/reports/4d083692-3c9a-4f2c-945f-e96fad972036", "/folders/folders/d4f1533a-229d-4d45-a5c9-b8a21fbc1e39"] }'`

■ import

You can specify information about the import operation that you want to perform using the request option of the transfer import command. The option accepts JSON input of type ImportRequest. The content can be contained in a quoted string or in a file. The filename must begin with an at sign (@). The filename can be specified in either of two forms:

`@filename.txt`

`@/path/.filename.txt`

The content of this option makes up the POST request through which the import package is sent.

Table A.13 HTTP POST Import Request Members

Name	Type	Description
version	integer	The schema version number of the JSON media type. This is version 1.
name	string	The name of the import job that is used to import objects from a transfer package to a target system.
description	string	A short description of the import job.
packageUri	string	The package to import.

Here is an example of a JSON file of type ImportRequest that is named import.json:

```
{
  "version": 1,
  "name": "My reports",
  "description": "import all my reports",
  "packageUri": "/transfer/packages/a2ef940e-14ac-4960-9a9a-7689702b06f0"
}
```

You can use either of the following syntax options of the command to import the information:

□ `sas-admin transfer import --request @/path-to-file/import.json`

Note: When you include the information in a file, the first character following the request option must be the at sign (@). Therefore, if a pathname is used, it must start with the at sign (@).

□ `sas-admin transfer import --request '{ "version": 1, "name": "My reports", "description" : "import all my reports ", "packageUri": "/transfer/packages/a2ef940e-14ac-4960-9a9a-7689702b06f0" }'`

See Also

- [“How To” on page 263](#)
- [“Command-Line Interface: Overview” on page 182](#)

Configuration Properties

Configuration Properties: Overview

You manage configuration properties for SAS Viya services using the Configuration pages in SAS Environment Manager.

Note: A [programming-only deployment on page 1](#) does not use SAS Viya services and SAS Environment Manager.

The exception is SAS Studio. You manage SAS Studio configuration properties by modifying its configuration file.

See:

- [“Introduction” on page 215](#)
- [“Configuration Properties: How To Configure SAS Studio” on page 217](#)

Configuration Properties: How To Configure Services

Introduction

These instructions explain how to view and modify service configuration properties using [SAS Environment Manager](#).

Navigation

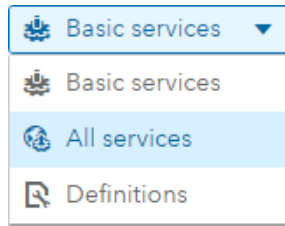
In the applications menu () , select **Administration** ⇒ **Manage Environment**. In the navigation bar, click  .


The Configuration page is an advanced interface. It is available to only SAS Administrators.

Edit Configuration Instances

Note: Most SAS Viya applications and servers have a corresponding service in which you set their configuration property values.

- 1 Using the select control, choose **All services**.

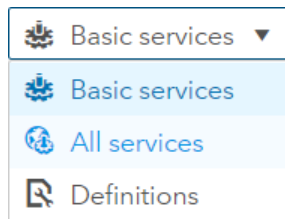



- 2 In the navigation pane, select a service whose configuration properties you want to change.
- 3 Next to the [configuration instance](#), click .
- 4 In the Edit Configuration dialog box, change the value in one or more of the configuration property fields.
- 5 When you are finished, click **Save**.
- 6 On a non-cloud platform, such as native Linux, some services require that you restart them when configuration changes are made. See [“What Services Must Be Restarted?”](#) on page 220.

Create Configuration Instances




In some situations, you might decide to create a configuration instance. For example, if you want to configure a logging level for a service that is not already associated with logging.level, you need to create a new configuration instance of logging.level for that service.

- 1 Using the select control, choose **All services**.



- 2 Select a service for which you want to create a new configuration instance.
- 3 At the top of the content pane, click .
- 4 In the Select Definition dialog box, select a [configuration definition](#) from which to create a new configuration instance.
- 5 In **Services**, make sure that the service displayed is the one for which you want to create a new definition. If the correct service is displayed, skip to [Step 6](#).

Otherwise, do the following:

- a Next to **Services**, click .
- b In the Choose Services dialog box, highlight the service to which the configuration instance you are creating applies, and click .
- c Remove any services for which you do not want to create a configuration instance, by highlighting the service and clicking .
- d When you are finished, click **OK**.

- 6 Continue entering values. When you are finished, click **Save**.

TIP Properties with a red asterisk (*) are required to have a value.




- 7 On a non-cloud platform, such as native Linux, some services require that you restart them when configuration changes are made. See “[What Services Must Be Restarted?](#)” on page 220

Review Default Configuration Values



- 1 In the top left corner of the window, make sure that **Basic services** is selected.
- 2 In **Basic services** list, select a service, application, or server whose configuration instance must be created.

TIP Incomplete required configuration instances are marked with a half-filled red circle.


 identities  SASLogon

- 3 On the right side of the window, next to the half-filled red circle , click .
- 4 Most configuration definitions apply to only one service. In the New Configuration dialog box, if there is no edit icon () next to the **Services** field, skip to [Step 5](#).

Otherwise, do the following:

- a Next to **Services**, click .
 - b In the Choose Services dialog box, highlight the service to which the configuration instance you are creating applies, and click .
 - c When you are finished, click **OK**.
- 5 Continue entering values. When you are finished, click **Save**.

TIP You are required to provide a value for properties marked with a red asterisk (*).

- 6 Repeat steps 2 – 5 for every configuration instance that is incomplete .

Configuration Properties: How To Configure SAS Studio

Update SAS Studio Configuration Properties

To customize web application configuration properties for SAS Studio 4.0, edit the following file:

```
/opt/sas/viya/config/etc/sasstudio/default/init_usermods.properties
```

Note: For sites that use Ansible: Ansible updates `init_deployment.properties` when it is run. Therefore, SAS Studio configuration changes that you make to `init_usermods.properties` are not overwritten by Ansible and are carried forward.

For a listing of configuration properties that you can update, see [“SAS Studio” on page 249](#).

Changes take effect after you restart the web application. For more information, see [“Operate” on page 668](#).

TIP Values that you specify in the `init_usermods.properties` file have precedence over corresponding values in other files. Unlike values in other files, values in the `init_usermods.properties` survive software upgrades.

Configuring Global Folder Shortcuts

In SAS Studio, you can create folder shortcuts from the Server Files and Folders section in the navigation pane. You might want to create global shortcuts for all the users at your site, so each user does not have to create these shortcuts manually.

- 1 In the `init_usermods.properties` file, specify a directory path for the `webdms.globalSettings` property.

By default, this directory path is `/opt/sas/viya/home/SASFoundation/GlobalStudioSettings`. (If you choose to use this default, you must create the `GlobalStudioSettings` directory.)

- 2 In an XML editor, create a `shortcuts.xml` file.

If you are trying to create a shortcut to a network location, here is the format of the `shortcuts.xml` file:

```
<?xml version="1.0" encoding="UTF-8"?>
<Shortcuts>
<Shortcut type="disk" name="network-location" dir="directory-path"/>
</Shortcuts>
```

- 3 Save the `shortcuts.xml` file to the global settings directory.

Operations

Automate Configuration Properties during Deployment (Ansible)

You can deploy SAS Viya with configuration values that are customized to your site by running your Ansible playbook with `sitedefault.yml`. Using `sitedefault.yml`, enables you to provision multiple machines in the same manner, and prevents you from having to modify configuration values with an administration interface after deployment.

Note: It is extremely important that the initial values applied with `sitedefault.yml` are correct. After you set a value with `sitedefault.yml`, you cannot re-run `sitedefault.yml` to change that value. You can re-run `sitedefault.yml` only to set properties that have not already been set. To change properties set with `sitedefault.yml`, you must use the `sas-bootstrap-config` CLI directly, or use another administration interface, such as SAS Environment Manager.

To set configuration values using `sitedefault.yml`, follow these steps:

- 1 Sign on your Ansible controller with administrator privileges, and locate the file, `/playbook/roles/consul/files/sitedefault_sample.yml`.
- 2 Make a copy of `sitedefault_sample.yml` and name the copy, `sitedefault.yml`.
- 3 Using a text editor, open `sitedefault.yml` and add values that are valid for your site.

For information about the properties used in `sitedefault.yml`, refer to [“sas.identities.providers.ldap” on page 230](#).

CAUTION! Some properties require passwords. If properties with passwords are specified in `sitedefault.yml`, you must secure the file appropriately. If you chose not to supply the properties in `sitedefault.yml`, then you can enter them using SAS Environment Manager. (Sign in to SAS Environment Manager as `sasboot`, and follow the instructions in [“Configure the Connection to Your Identity Provider” in SAS Viya for Linux: Deployment Guide](#).)

- 4 When you are finished, save `sitedefault.yml` and make sure that it resides in the `/playbook/roles/consul/files` directory of the playbook.
- 5 Run your Ansible playbook using the `sitedefault.yml` file.

Here is an example:

```
ansible-playbook site.yml
```

For a complete list of playbook commands, see [“Commands” in SAS Viya for Linux: Deployment Guide](#).

- 6 After the playbook is run, verify that the configuration values are successfully loaded into the configuration server by performing the following steps:
 - a Verify that a copy of `sitedefault.yml` resides in `/viya/config/etc/consul.d/default/`.
 - b Verify that `config-kv-bulkload-sitedefault.json` resides in `/viya/config/etc/consul.d/`.
 - c View the configuration properties for a configuration definition such as, SAS Logon Manager, in SAS Environment Manager to verify that the specified values are present.

For more information, follow the first five steps in [“Edit Configuration Instances” on page 215](#).

Configuration Properties: Concepts

What Is SAS Viya Configuration?

From SAS Environment Manager, you can manage the configuration needs of the various SAS Viya services.

Configuration Components

A service’s configuration consist of the following components:

- *configuration definition*: A schema that describes a type of configuration. You create configuration instances from a configuration definition. Some examples of configuration definitions are: `jvm`, `spring`, and `sas.reportdata`.

Note: Configuration definitions that apply to one or a small set of services are referred to as service configuration definitions. System configuration definitions can apply to any service.

- *configuration instance*: A collection of name-value pairs that a service uses. (These name-value pairs can sometimes be nested.)

Note: Certain configuration instances are required for a service to be able to run. See [“Review Default Configuration Values” on page 217](#).

What Services Must Be Restarted?

On a non-cloud platform, such as native Linux, whenever a change is made to a Java virtual machine (JVM) configuration property (a Java option), any services that rely on that property must be restarted. For information about how to restart one or more services, see [“General Servers and Services: Operate” on page 599](#).

A change to configuration property values for any of the following services, requires a restart of the service:

- SAS Cache Locator
`sudo service sas-viya-cachelocator-default restart`
- SAS Cache Server
`sudo service sas-viya-cacheserver-default restart`
- SAS Configuration Server (Consul)
`sudo service sas-viya-consul-default restart`
- SAS Message Broker (RabbitMQ)
`sudo service sas-viya-rabbitmq-server-default restart`
- SAS Infrastructure Data Server (PostgreSQL)

For information, see [“Operate a Cluster” on page 683](#)

Note: You must be signed in to the machine where these services reside with sudo privileges to run these scripts.

See Also

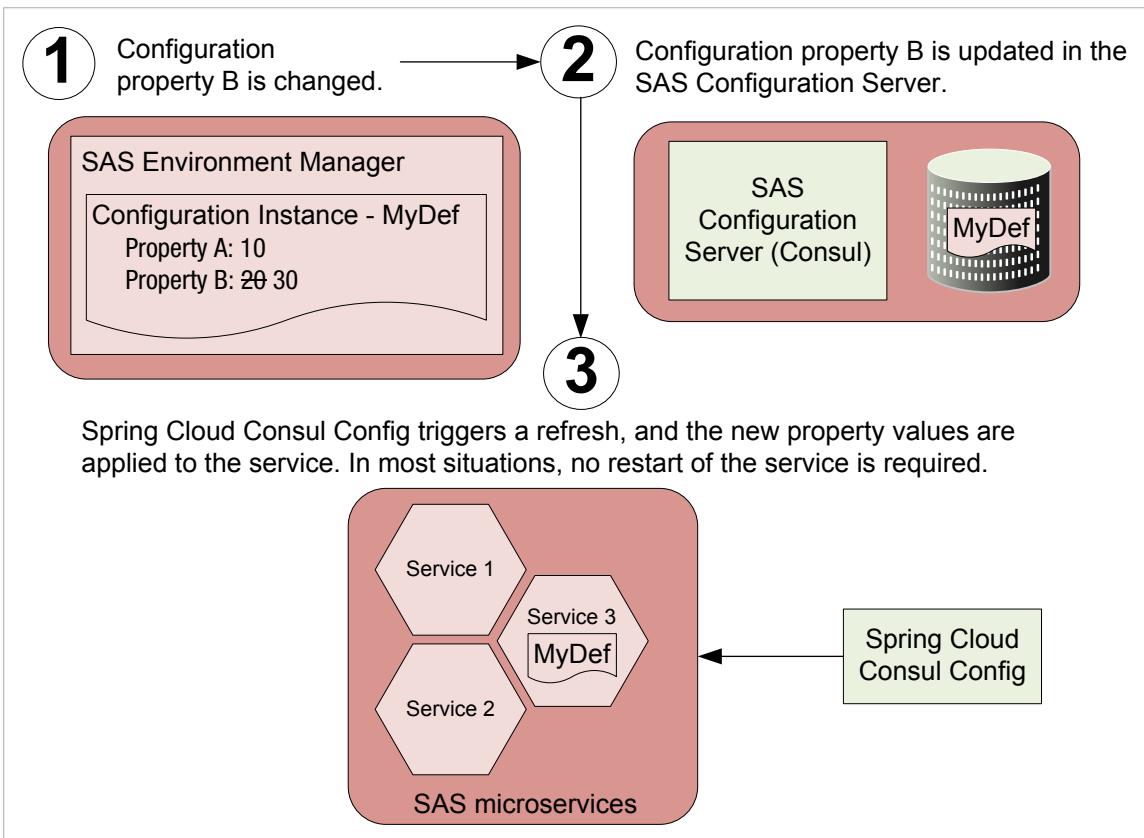
- [“A Particular Server or Service” on page 600](#)

How SAS Viya Configuration Works

Spring-Based Microservices

For SAS Viya, Spring-based microservices, property changes are made in SAS Environment Manager and stored in SAS Configuration Server. SAS Viya triggers a refresh, and the new property values are applied to the service. In most situations, no restart of the service is required.

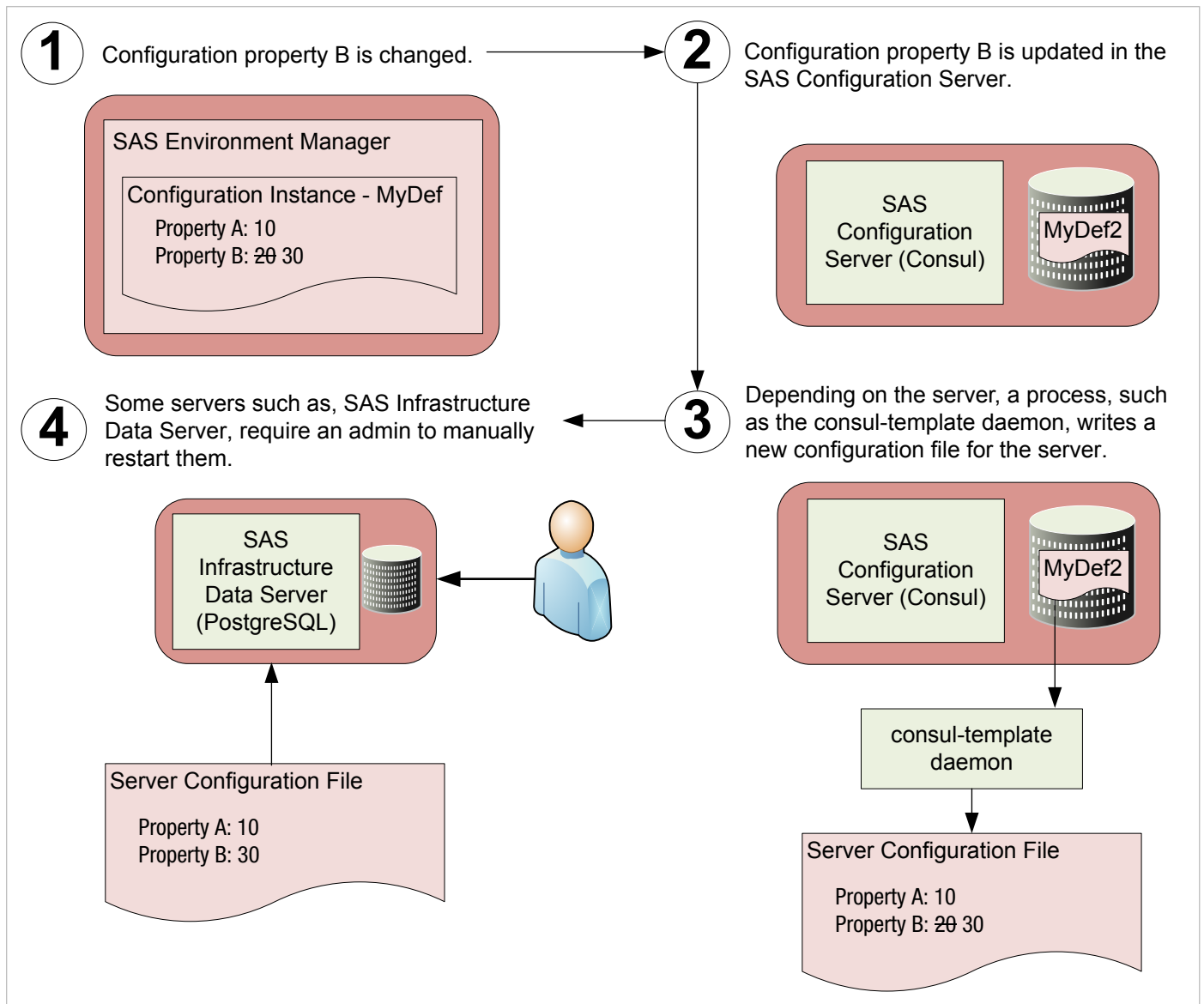
Figure A.1 How Configuration Properties Are Updated (Spring-Based Services)



Non-Spring-Based Servers

For SAS Viya, non-Spring-based servers, property changes are made in SAS Environment Manager and stored in SAS Configuration Server. A tool, such as the consul-template daemon, extracts the configuration change from SAS Configuration Server and updates the appropriate service configuration file. Some servers, such as SAS Infrastructure Data Server, require you to manually restart them for their configuration changes to take effect.

Figure A.2 How Configuration Properties Are Updated (Non-Spring-Based Servers)

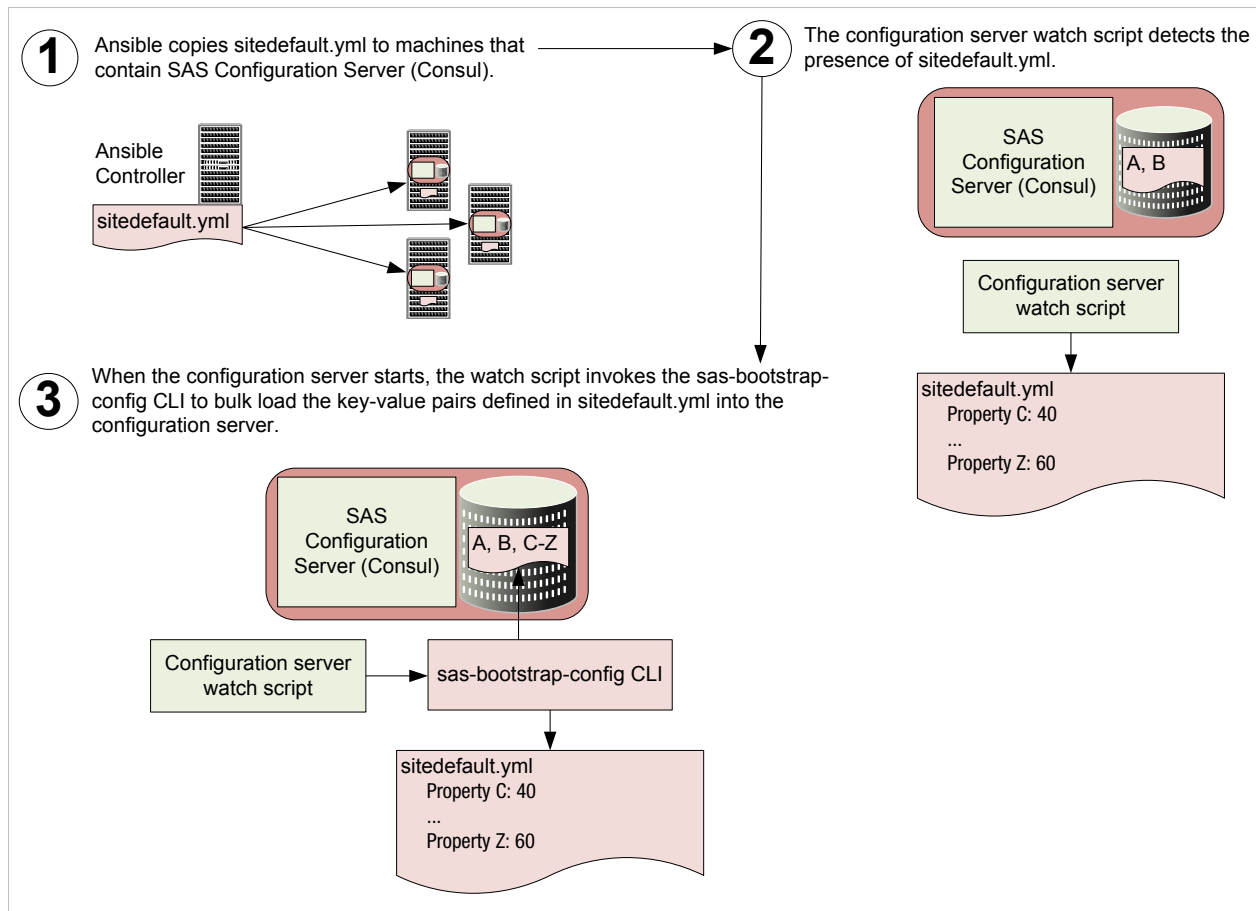


Bulk Loading of Configuration Values (sitedefault.yml)

You can deploy SAS Viya with configuration values that are customized for your site by running your Ansible playbook with `sitedefault.yml`. When `sitedefault.yml` is present in the playbook `roles/consul/files` directory, Ansible copies it to the machines that contain the SAS Configuration Server (Consul). When the configuration server starts, the watch script invokes the `sas-bootstrap-config` CLI to bulk load the key-value pairs that are defined in `sitedefault.yml`.

Note: The `sas-bootstrap-config` CLI uses a *check and set policy*. If a property currently exists in the configuration server, the CLI does not update the property. Therefore, it is extremely important that the initial values applied with `sitedefault.yml` are correct. After you set a value with `sitedefault.yml`, you cannot re-run `sitedefault.yml` to change that value. You can re-run `sitedefault.yml` only to set properties that have not already been set. To change properties set with `sitedefault.yml`, you must use the `sas-bootstrap-config` CLI directly, or use another administration interface, such as [SAS Environment Manager](#).

Figure A.3 How Configuration Properties Are Updated (Non-Spring-Based Services)



Configuration Properties: Reference (Services)

Note: If you are a tenant administrator in a multi-tenant system, you are not able to access all of these configuration instances.

Application Registry Service

The Application Registry service registers applications to enable integration with SAS Home and with the Application Switcher (side menu).

`sas.appregistry`

The set of configuration properties for the Application Registry service.

`supplementalProperties`

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

Audit Service

The Audit service provides a framework for reporting on audit events.

sas.audit.archive

The set of properties that are used to archive audit records.

enabled

Enables archiving of audit records.

batchSize

The number of audit records to process in a single batch during an archive request.

scanSchedule

The schedule that determines when an archive request is initiated.

localRetention

The retention period for persisting audit records within the service.

storageType

The external storage mechanism to use for archiving audit records. Must be set to 'none' or 'local'.

storage.local.destination

The file location to use when 'storageType' is set to 'local'.

sas.audit.record

The set of properties that are used to control how audit events are recorded.

type

Customizable properties to control which type of audit events are recorded.

application

Customizable properties to control which application-specific audit events are recorded.

Authorization Service

The Authorization service provides the general authorization system. See [SAS Viya Administration: General Authorization](#).

sas.authorization

The set of configuration properties for the Authorization service.

maxAncestryCacheSize

Specifies the maximum number of ancestors to cache per object. The default value is 1000. The cache enhances performance in container-based inheritance.

rules.executorThreads (a supplemental property.)

Specifies the number of threads that are available for bulk processing of authorization rules. The default value is 20. Modify this value only if you are directed to do so by SAS Technical Support.

remote (a supplemental property.)

Disables enforcement in the general authorization system, if set to `false`. The default value is `true`. An administrator might temporarily disable authorization if rules that inadvertently prevent access are introduced. Do not disable authorization while the system is available to other users.

Backup Service

The Backup service manages the backup and recovery of configuration information and user-created content in a SAS deployment.

sas.deploymentbackup

The set of configuration properties for the Backup service.

jobTimeout

The number of minutes a backup job or a restore job is allowed to run before the job is marked 'Failed.'

retentionPeriod

The number of days that backups are stored before they are removed from the backup vault.

scheduledBackupAllowed

Allows scheduled backups to run. In this release, the default value is false and cannot be changed.

vaultLocation

The location where all backups are stored. In a multi-machine deployment, the install user must have Read and Write permissions on every machine.

agentType

The type of communication (messaging or SSH) that is used between the Backup service and the Backup agent.

Cache Services

The Cache services provide other microservices the ability to distribute cached data across instances. The Cache services consist of both a Cache Locator and a Cache Server.

sas.cache.default

The set of properties that are used to configure global settings for the Cache services.

mode

Specifies the cache mode. Valid values are 'client' or 'local'. SAS Cache Server requires 'local'.

ackSevereAlertThreshold

The number of seconds the distributed system waits after the ack-wait-threshold for an acknowledgment from a system member before sending a severe alert. A value 0 (zero) disables this feature.

ackWaitThreshold

The number of seconds a distributed message waits for an acknowledgment from a system member before sending an alert.

conserveSockets

Allows sockets to be shared by a system member's threads.

deployWorkingDir

The working directory used when deploying JAR application files to distributed system members. This directory can be local and unique to the member, or it can be a shared resource.

disableAutoReconnect

Disables the ability of a system member to reconnect and re-initialize after it has been forced out of the distributed system.

enableNetworkPartitionDetection

Enables the distributed system to detect and handle splits in the distributed system. Splits are typically caused by a partitioning of the network (split brain) where the distributed system is running.

groups

The list of groups that this system member belongs to. Use a comma to separate group names.

locatorDiscoveryAttempts

The number of service discovery attempts allowed before a registered cache locator is found. A value of 0 (zero) allows for an unlimited number of attempts.

locatorWaitTime

The number of seconds that a system member waits for a locator to join the distributed system.

logLevel

Indicates the lowest diagnostic log level (TRACE, DEBUG, INFO, WARN, ERROR, and FATAL) that is processed. Log events whose levels are below the specified value are ignored.

maxConnections

The maximum number of connections to pool when connected to a cache server.

membershipPortRange

The port range used when selecting ephemeral ports for members of the distributed system. Values are 32768 to 61000.

memberTimeout

The number of milliseconds the distributed system waits before it determines that a system member has timed out.

mode

The mode of operation to use when connecting to the cache servers.

pingInterval

The ping interval for the cache client to check the availability of servers in milliseconds.

retryAttempts

The number of retry attempts for operations if a time-out or exception occurs.

tcpPort

The TCP port a member of the distributed system listens on for cache communications.

Cache Locator Service

The Cache Locator service provides discovery information to SAS Viya microservices for the purpose of forming a distributed data cache. SAS Cache Locator is based on the open-source Apache Geode project.

sas.cache.locator

The set of properties that provide customization for the Cache Locator service.

host

The host where the service is deployed.

hostnameForClients

The external host name of the cache locator if different from the local bind address or host name.

port

The port registered for the cachelocator-listener.

retryCount

The number of attempts the service makes to register the cachelocator-listener.

retryPeriod

The amount of time between registration attempts for the cachelocator-listener.

timeout

The amount of time this service waits to start the locator process.

timeoutInterval

The amount of time between attempts checking for the start of the locator process.

Cache Server Service

The Cache Server service hosts long-lived data regions (a cache) and serves the contents to SAS Viya microservices. Like SAS Cache Locator, SAS Cache Server is based on the open-source Apache Geode project.

sas.cache.config

The set of properties that provide customization for caching.

distributedCache

Specifies that the cache should be distributed. Valid values are `true` or `false`. SAS Cache Server requires the value `true`.

sas.cache.server

The set of properties that provide customization for the cache server.

autoStartup

Specifies whether the cache server should be started automatically on start-up.

host

The host where the service is deployed.

hostnameForClients

The external host name of the cache server if different from the local bind address or host name.

maxTimeBetweenPings

The maximum time in milliseconds between messages or a ping from a cache client.

port

The port registered for the cache server.

CAS Management Service

The CAS Management service provides access to shared data for users and applications. The service also provides information about the SAS Viya system for operations such as monitoring and auditing.

sas.casmanagement

The set of properties that are used to configure private settings for the CAS Management service.

supplementalProperties

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

sas.casmanagement.global

The set of properties that are used to configure global settings for the CAS Management service.

The `sas.casmanagement.global` configuration instance applies to all SAS Viya servers and services (global).

- The set of properties used by applications to access shared data and analytics, such as map data.

application.casServer

The name of the CAS server used for application work.

application.caslib

The name of the caslib used for application data.

- The set of properties used to identify the default CAS server for users.

default.casServer

The name of the default CAS server for users.

default.caslib

The name of the default caslib for users.

- The set of properties used by the system for data produced during normal operation, such as audit records and monitoring data.

system.casServer

The name of the system CAS server.

system.caslib

The name of the system caslib.

CAS Proxy Service

The set of configuration properties for the CAS Proxy service.

jobExecutionProvider

Configurable values for the CAS language (CASL) job execution provider job.

casJESExpiresAfter

The amount of time after job completion before the job execution service deletes the job. Specify time in W3C XML duration format (for example, PT5M = 5 minutes). A null value indicates that job is not deleted.

casJESHeartbeatInterval

The heartbeat value (in seconds) to use with a CASL job execution provider job. A zero or negative value indicates that the heartbeat is not checked.

supplementalProperties

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

Collections Service

The Collections service enables access to personal and shared collections.

sas.collections

The set of configuration properties for the Collections service.

supplementalProperties

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

Configuration Service

The Configuration service manages changes to the configuration of services. See [“Configuration Properties: Concepts” on page 219](#).

sas.configuration

The set of configuration properties for the Configuration service.

forceWrite.enabled

Enables writing to the persistence layer for every operation even when that operation made no changes.

locking.enabled

Enables locking between multiple instances of the SAS Configuration Service. Locking must be enabled when more than one SAS Configuration Service instance is present in the deployment.

supplementalProperties

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

Cross Domain Proxy Service

The Cross Domain Proxy service provides access to external web resources over HTTP.

sas.crossdomainproxy

The set of configuration properties for the Cross Domain Proxy service.

allowedDomains

The list of domains (a whitelist) that the cross-domain proxy is allowed to access. The value is a Java regular expression. Use the Or character (|) to separate multiple domains (for example, `https?://*\.sas\.com(:\d+)?/|https?://*\.foo\.bar\.org/`).

TIP SAS recommends that you escape dot (.) characters in regular expressions with a slash (\) (for example, `*\.sas\.com`). Also, add a trailing forward slash (/) with each domain (for example, `*\.sas\.com/`).

allowSystemDomains

Enables the list of trusted domains (if any) required by SAS. If the cross-domain proxy is denied access to one or more of these domains, certain SAS features are disabled. (This list of trusted domains is displayed in the property description in SAS Environment Manager.)

sas.crossdomainproxy.system

The set of system configuration properties for the Cross Domain Proxy service.

excludeRequestHeaders

The list of header names (a blacklist) which the cross-domain proxy excludes from requests sent to a destination URL. The value is a Java regular expression. Use the Or character (|) to separate multiple header names (for example, `cookie|Authorization`).

maxPooledConnectionsPerRoute

The maximum number of pooled connections per route. (This value must be a positive integer.)

maxPooledConnections

The maximum number of total pooled connections. (This value must be a positive integer.)

connectionTimeoutInMinutes

The number of minutes allowed before the connection to the HTTP client times out. A value of zero (0) specifies no time-out.

Device Management Service

The Device Management service provides the means to maintain the server's device blacklist and whitelist tables, including controlling which security model is in place. See [SAS Viya Administration: Mobile](#).

sas.devicemanagement

The set of configuration properties for the Device Management service.

offlineLimitDays

The number of days before the mobile application goes off-line.

passcodeAttempts

The number of passcode attempts before the user is locked out of the mobile application.

passcodeTimeoutMinutes

The number of minutes before the passcode expires on the mobile application.

whitelistSupportEnabled

Enables whitelist support for mobile device security on the server.

Home Service

The Home service supports the functionality of the SAS Home application.

`sas.homeservice`

The set of configuration properties for the Home service.

`supplementalProperties`

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

Identities Service

The Identities service retrieves information about identities (users or groups) from your identity provider. It also enables the creation and management of custom groups. For detailed information about this functionality, see [SAS Viya Administration: Identity Management](#). Here are the configuration properties for the Identities service:

`sas.identities`

The set of properties that are used to configure global settings for the Identities service.

`cache.enabled`

Enables identities information to be cached. Caching is enabled by default.

`cache.providerPageLimit`

The number of identities to process in a given request when loading the cache. The default value is 1000.

`cache.cacheRefreshInterval`

The refresh interval for the identities cache.

Note: Do not set `cache.cacheRefreshInterval` below 20 minutes. Doing so might have a significant impact on your overall system, especially on the LDAP and SAS Infrastructure Data (PostgreSQL) servers.

Use the following conventions to specify the unit of time for the refresh interval:

- d - refers to days (for example, 6d).
- h - refers to hours (for example, 6h).
- m - refers to minutes (for example, 6m).
- s - refers to seconds (for example, 6s).
- ms - refers to milliseconds (for example, 6ms).

`defaultProvider`

The default provider. The default value is `loca1`. (For this release, SAS recommends that you do not change this value.)

`sas.identities.providers.ldap`

The set of properties that are used to configure your LDAP provider.

`membershipCacheRefreshInterval`

Specifies the interval that is used to refresh the membership cache for the LDAP provider. The default value is `6h`.

pagedResults

Enables the LDAP server to use pagination when processing requests. Pagination is enabled by default.

pageSize

The number of identity requests per page to be processed by the LDAP server (if pagination is enabled). The default value is 500.

sas.identities.providers.ldap.connection

The set of properties that are used to configure your LDAP provider.

host

The host name of the LDAP server to connect to.

password

The password for logging on to the LDAP server. If **anonymousBind** is enabled, specify a value of **none**.

pool.enabled

Enables pooling of LDAP connections. Pooling is enabled by default.

pool.evictionTimePeriodMillis

The number of milliseconds that the idle-object evictor thread sleeps between runs. If the value is non-positive, the idle object evictor thread does not run. The default value is 240000.

pool.idleTimeMillis

The minimum amount of time in milliseconds that objects can sit idle in the pool before becoming eligible for eviction by the idle-object evictor, if present. The default value is 480000.

pool.maxActive

The maximum number of active connections of a given type (either Read-Only or Read-Write) that can be allocated from the pool at the same time. The default value is 8.

pool.maxIdle

The maximum number of active connections of a given type (either Read-Only or Read-Write) that can remain idle in the pool without extra connections being released. For no limit, specify a non-positive value. The default value is 8.

pool.whenExhaustedAction

An integer that indicates the behavior when the pool is exhausted. Valid values are: 0 (fail), 1 (block), or 2 (grow).

pool.testOnReturn

Enables validation of objects before they are returned to the pool. This option is disabled by default.

pool.maxSize

The maximum number of active connections of all types that can be allocated from the pool at the same time. For no limit, specify a non-positive value. The default value is -1.

pool.minIdle

The minimum number of active connections of a given type (either Read-Only or Read-Write) that can remain idle in the pool without extra connections being created. To create no extra connections, specify zero. The default value is 0.

pool.testOnBorrow

Enables validation of objects before they are borrowed from the pool. If an object fails validation, it is dropped from the pool and an attempt is made to borrow another object. This option is enabled by default.

pool.maxWait

The maximum amount of time in milliseconds that the pool waits for a connection to be returned before throwing an exception. For no limit, specify a non-positive value.

pool.testWhileIdle

Enables validation of objects by the idle-object evictor, if present. If an object fails validation, it is dropped from the pool. This option is enabled by default.

port

The port for connecting to LDAP.

Note: When connecting via LDAP, port values are set to 389. When connecting via Lightweight Directory Access Protocol over TLS (LDAPS), port values are set to 636.

url

The URL for connecting to LDAP.

The default is: `url: ldap://${sas.identities.providers.ldap.connection.host}:${sas.identities.providers.ldap.connection.port}`

When the host and port properties have been specified, the `url` must be changed if you are connecting via the LDAPS protocol.

userDN

The distinguished name (DN) of the user account for logging on to the LDAP server (for example, `cn=AdminUser, dn=example, dn=com`). If **anonymousBind** is enabled, specify a value of `none`.

anonymousBind

Defines whether Read-Only operations are performed using an anonymous (unauthenticated) context.

sas.identities.providers.ldap.group (Field Mappings)

The set of properties that are used to configure the mapping of group-level fields in your LDAP provider to group-level fields in SAS. For each of the following SAS fields, you specify the corresponding field in your LDAP provider. The default values are valid for most implementations of Microsoft Active Directory. For other LDAP providers, you must provide different values for some fields.

Property	Description	Default (valid for most implementations of Microsoft Active Directory)
<code>accountId</code>	The field in the LDAP provider that is used to populate the group's ID.	<code>sAMAccountName</code>
<code>createdDate</code>	The field in the LDAP provider that is used to populate the group's account created date.	<code>whenCreated</code>
<code>description</code>	The field in the LDAP provider that is used to populate the group's description.	<code>description</code>
<code>member*</code>	The field in the LDAP provider that is used to populate the members of the group.	<code>member</code>
<code>memberOf</code>	The field in the LDAP provider that is used to populate the groups that this group is a member of.	<code>memberOf</code>
<code>modifiedDate</code>	The field in the LDAP provider that is used to populate the date on which the group's account was last modified.	<code>whenChanged</code>
<code>name</code>	The field in the LDAP provider that is used to populate the group's name.	<code>displayName</code>

Property	Description	Default (valid for most implementations of Microsoft Active Directory)
objectClass	The object class value to use when loading groups.	group

* For group membership to work, the Identities service requires that the LDAP attribute be a fully qualified DN, not simply a UID (for example, **member: uid=user1,ou=users,dc=example,dc=com**).

sas.identities.providers.ldap.group (Additional Properties)

The set of properties that are used to configure information for retrieving group information from your LDAP provider.

Note: The Identities service does not process referrals.

baseDN

The point from which the LDAP server searches for groups (for example, `ou=groups,dc=example,dc=com`).

distinguishedName

The field in the LDAP provider that is used to populate the group's distinguished name value.

Note: If your LDAP server does not support an explicit distinguished name attribute (for example, OpenLDAP), you must set this property to `none`.

objectFilter

The filter for customizing results that are returned when groups are queried [for example, `(objectClass=group)`].

You can create a custom filter to exclude identities whose accounts are disabled or expired, or to exclude objects that represent computer resources rather than actual groups. If you have a large number of groups, using a custom filter can improve performance and reduce memory requirements. In addition, user management tasks can be performed more efficiently if only relevant identities are listed in SAS Environment Manager.

searchFilter

The filter that is used to find a group account. The default filter is `${sas.identities.providers.ldap.group.accountId}={0}`.

sas.identities.providers.ldap.user (Field Mappings)

The following properties specify the mapping of user-level fields in your LDAP provider to user-level fields in SAS. For each of the following SAS fields, you specify the corresponding field in your LDAP provider. The default values are valid for most implementations of Microsoft Active Directory. For other LDAP providers, you must provide different values for some fields.

Property	Description	Default (valid for most implementations of Microsoft Active Directory)
accountId	The field in the LDAP provider that is used to populate the user's ID.	sAMAccountName
address.country	The field in the LDAP provider that is used to populate the user's country.	co

Property	Description	Default (valid for most implementations of Microsoft Active Directory)
address.locality	The field in the LDAP provider that is used to populate the user's city.	l
address.postalCode	The field in the LDAP provider that is used to populate the user's postal code.	postalCode
address.region	The field in the LDAP provider that is used to populate the user's region or state.	region
address.street	The field in the LDAP provider that is used to populate the user's street address.	street
createdDate	The field in the LDAP provider that is used to populate the user's account created date.	whenCreated
description	The field in the LDAP provider that is used to populate the user's description.	description
emailAddress.other	The field in the LDAP provider that is used to populate the user's alternate email address.	otherMailbox
emailAddress.work	The field in the LDAP provider that is used to populate the user's work email address.	mail
emailAddress.sms	The field in the LDAP provider that is used to populate the user's SMS email address.	
memberOf	The field in the LDAP provider that is used to populate the groups that this user is a member of.	memberOf
modifiedDate	The field in the LDAP provider that is used to populate the date on which the user's account was last modified.	whenChanged
name	The field in the LDAP provider that is used to populate the user's name.	displayName
objectClass	The type of user objects that are being searched for.	organizationalPerson
phone.business	The field in the LDAP provider that is used to populate the user's work phone number.	telephoneNumber
phone.businessFax	The field in the LDAP provider that is used to populate the user's work fax number.	facsimileTelephoneNumber
phone.home	The field in the LDAP provider that is used to populate the user's home phone number.	homePhone
phone.mobile	The field in the LDAP provider that is used to populate the user's mobile phone number.	mobile
phone.pager	The field in the LDAP provider that is used to populate the user's pager number.	pager

Property	Description	Default (valid for most implementations of Microsoft Active Directory)
title	The field in the LDAP provider that is used to populate the user's title.	title

sas.identities.providers.ldap.tenancy

The set of specific, multi-tenancy properties that can be used when implementing an LDAP provider.

groupRdn

The relative distinguished name group (RDN) value.

tenantKey

The default value (OU) for tenantKey.

userRdn

The relative distinguished name user (RDN) value.

sas.identities.providers.ldap.user (Other Properties)

The set of properties that are used to configure additional information for retrieving user information from your LDAP provider.

Note: The Identities service does not process referrals.

baseDN

The point from which the LDAP server searches for users.

distinguishedName

The field in the LDAP provider that is used to populate the user's distinguished name value.

Note: If your LDAP server does not support an explicit distinguished name attribute (for example, OpenLDAP), you must set this property to `none`.

objectFilter

The filter for customizing results that are returned when querying users.

You can create a custom filter to exclude identities whose accounts are disabled or expired, or to exclude objects that represent computer resources rather than actual users. If you have a large number of users, using a custom filter can improve performance and reduce memory requirements. In addition, user management tasks can be performed more efficiently if only relevant identities are listed in SAS Environment Manager.

Here is an example of a filter that excludes identities that represent computers and identities that are inactive. This filter is compatible with Microsoft Active Directory.

```
(&(objectCategory=person)(objectClass=user)(!(userAccountControl:1.2.840.113556.1.4.803:=2)))
```

For OpenLDAP, the filter `(objectclass=person)` excludes identities that represent resources other than users.

searchFilter

The filter used for locating a user account in the LDAP provider so that the user can make a connection using an ID and password.

The default filter is `${sas.identities.providers.ldap.user.accountId}={0}`.

Mail Service

The Mail service provides a client the ability to send email to a configured SMTP server using a REST API.

sas.mail

The set of configuration properties for the Mail service.

allowAllSenders

Provides the ability to override restriction on the 'from' mail address allowed to send mail.

fromAddress

Default 'from' mail address to use when mail is sent directly from a service. (The default is `noreplies@company.com`.)

fromPersonalName

Default personal name to use when mail is sent directly from a service. (The default is `Service`.)

host

The mail server host (machine).

password

The optional password for connecting to the mail server.

port

The mail server port. (The default is 25.)

properties

Optional properties set on the remote mail server.

sizeLimit

The maximum size of mail sent to the configured mail server (in megabytes).

username

The optional user name for connecting to the mail server.

Maps Service

The Maps service returns polygon information for selected identifiers from a given table.

sas.maps

The set of configuration properties for the Maps service.

defaultOSMCommunicationProtocol

The protocol (HTTP, HTTPS) that is used for the default Open Street Map servers.

localEsriServicesRequiresAuthentication

Indicates that the local Esri map services URL requires an authentication token for access.

localEsriServicesUrl

The URL to the local Esri map services. The URL consists of a protocol, host, port, and path (for example, `http://myserver:6080/arcgis/rest/services/`).

Note: If your on-premises Esri servers use a different network domain than your SAS Viya system, then you must add the necessary map URLs to the whitelist of domains that the cross-domain proxy is allowed to access. For more information, see [“allowedDomains” on page 229](#).

supplementalProperties

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

useArcGISOnlineMaps

Enable access to background maps from the Esri ArcGIS Online catalog.

The set of custom configuration properties for Open Street Map server settings.

customOSM.maxResolution

The maximum resolution in meters per pixel of each map tile. The value must be a decimal number.

customOSM.numResolutions

The number of tile levels configured on the tile servers. The value must be a positive integer.

customOSM.servers

A comma-separated list of servers, with paths to tiles (for example, <http://myhost1.myorg.com/tiles/>, <http://myhost2.myorg.com/tiles/>).

Monitoring Service

The monitoring service provides information about the machines and services in your environment. See [SAS Viya Administration: Monitoring](#).

sas.monitoring

The set of configuration properties for the Monitoring service.

Report Data Service

The Report Data service retrieves data from reports.

sas.reportdata.system

The set of system configuration properties for the Report Data service.

Note: In a multi-tenant configuration, `sas.reportdata.system` properties apply to all tenants.

supplementalProperties

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

executorExpirationIntervalMinutes

The number of minutes before inactive data executor sessions are closed if they have no active queries.

executorForceExpirationIntervalMinutes

The number of minutes before inactive data executor sessions are forced closed even if they have active queries.

resultCacheErrorExpirationSeconds

The number of seconds before the error cases for a report result are removed from the cache.

resultCacheTimeToLiveSeconds

The number of seconds before a report result is removed from the cache.

tempCacheTimeToIdleSeconds

The number of seconds allowed for a client to retrieve temporary files of result data before they are removed from the cache.

xmlParserPoolSize

The number of XML parsers to be instantiated during application start-up.

sas.reportdata.properties

The set of configuration properties for the Report Data service.

supplementalProperties

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

comparisonEpsilon

The number in E notation that is the variability allowed when comparing floating point numbers for equality.

decisionTreePredictorCardinalityLimit

The maximum cardinality of independent variables allowed to run a decision tree.

decisionTreeResponseCardinalityLimit

The maximum cardinality of a dependent variable allowed to run a decision tree.

defaultInteractiveDrillDepth

The number of interactive drill levels included in the offline data for report viewers.

defaultMaxCellsProduced

The maximum number of data cells delivered for each query result to report viewers.

enableResultCache

Enable report result caching.

exportExcelRowLimit

The maximum number of rows allowed for export files formatted for Excel.

exportExcelColumnLimit

The maximum number of columns allowed for export files formatted for Excel.

exportTSVandCSVRowLimit

The maximum number of rows allowed for tab- and comma-separated export files.

exportTSVandCSVColumnLimit

The maximum number of columns allowed for tab- and comma-separated export files.

ignoreMissingValuesInCountDistinct

Ignore missing values in count distinct.

maxTiesToIncludeOnRank

The maximum number of ties allowed for a rank.

modelingClassCardinalityLimit

The maximum number of class values allowed to run on fit models.

modelingGroupByCardinalityLimit

The maximum number of group by values allowed to run on fit models.

socketTimeoutLiveCancellableMillis

The number of milliseconds allowed for executing live, cancellable data queries.

socketTimeoutLiveNonCancellableMillis

The number of milliseconds allowed for executing live, non-cancellable data queries.

socketTimeoutSubscribeMillis

The number of milliseconds allowed for executing subscribe data queries.

A map of the maximum result rows values for the supported visual types.

maxRowsLookup.bubble

The maximum result rows for a bubble visual.

maxRowsLookup.buttonBar

The maximum result rows for a button bar visual.

maxRowsLookup.crossTab

The maximum result rows for a multidimensional table visual.

maxRowsLookup.customContent

The maximum result rows for a custom content.

`maxRowsLookup.dropdown`

The maximum result rows for a drop-down control.

`maxRowsLookup.dualAxisTimeSeries`

The maximum result rows for a dual axis time series visual.

`maxRowsLookup.geoBubble`

The maximum result rows for a geo bubble visual.

`maxRowsLookup.geoHeatmap`

The maximum result rows for a geo heat map visual.

`maxRowsLookup.geoRegion`

The maximum result rows for a geo region visual.

`maxRowsLookup.geoScatter`

The maximum result rows for a geo scatter visual.

`maxRowsLookup.graphDefault`

The maximum result rows for a default graph visual.

`maxRowsLookup.heatbox`

The maximum result rows for a heat box visual.

`maxRowsLookup.heatmap`

The maximum result rows for a heat map visual.

`maxRowsLookup.kpi`

The maximum result rows for a kpi visual.

`maxRowsLookup.list`

The maximum result rows for a list visual.

`maxRowsLookup.listTable`

The maximum result rows for a list table visual.

`maxRowsLookup.scatter`

The maximum result rows for a scatter visual.

`maxRowsLookup.textInput`

The maximum result rows for a text input control.

`maxRowsLookup.timeSeries`

The maximum result rows for a time series visual.

`maxRowsLookup.treeMap`

The maximum result rows for a treemap visual.

`maxRowsLookup.wordCloud`

The maximum result rows for a word cloud visual.

Report Package Service

The Report Package service executes reports to generate corresponding “report packages.” The report package includes the report.xml, CSS style sheets, images, CSV data files, and so on, that are required to render the report.

`sas.reportpackages.system`

The set of system configuration properties for the Report Packages service.

Note: In a multi-tenant configuration, `sas.reportpackages.system` properties apply to all tenants.

`supplementalProperties`

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

backgroundThreadMonitorSecs

The frequency in seconds at which the background thread monitor runs. A value of zero indicates that the monitor is disabled.

packageExpirationTime

The amount of time in seconds after which the report package expires from the cache.

useProxyServiceForExternallImages

Enable the Cross Domain Proxy service to retrieve the external images in the report.

xmlParserPoolSize

The number of XML parsers to be instantiated during application start-up.

`sas.reportpackages.properties`

The set of configuration properties for the Report Packages service.

supplementalProperties

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

highContrastTheme

The name of the theme to be used when reports are displayed in high contrast.

imageDefaultMaxBytes

The maximum number of bytes for an image. Larger images are scaled down, unless 'noscale' is specified in the report.

subscribeConcurrentRequestLimit

The maximum number of report packages that can be generated concurrently per user.

subscribeConcurrentRequestLimitGuest

The maximum number of report packages that can be generated concurrently for the Guest user.

Report Renderer Service

The Report Renderer service creates PDF documents from report packages.

`sas.reportrenderer.system`

The set of system configuration properties for the Report Renderer service.

Note: In a multi-tenant configuration, `sas.reportrenderer.system` properties apply to all tenants.

cacheDuration

The number of seconds allowed before rendered reports are deleted from the cache.

workingDirectory

The working directory used for building rendered reports.

supplementalProperties

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

`sas.reportrenderer.properties`

The set of configuration properties for the Report Renderer service.

timeoutMillis

The number of milliseconds allowed before the rendering process times out.

footerContentFormatted

The HTML formatted footer to be included on each PDF rendered page.

supplementalProperties

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

Secret Manager Service

The Secret Manager service manages certificates generated by and stored by HashiCorp Vault. Vault provides a secure interface to secrets. These connections are secured by a Public Key Infrastructure (PKI) based on HashiCorp Vault, which is configured by SAS. The certificates are all signed by a Vault-generated root CA and intermediate certificate.

sas.vault

The set of configuration properties used to configure SAS Secret Manager (Vault).

certificate_role

Role of the certificates.

certificate_role_allow_any_name

The common name represents the name protected by the TLS certificate. Any name is allowed for the common name. The certificate is valid only if the requested host name matches the certificate common name. It consists of a single host name (for a single-name certificate) or a wildcard name (for a wildcard certificate, *.example.com).

certificate_role_key_bits

The key-length (in bits) of the certificate generated from Vault.

certificate_role_key_type

The certificate encryption algorithm. RSA or Elliptic-Curve (EC) can be specified.

certificate_role_key_usage

A comma-separated list that defines the allowed uses of a generated certificates. You can specify one or all of the following:

`DigitalSignature,KeyAgreement,KeyEncipherment`

certificate_role_max_ttl

The maximum length of time in hours that a Vault-issued certificate lasts before expiring. Must be greater than the `certificate_role_ttl`.

certificate_role_ttl

The default length of time that a Vault-issued certificate lasts before expiring.

intermediate_ca

An intermediate certificate is a subordinate certificate issued by the trusted root specifically to issue end-entity server certificates.

intermediate_ca_common_name

The common name (CN) for the Vault-issued intermediate certificate authority (CA) certificate. For SAS Viya, the name is SAS Viya intermediate CA. The CN identifies the host name associated with the certificate.

intermediate_ca_desc

The name for the Vault-issued intermediate CA certificates and SAS Viya Intermediate CA.

intermediate_ca_key_bits

The key-length (in bits) of the intermediate certificate generated from Vault.

`intermediate_ca_max_ttl`

The maximum length of time in hours that a Vault-issued intermediate certificate lasts before expiring. Must be greater than the `intermediate_ca_ttl`.

`intermediate_ca_ttl`

The default length of time that a Vault-issued intermediate certificate lasts before expiring.

`root_ca`

A public key certificate that identifies a root CA. A root certificate is the top-most certificate of the tree. The private key is used to sign other certificates.

`root_ca_common_name`

The common name for the Vault-issued root CA certificate and SAS Viya Root CA. The CN identifies the host name associated with the certificate.

`root_ca_desc`

The name for the Vault-issued Root CA and SAS Viya Root CA.

`root_ca_key_bits`

The key-length (in bits) of the root certificate generated from Vault.

`root_ca_max_ttl`

The maximum length of time in hours that a Vault-issued root CA certificate lasts before expiring. Must be greater than the `root_ca_ttl` value.

`root_ca_ttl`

The default length of time that a Vault-issued root CA certificate lasts before expiring.

`systems`

The time in hours that secrets and tokens are managed.

`system_max_lease_ttl`

The maximum amount of time (in hours) that Vault-issued secrets and tokens are valid. This value must be larger than the `vault_token_default_lease_ttl` value for the token configuration instance.

`tokens`

The time in hours that secrets and tokens are valid.

`vault_token_default_lease_ttl`

The default length of time (in hours) that Vault-issued tokens are valid.

Note: Changes to this value take effect after running the Ansible renewal playbook.

Configuration Properties: Reference (Applications)

Note: If you are a tenant administrator in a multi-tenant system, you are not able to access all of these configuration instances.

SAS Data Studio

SAS Data Studio provides a way for you to prepare data, including data transformations.

`sas.datastudio`

The set of configuration properties for SAS Data Studio.

`casSessionNumNodes`

The number of nodes on which to start a SAS Data Studio CAS session (0 means all nodes).

casSessionTimeout

The number of seconds to set as the CAS session time-out for sessions that SAS Data Studio creates.

interactiveJobExpiresAfter

The amount of time after an interactive data plan job completes before the job execution service deletes the job. Specify time in W3C XML duration format (for example, PT5M = 5 minutes). A null value indicates that job is not deleted.

saveTableJobExpiresAfter

The amount of time after a data plan job, which saves a table, completes before the job execution service deletes the job. Specify time in W3C XML duration format (for example, PT5M = 5 minutes). A null value indicates that job is not deleted.

SAS Home

Here are the configuration properties for SAS Home:

sas.home

The set of configuration properties for SAS Home.

supplementalProperties

The set of user-added, advanced properties.

Note: Do not enter a property name-value pair unless you are directed to do so by SAS Technical Support.

SAS Infrastructure Data Server

SAS Infrastructure Data Server is based on PostgreSQL 9.1.9 and is configured specifically to support SAS software. See “[Overview](#)” on [page 683](#). Here is the list of SAS Infrastructure Data Server configuration definitions that consist of third-party PostgreSQL and pgpool-II configuration properties.

sas.dataserver.common

The set of properties that are common to a cluster (that is, both to pgpool-II and to PostgreSQL nodes).

For a list of the valid PostgreSQL property names (configuration parameters) and descriptions, see <https://www.postgresql.org/docs/9.1/static/runtime-config.html>.

sas.dataserver.conf

The set of properties that are used to set up the SAS Infrastructure Data Server node database configuration file, `postgresql.conf`.

For a list of the valid PostgreSQL property names (configuration parameters) and descriptions, see <https://www.postgresql.org/docs/9.1/static/runtime-config.html>.

sas.dataserver.hba

The set of properties that are used to set up the SAS Infrastructure Data Server node host-based authentication file, `pg_hba.conf`.

For a list of the valid PostgreSQL property names (authorization records) and descriptions, see <https://www.postgresql.org/docs/9.1/static/auth-pg-hba-conf.html>.

sas.dataserver.pool

The set of properties that are used to set up the pgpool-II node configuration file, `pgpool.conf`.

For a list of the valid pgpool-II property names (key-value pairs) and descriptions, see http://www.pgpool.net/docs/pgpool-II-3.5.4/doc/pgpool-en.html#pgpool_conf.

sas.dataserver.pool.hba

The set of properties that are used to set up the pgpool-II node host-based authentication file, pool_hba.conf.

For a list of the valid pgpool-II property names (key-value pairs) and descriptions, see <http://www.pgpool.net/docs/pgpool-II-3.5.4/doc/pgpool-en.html#hba>.

SAS Logon Manager

SAS Logon Manager provides OAuth2 and OpenID Connect services for authentication, and a user interface for sign-in, sign out, and other related functions. See “[Authentication: Overview](#)” on page 31. Here are the configuration properties for SAS Logon Manager:

cors

The set of properties that are used to configure Cross-Origin Resource Sharing (CORS) security for SAS Logon Manager (SASLogon) only. (Use [sas.common.web.security.cors](#) to configure CORS for other services.)

Note: Modifying one of these property values requires you to restart one or more SAS Viya services. For more information, see “[General Servers and Services: Operate](#)” on page 599.

default.allowed.uris

The comma-separated list of URIs that are allowed by default to be called from another origin. The value can contain regular expressions.

default.allowed.origins

The comma-separated list of origins that are allowed by default. The list can contain regular expressions.

default.allowed.headers

The comma-separated list of HTTP headers that are allowed by default in cross-origin requests.

default.allowed.methods

The comma-separated list of HTTP methods that are allowed by default in cross-origin requests.

default.allowed.credentials

Require that cross-origin requests must be made using credentials.

default.max_age

The maximum number of seconds that the response to the preflight request can be cached without sending another preflight request.

xhr.allowed.uris

The comma-separated list of URIs allowed in XMLHttpRequest (XHR) requests called from another origin. The value can contain regular expressions.

xhr.allowed.origins

The comma-separated list of origins allowed in XHR requests. The value can contain regular expressions.

xhr.allowed.headers

The comma-separated list of HTTP headers allowed in XHR cross-origin requests.

xhr.allowed.methods

The comma-separated list of HTTP methods allowed in XHR cross-origin requests.

xhr.allowed.credentials

Require that XHR cross-origin requests must be made using credentials.

xhr.max_age

The maximum number of seconds that the response to the XHR preflight request can be cached without sending another preflight request.

sas.logon.callback

The set of properties that are used to configure the whitelist of URIs for trusted applications.

allowed.uris

The comma-delimited list of URIs that users can be redirected to after signing in following a time-out or logoff.

sas.logon.custom

The set of properties that are used to provide custom content that is included on the Sign In to SAS page.

login

The URI of the custom content included on the Sign In to SAS page.

logout

The URI of the custom content included on the Sign In to SAS page when users sign out of the system.

timeout

The URI of the custom content included on the time-out page.

sas.logon.groups

The set of properties that are used to customize lookup of group authorities.

assumable

Specifies groups that have an elevated level of access that the user must approve at sign-in in order to assume those groups in a session.

approvalExpirySeconds

When approving or denying access to a third-party application, specifies the number of seconds that the approval or denial should be remembered.

groupLookupRequired

Requires groups to be determined for authentication to succeed.

recursion.enabled

Allows recursive lookups of authorities for groups assigned to users.

requiresRecursion

The comma-separated list of groups that require a recursive lookup to determine externally assigned authorities.

sas.logon.initial

The set of properties that are used to initially configure the system.

Note: Modifying one of these property values requires you to restart one or more SAS Viya services. For more information, see [“General Servers and Services: Operate” on page 599](#).

reset.enabled

Displays a password reset link for the initial user account at start-up.

user

The user name for the initial user account.

passwordResetLifetime

The number of milliseconds for which that the password reset code is valid after restart.

redirectUri

The URI to which the initial user should be redirected after resetting the password.

sas.logon.jwt

The set of properties that are used to configure how JSON web tokens are issued.

signingKey

Either a Base64-encoded RSA private key that is used to digitally sign tokens, or a simple passphrase for HMACs. Enter a value only if you want to override the system-generated RSA private key.

issuer.uri

The URI of the application, for the issuer claim in tokens (for example, <https://example.com/SASLogon>).

claims.exclude

The comma-separated list of claims that should be excluded from the JSON web token.

policy.accessTokenValiditySeconds

The default number of seconds that access tokens are valid for after being issued in the default zone.

policy.refreshTokenValiditySeconds

The default number of seconds that refresh tokens are valid for after being issued in the default zone.

policy.global.accessTokenValiditySeconds

The default number of seconds that access tokens are valid for after being issued in all zones.

policy.global.refreshTokenValiditySeconds

The default number of seconds that refresh tokens are valid for after being issued in all zones.

refresh.restrictGrant

Grant refresh tokens only to clients with a scope of `refresh_token` for offline access.

sas.logon.kerberos

The set of properties that are used to enable sign-ins using Integrated Windows Authentication (IWA).

Note: Modifying one of these property values requires you to restart one or more SAS Viya services. For more information, see [“General Servers and Services: Operate” on page 599](#).

servicePrincipal

The name of the service principal in the keytab.

keyTabLocation

The URL of the keytab file (for example, `file:///opt/sas/viya/conf/etc/my_keytab`).

stripRealmForGss

Removes the `@...` from the end of the user name.

holdOnToGSSContext

Enables Kerberos delegation to SAS Cloud Analytic Services.

debug

Enables the debug mode of the JAAS Kerberos login module.

sas.logon.oauth.providers.external_oauth

The set of OAUTH provider properties that are used to enable sign-ins using an external provider. Modifying one of these property values requires you to restart SAS Logon Manager. For more information, see SAS Viya Administration in SAS Help Center.

authUrl

The URL to the authorization endpoint.

tokenUrl

The URL to the token endpoint.

tokenKey

The HMAC key or RSA public key used to sign tokens.

tokenKeyUrl

The URL to obtain the token key.

emailDomain

The email address domain of users that can sign on with this provider.

issuer

The principal that issued the token, as a case-sensitive string or URI.

linkText

The text that should be displayed on the sign-in page for this provider.

relyingPartyId

The client ID registered in the provider.

relyingPartySecret

The secret registered in the provider for the client ID.

scopes

The comma-delimited list of scopes for the authorization request.

addShadowUserOnLogin

Adds a local shadow user upon successful authentication.

showLinkText

Shows the link text on the sign-in page.

type

Either 'oidc1.0' or 'oauth2.0'.

attributeMappings.user_name

The attribute claim to use as the user name.

sas.logon.pam

The set of properties that are used to enable sign-ins using PAM.

enabled

Enables sign-in using PAM.

serviceName

The service name in the PAM configuration.

sas.logon.provider.guest

The set of properties that are used to configure guest access to the system.

Apply configuration only to this tenant (provider)

When off, the configuration applies to all tenants including the provider. Each tenant can override the configuration from within its own environment.

Note: **Apply configuration only to this tenant (provider)** is only available to provider tenants in a multi-tenant environment.

enabled

Enable anonymous guest access to web applications.

sas.logon.saml

The set of Security Assertion Markup Language (SAML) properties that are used to enable sign-ins using an external identity provider.

Note: Modifying one of these property values requires you to restart one or more SAS Viya services. For more information, see [“General Servers and Services: Operate” on page 599](#).

entityBaseURL

The URL of the application where SAML assertions are accepted, (for example: <https://example.com/SASLogon>).

setProxyParams

Allows the base URL to reside behind an HTTP proxy.

CAUTION! Do not modify setProxyParams. The value should remain off (false).

entityID

The entity ID of the service provider.

serviceProviderKey

The PEM-encoded, RSA private key that is used by the service provider.

serviceProviderKeyPassword

The password for the private key.

serviceProviderCertificate

The PEM-encoded, X.509 certificate that is used by the service provider.

wantAssertionSigned

Specifies that the assertions must be signed.

signatureAlgorithm

The algorithm for SAML signatures. The accepted values are SHA1, SHA256, and SHA512.

signMetadata

Specifies that the local service provider should sign metadata.

signRequest

Specifies that the local service provider should sign SAML requests.

socket.connectionManagerTimeout

The number of milliseconds before the connection pooling times out for HTTP requests for SAML metadata.

socket.soTimeout

The number of milliseconds before the read times out for HTTP requests for SAML metadata.

sas.logon.saml.providers.external_saml

The set of Security Assertion Markup Language (SAML) identity provider properties that are used to enable sign-ins using an external provider.

idpMetadata

The identity provider metadata or the URL to the metadata.

metadataTrustCheck

Specifies that the identity provider certificate must be trusted.

assertionConsumerIndex

The index of the assertion consumer service to use from identity provider metadata. The value must be a positive integer.

nameID

The default name ID format.

linkText

The hyperlink to display on the sign-in page.

addShadowUserOnLogin

Adds a local shadow user upon successful authentication. If set to `false`, users must be pre-created in the database to log on.

skipSslValidation

Skips Transport Layer Security (TLS) validation of the certificate.

showSamlLoginLink

Displays a link to the identity provider on the sign-in page.

sas.logon.sas9

The set of properties that are used to enable sign-ins using SAS 9.4 credentials.

enabled

Enable sign-ins using SAS 9.4 credentials.

baseServicesUrl

The base URL of the SAS 9.4 services.

sas.logon.sessions

The set of properties that are used to configure how concurrent sessions are handled.

maxConcurrentSessions

The maximum number of allowed concurrent sessions. A value of -1 allows an unlimited number of sessions.

rejectNewSessionsIfMaxExceeded

Rejects new sessions if the maximum number of sessions is exceeded. If false, when the maximum number of sessions is reached, an existing session is invalidated to allow a new one to be created.

sas.logon.tenancy

The set of properties that are used to configure multi-tenancy.

bootstrap.enabled

Automatically configure identity zones and LDAP when tenants are onboarded or access policy is changed.

autoUpdateLdapConfiguration

Automatically update all identity zones' LDAP configurations when the provider LDAP configuration is changed.

SAS Studio

For more information, see [“Update SAS Studio Configuration Properties” on page 217](#).

Table A.14 SAS Studio: Configuration Properties

Property	Default Value	Description
sasstudio.appserver.https.keystorefile	(blank)	Specifies the keystore file to use for HTTPS.
sasstudio.appserver.https.keystorepass	(blank)	Specifies the keystore password to use for HTTPS.

Property	Default Value	Description
sasstudio.appserver.https.port	38443	Specifies the port to use for HTTPS.
sasstudio.appserver.port	38080	Specifies the port to use for HTTP.
webdms.allowBackgroundSubmit	true	Specifies whether the Background Submit option is available when you right-click a .sas file in the navigation tree in the SAS Studio workspace.
webdms.allowFolderShortcuts	true	Specifies whether you can create folder shortcuts in the user interface.
webdms.batchSubmissionResultsRetentionPeriod	24	Specifies the number of hours to keep the output files from a background submission.
webdms.customPathRoot	(blank)	Specifies a path that determines the root node in the Folders tree.
webdms.defaultEncoding	UTF-8	Specifies the default SAS encoding method.
webdms.defaultVFN	ANY	Specifies the default value for the VALIDVARNAME option.
webdms.globalSettings	!SASRoot/ GlobalStudio Settings	Specifies the directory location for global XML files.
webdms.longPollingHoldTimeSeconds	30	Specifies the maximum number of seconds to wait for a message from the client.
webdms.maxNumActiveBatchSubmissions	3	Specifies the maximum number of active background jobs for the current SAS Studio user.
webdms.maxNumActiveBatchSubmissionsSystem	24	Specifies the maximum number of background jobs that can be submitted for a given instance of SAS Studio across all users.
webdms.maxSessionTimeoutInHours	1	Specifies the maximum number of hours a user can specify for the session time-out value in preferences.
webdms.showSystemRoot	true	Specifies that the system root location be displayed in the Folders tree. Note: Set the value to false when the LOCKDOWN statement or the LOCKDOWN system option is used.
webdms.studioDataParentDirectory	(blank)	Specifies the location of SAS Studio preferences, snippets, my tasks, and more. This preference is specific to the local computer. The default value is blank. An administrator must mount a shared location to access data from any workspace server session.

Property	Default Value	Description
webdms.workspaceServer.allowGetRecordCount	true	Specifies whether to retrieve all of the rows for database tables. If you set this property to <code>false</code> , performance improves, but you might not see all rows of the table. For example, for large tables, total rows and filtered rows appear as Unavailable in the user interface. If the table has fewer than 100 rows or you scroll to the last page of the table, the values for the total rows and filtered rows are shown.
webdms.workspaceServer.cacheTableRow	true	Specifies whether to cache the rows from database tables to improve performance. If you use caching, the row count could be wrong if you modify the table. You must click Refresh to remove the value from the cache and to force a re-query of the database. If correct row count is more important than performance improvement, set this property to <code>false</code> to disable caching.
webdms.workspaceServer.hostName	localhost	Specifies the host to use to connect to the workspace server.
webdms.workspaceServer.largeTableRows	50,000	Specifies the maximum number of rows to display in the table viewer. If the number of rows in the table is unknown or greater than the value specified for the <code>webdms.workspaceServer.largeTableRows</code> property, the following behavior occurs: <ul style="list-style-type: none"> ■ SAS Studio displays a warning that sorting could take a long time. ■ SAS Studio does not generate a list of distinct values to select from when SAS Studio filters the data.
webdms.workspaceServer.port	8591	Specifies the port to use to connect to the workspace server.

Configuration Properties: Reference (System)

Note: If you are a tenant administrator in a multi-tenant system, you are not able to access all of these configuration instances.

Commons REST Client

The following are properties to configure the commons REST client library, a library that all SAS Viya microservices incorporate.

`sas.commons.rest.client`

The set of configuration properties for the commons REST client library.

Bypass HTTP proxy

Enables requests to be routed directly to the service rather than through the HTTP proxy. Changing **Bypass HTTP proxy** requires you to restart all SAS Viya services.

Java Virtual Machine (JVM)

The set of properties (Java options) that are used to configure the Java Virtual Machine when it is launched. Each JVM property defined in SAS Environment Manager corresponds to a single Java option.

To define service or global options for the JVM, follow the steps listed in “[Create Configuration Instances](#)” on page 216.

Note: Creating or modifying one of these property values requires you to restart one or more SAS Viya services. For more information, see “[General Servers and Services: Operate](#)” on page 599.

When adding each JVM property, remember these guidelines:

- For the list of the valid Java options and descriptions, see <http://docs.oracle.com/javase/6/docs/technotes/tools/windows/java.html>.
- The property name for each Java option that you add must start with the string, `java_option_` (for example, `java_option_xmx`).
- The property value is a single Java command-line option (for example, `-Xmx512m`).
- When the property names match, Java options specified at the service level override global Java options.
- Matching a Java option’s property name (with a value consisting of a zero-length string) is the only way to disable Java option values.
- There is no control over the order that the JVM processes Java options.

Security

The following are properties to configure web security.

sas.common.web.security

The set of properties that are used to configure web security.

`content-security-policy`

The string used for the Content-Security-Policy HTTP header.

`content-security-policy-enabled`

Sends the Content-Security-Policy header in HTTP responses to prevent injection attacks.

`x-content-type-options`

The string used for the X-Content-Type-Options header for unsecured endpoints.

`x-content-type-options-enabled`

Sends the X-Content-Type-Options header in HTTP responses for unsecured endpoints.

`x-frame-options`

The string used for the X-Frame-Options HTTP header. A restart is required to pick up changes to this property.

`x-frame-options-enabled`

Sends the X-Frame-Options header in HTTP responses. A restart is required to pick up changes to this property.

`x-xss-protection`

The string used for the X-XSS-Protection header for unsecured endpoints.

`x-xss-protection-enabled`

Sends the X-XSS-Protection header in HTTP responses for unsecured endpoints.

sas.commons.web.security.cors

The set of properties that are used to configure Cross-Origin Resource Sharing (CORS) security. By default, CORS is enabled. For more information about CORS, see [CORS support in Spring Framework](#).

To configure CORS for SAS Logon Manager, SASLogon, use [cors](#).

allowCredentials

Allows credentials to be used in cross-origin requests. By default, this property is set to **On**.

allowedHeaders

The comma-separated list of HTTP headers that are allowed, by default, in cross-origin requests. Specify an asterisk (*) to match any header.

allowedOrigins

The comma-separated list of origins that are allowed by default. The list can contain regular expressions. Specify an asterisk (*) to match any origin.

allowedMethods

The comma-separated list of HTTP methods that are allowed, by default, in cross-origin requests. Specify an asterisk (*) to match any method.

sas.commons.web.security.csrf

The set of properties that are used to configure Cross-Site Request Forgery (CSRF) security. By default, CSRF is enabled. To disable it, create a new configuration for the security definition. Specify the property name as **enable-csrf** and the value as **false**. For more information, see [“Create Configuration Instances” on page 216](#).

SAS Viya protects against CSRF using the following:

- Synchronizer Tokens: Randomly generated tokens that are associated with the user’s current session. CSRF is checked only on requests with authenticated sessions, and is always skipped on GET, HEAD, TRACE, and OPTIONS requests. For more information, see [Cross-Site Request Forgery \(CSRF\) Prevention Cheat Sheet](#).
- Header Checking: A filter that checks that the HTTP Referer header has the host and port of the requested URI or matches an optional whitelist of URIs that is configured as a comma-separated list in the *sas.commons.web.security.csrf.allowedUri* property.

For more information about CSRF, see [Common application properties](#).

allowedReferers

This property is currently not supported and should be left blank.

allowedUri

The comma-separated list of referer URIs that are allowed by default. The list must contain regular expressions.

failIfNoHeaders

Blocks requests if both the Origin and Referer headers are absent.

sas.security

The set of configuration properties that are used to configure security for SAS Viya servers and services. The *sas.security* configuration instance applies to all SAS Viya servers and services (global).

network.databaseTraffic.enabled

Toggle security for database traffic.

- network.sasData.enabled
Toggle security for other SAS information.
- network.serverControl.enabled
Toggle security for serverControl.
- network.web.enabled
Toggle security for web based traffic.

Spring Boot Services

Here is the list of third-party, Spring Boot services that you can configure. For a list of the valid property names and descriptions, see <https://docs.spring.io/spring-boot/docs/current/reference/html/common-application-properties.html>.

Endpoints

The set of properties that are used to configure Spring Actuator endpoints.

Flyway

The set of properties that are used to configure Spring Flyway integration.

Liquibase

The set of properties that are used to configure Spring Liquibase integration.

Logging

The set of properties that are used to configure logging.

Logging.Level

The set of properties that are used to configure logging levels.

Management

The set of properties that are used to configure Spring application management.

Multipart

The set of properties that are used to configure Spring multipart handling.

Security

The set of properties that are used to configure Spring security.

Server

The set of properties that are used to configure the embedded Spring server.

Shell

The set of properties that are used to configure the Spring remote shell.

Spring

The set of properties that are used to configure other Spring features.

zones

The set of properties that are used to configure zone information for multi-tenancy. Modifying one of these property values requires you to restart the service.

internal.hostnames




The comma-separated list of internal host names that are used to access the provider zone, or that are used in a subdomain to access other zones.

TIP Be sure to specify the base host name without any tenant prefixes.

Configuration Properties: Interfaces

There are several interfaces that you can use to manage configuration properties for SAS Viya servers, services, and applications. The following table lists these interfaces and the shading indicates the relative amount of SAS Viya configuration that each covers:

Table A.15 Interfaces to CAS Administration

 Ansible	A software orchestration tool that provides a straightforward approach to deploying and provisioning SAS Viya. You can set configuration property defaults at installation.
 SAS Environment Manager	A graphical enterprise web application that enables you to modify and view SAS Viya configuration properties.
 Command-line interface	A command-line interface that enables you to manage SAS Viya configuration properties.

Content Management

Content Management: Overview

The Content page contains objects (such as SAS content and reports) that you save and that are organized into folders. When you open the Content window, you have access to your own data in the folder named **My Folder** unless you are an administrator.

When you log on to SAS Environment Manager for the first time, a folder named **My Folder** is automatically created for you. This folder contains items that you do not want to share with other users. The contents of this folder are visible only to you and administrative users.



When you identify an item such as a report as a favorite in the SAS Home window, the reference to that item is stored in the **My Favorites** folder. You can select the entries in this folder to quickly access frequently used reports and data.


As you work with reports and data, a list of the items that you access is kept in the **My History** folder. You can select entries in this folder to quickly return to items that you have recently worked with. The folder stores the 40 most recent items that you access. This folder exists only after you perform an action that creates a history record (such as creating and saving a report in SAS Visual Analytics or modifying the description of a folder in SAS Environment Manager). This folder is in the **Users** ⇒ *user_ID* folder.

As you work, data that is used by various applications is stored in the **Application Data** folder. This folder exists only after you use an application to modify an object in an application (such as creating a folder in SAS Environment Manager). This folder is in the **Users** ⇒ *user_ID* folder.


Content Management: How To

Navigate the Folders

To open the Content page, select  **Content** from the navigation menu. The left side of the page displays a list of the folders to which you have access. Click  to the right of a folder name to open the folder and to view the content and subfolders.


Click  to move up one level in the folder hierarchy.


The name of the folder that you are currently accessing is displayed in the menu at the top of the page. To move to a different level in the folder hierarchy, click the folder name and select the folder that you want to access from the menu.

Click  to sort the folders and objects by name or date of last use.

View and Edit Authorization for a Folder and Folder Objects


By default, the authorization for a top-level folder allows the owner full access to the folder. See [“Inheritance” on page 119](#) for information about folder authorization.

To view the authorization for a folder or an object inside a folder, select the folder or object in the list, select  and then select **View Authorization**, or select **View Authorization** from the pop-up menu. The View Authorization window appears, where you can view the permission settings for the selected folder.


To edit the authorization for a folder or object, select  and then select **Edit Authorization**, or select **Edit authorization** from the pop-up menu. You can also select **Edit** on the View Authorization window. The Edit Authorization window appears, where you can change the permission settings for the selected folder or object. See [“General Authorization: How to \(Authorization Window\)” on page 110](#) for information about using the Authorization window to set permissions for a folder or object.

View Content Properties



The right side of the Content page displays the properties for the currently selected object or folder. Expand the **Basic Properties** or **Advanced** sections to view the properties for the selected object or folder.

Click  to change the name and description for a folder.

Create a New Folder


Click the **New Folder** icon  to create a new folder. The folder appears in the current location in the hierarchy, with the default name of **New Folder**. If you do not have permission to create a folder at the current level, the icon is not selectable. By default, only a SAS administrator can create a top-level folder.

Search Folders

Click  to search the folders. After the results are displayed in the Search Results dialog box, you can filter the results by object type, the user that modified the object, and the date on which the object was modified. Click  next to an object in the **Results** list to view the object's type, location, date created, and date modified.

Add a Shortcut


You can create a shortcut to a folder or an object and save the shortcut in a folder that you choose. Shortcuts enable you to quickly access folders or objects rather than having to navigate to them each time.

- 1 Right-click on the folder or object for which you want to create a shortcut and select **Add as shortcut** from the pop-up menu.
- 2 In the Add as shortcut window, select the location to which you want to save the shortcut. Click  if you need to create a new folder.
- 3 Click **OK** to save the shortcut. The shortcut is named **Objectname-Shortcut** in your selected location.

Note: You cannot specify authorization settings for a shortcut. Authorization settings are specified for the child member that is associated with the shortcut.

Export the Reports in a Folder

You can export the reports in a folder as a JSON package. Conveyed permissions on a report are not exported along with the report. Only permissions that are directly set on a report are exported.

- 1 Select the folder that contains the reports that you want to export.
- 2 Click  or select **Export** from the pop-up menu.
- 3 In the **Export file** field, specify the name, without a file extension, of the exported package file. The export function adds the `.json` extension to the filename that you specify. If you do not specify a filename, the file is exported using the default name `Package.json`. If a file with the name `Package.json` already exists, the file is exported using the convention `Package (1).json`, `Package (2).json`, and so on.


Note: If you export the contents of a folder that does not contain any reports, the export process creates a file named `undefined.json` that does not contain any content.

- 4 Click **Export** to create the package file. The file is saved to the default location where files are saved for your browser.

Note: If you are exporting a folder that contains many reports, the export process might be lengthy.

Import a Package to a Folder

You can import a JSON package file that contains reports to a folder. All of the reports in the package file are imported. You cannot selectively import reports from a file.

- 1 Click .
- 2 In the Import dialog box, navigate on your file system to the package file that you want to import. The dialog box displays a list of the contents of the selected package file. If you select a package file that contains a large amount of content, it might take several minutes for the dialog box to display the folder tree.
- 3 Select **Properties** to display information about any object that you select in the package file contents list. Select **Dependencies** to display a list of the objects that a selected object depends on.
- 4 Select **Compare** to compare the contents of the package file to the objects that are currently in the destination folder. Select an object in the contents list to view information about whether the object exists in the destination folder.
- 5 To import the objects in the package file, select **Import**.

Note: If you import a top-level folder, you must refresh the **Content** view.

Move an Object or Folder

If you have the appropriate permissions, you can move an object or subfolder to a folder that you choose. You must have these permissions:

Source folder

Read, Remove, Read (convey), Update (convey)


Object in the source folder that you are moving

Read, Update


Target folder

Read, Add

See “[Inheritance](#)” on page 119 for more information about permissions.

- 1 Right-click on the object that you want to move and select **Move to folder** from the pop-up menu.
- 2 In the Move to folder window, select the location to which you want to move the object. Click  if you need to create a new folder.
- 3 Click **OK** to move the object.

Delete Content

Select a folder or other content in the hierarchy and click  to delete the folder or content.

If you are deleting content in a folder, you must have Remove permission for the folder and Delete permission for the object that you are deleting.

Content Management: Concepts

Folders

Information that you or other users save is stored and organized in folders. A folder is a virtual container rather than a representation of a physical file system. A folder contains members, which are URIs for other folders, SAS resources, or resources outside of SAS.

A member in a folder can be either a child or a reference.

Child Members

The URI of a resource is a child member to only one folder (its parent). Because a child member can have only one parent, you cannot copy or duplicate the child member to another folder. However, if you have the proper authorization, you can move a child member to another folder, which then becomes its new parent.

An example of a folder that contains child members is a department’s folder. It contains the reports that everyone in the department uses. This folder is shared, so everyone in the department can access the folder and its reports. Examples of child members include the following:

- Subfolders
- Reports
- Data preparation plans

Reference Members

A reference member is a pointer to a resource that exists as a child in another folder. Reference members in multiple folders can point to the same object

An example of a reference member is an entry in a history folder. If you access a report in your department’s shared folder, an entry is saved in your history folder that contains the URI for the report. Many other users can access the same report, so each of those users will have an entry in their history folder with a reference to the same report. Reference members include the following:

- History (reports that have been recently accessed)
- Favorites

- Shortcuts to objects (child members) or folders

Access to a reference member does not affect access to the child member that is associated with the reference member.

Promotion (Import and Export)

Promotion Overview

Promotion is the process of capturing content and moving it to a different location. The following scenarios are supported:

- promoting content from SAS 9.4 to SAS Viya. SAS Visual Analytics content can be promoted from SAS Visual Analytics 7.4 to SAS Visual Analytics 8.2.
- moving content from one SAS Viya environment to another (for example, from a test environment to a production environment). SAS Visual Analytics content can be promoted to the same release or to a newer release. For example, SAS Visual Analytics content can be promoted from SAS Visual Analytics 8.1 to SAS Visual Analytics 8.2 or from SAS Visual Analytics 8.2 to SAS Visual Analytics 8.2.

Within these scenarios, the following types of content can be promoted:

- reports and folders
- explorations
- graph templates

How To

In SAS Viya, promotion is performed using the transfer plug-in to the admin command-line interface (CLI).

Prerequisites

Note: You must have administrator privileges to promote data. For details and instructions, see [SAS Viya Administration: Identity Management](#).

Before using the transfer commands to perform imports or exports, do the following:

- 1 Familiarize yourself with the general [Command-line interface on page 182](#) documentation.
- 2 Follow the instructions [here on page 185](#) to set up your environment.

Note: Use the Help from within the transfer plug-in to learn about the available commands, subcommands, and options. For more information, see [“Integrated Help” on page 187](#).

See Also

- [“Inventory” on page 182](#)
- [“Command-Line Interface: Preliminary Instructions” on page 185](#)

Promote Content from SAS 9.4 to SAS Viya

Promotion of content from SAS Visual Analytics 7.x or later is supported. For earlier releases, the promotion will be attempted, but you might need to make some adjustments to the content after the promotion.

Promote Content from SAS Visual Analytics 7.x

Note: If you logged in using a named profile, you must do one of the following to use that profile:

- set the `SAS_CLI_PROFILE` environment variable to the name of the profile. This will remain in affect until you log off from the environment. For example, use the following command to set the `SAS_CLI_PROFILE` environment variable to `Target1`:

```
export SAS_CLI_PROFILE=Target1
```

- include the `profile` global option on each CLI command in this section as follows: `sas-admin --profile profile-name transfer CLI-commands`

You can promote reports and explorations that were created using SAS Visual Analytics 7.x to SAS Viya. To promote SAS Visual Analytics 7.x reports that contain stored processes, see [“Promote Reports with Stored Processes from SAS Visual Analytics 7.x” on page 265](#). For other types of content, follow these steps:

- 1 Make sure that you have completed the prerequisite steps. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).
- 2 In SAS 9.4, export the objects that you want to promote. You can use either the Export SAS Package Wizard (available in SAS Management Console) or the batch export tool.

Ensure that the package file is in a location that is accessible by the SAS Viya machine.

TIP If the content that you are promoting has images or links to other reports, select the **Include dependent objects** option from SAS Management Console in order to get those content pieces inside the package. You can deselect tables or libraries from the list since they are not supported by the SAS Viya promotion process.

- 3 From a command window on your SAS Viya machine, navigate to this directory: `/opt/sas/viya/home/bin`. Then, run the `sas-admin transfer upload` command to upload the SAS 9.4 package file from your local machine to the SAS Viya environment.

For example, the following command uploads a SAS 9.4 package file called `myPackage.spk`. Since the `--profile` option is not specified and the `SAS_CLI_PROFILE` environment is not set, the file is uploaded to the SAS Viya environment that is specified in your default profile. The `--mapping` option creates a mapping file that you can edit to specify substitution values for data libraries and other connections. The file is written in YAML format.

```
sas-admin transfer upload --spk MyPackage.spk --mapping MyPackageMapping.yml
```

TIP For those familiar with administering SAS 9.x, the SAS Viya mapping file is analogous to the SAS 9.x *substitution properties* file.

- 4 Open the mapping file using a text editor. If no path was specified in the `--mapping` option, the file is located in the directory where the upload command was executed.

For example, to migrate a package file that contains reports and explorations, on the `target:` lines in the mapping file, enter the table name, caslib, and server name where each table will reside in SAS Viya. An example of such a mapping file is shown below:

```
version: 1
```



```

connectors:
  Table:
  - resourcename: ""
    source: default
    target: ",HPS,cas-shared-mpp"
  - resourcename: ""
    source: /<path to table>/ORION_STAR_SCHEMA(Table)
    target: ORION_STAR_SCHEMA
  - resourcename: ""
    source: /<path to table>/NETWORKDIAGDATA(Table)
    target: NETWORKDIAGDATA,Public,cas-shared-default
  - resourcename: ""
    source: /<path to table>/RESORT_CUSTOMERS(Table)
    target: RESORT_CUSTOMERS,Public,cas-shared-default
  - resourcename: ""
    source: /<path to table>/SIMBA(Table)
    target: SIMBA

```

Note: The information about the `target:` line below the `source: default` line applies to any other table in the file for which one or more of the values were omitted from the `target:` line. For more information about the mapping file, see [“Edit the Mapping File” on page 268](#).

- 5 Use the `sas-admin transfer list` command to obtain the package ID of the package that you just uploaded. You can also obtain the package ID of the package from the output of the `sas-admin transfer upload` command from the prior step.
- 6 Run the `sas-admin transfer import` command to import the content and the table connections to SAS Viya. Here is an example:

```

sas-admin transfer import --id 9d4ca052-1788-4e68-b558-a29069d22451
--mapping MyPackageMapping.yml

```

Note: If you are importing a report that contains a custom theme, see [“Reference” on page 270](#).

Promote Reports with Stored Processes from SAS Visual Analytics 7.x

Unlike other types of content, promotions of reports that contain stored processes from SAS 9.4 systems are handled differently. The report object is promoted into SAS Viya, but the stored process is not. Instead, access is provided to the stored processes within the SAS 9.4 system directly. This is accomplished by specifying an endpoint that is used to call the stored process server.

To promote a SAS Visual Analytics 7.4 report that contains a stored process to SAS Viya, follow these steps:

- 1 Add a whitelist entry for the SAS Viya host machine into the SAS 9.4 system. To enable the SAS Viya deployment to access the SAS 9.4 system, you must add an entry into `sas.web.csrf.referers.knownHosts` on the SAS 9.4 server where the stored process server is located. For information about how to update this value, see [Modifying the Whitelist for URLs and HTTP Request Methods](#).
- 2 Determine the value of the stored process URL to use.
 - a Log on to SAS Management Console.
 - b On the **Plug-ins** tab, select **Application Management** ⇒ **Configuration Manager** ⇒ **SAS Application Infrastructure**. Right-click **Stored Process Web App 9.4**, and select **Properties**.
 - c Check the values from the **Internal Connection** tab and the **External Connection** tab.
 - If the values from the **External Connection** tab are different from the **Internal Connection** tab, and the **Use internal connection information** check box is not selected, then use the values from the **External Connection** tab for the following step.

- If the values from the **External Connection** tab are the same as the **Internal Connection** tab, or the **Use internal connection information** check box is selected, then use the values from the **Internal Connection** tab for the following step.
- d Concatenate the following values together to compose the URL:
- **Communication Protocol** value
 - `://`
 - **Host Name** value
 - `:`
 - **Port Number** value
 - **Service** value
 - `/do`
 - Click **Cancel**.
- 3 Follow steps 1 through 3 [here](#) to export the report that contains the stored process.
- 4 On [the upload step](#), a mapping file was created. Add the URL that you identified in step 2 above to the mapping file.

Here is a sample mapping file with the URL included:

```
---
version: 1
connectors:
  Table:
    - resourceName: ""
      source: "default"
      target: ""
options:
  storedProcessBaseUrl: "http://serverA.demo.com:7980/SASStoredProcess/do"
```

- 5 Follow steps 5 and 6 [here](#) to import the content to SAS Viya.

Note: You must adjust the security options on the SAS Stored Process web application to allow stored processes to be included in an inline frame (IFRAME).

Promote Content from One SAS Viya Environment to Another

Promote Content from SAS Visual Analytics 8.1

Follow these steps to promote content from SAS Visual Analytics 8.1 to SAS Visual Analytics 8.2. You can promote folders and reports.

- 1 From the source machine, perform the steps in the “Prerequisites” section in [SAS Viya 3.2 Administration/ Promotion \(Import and Export\)](#). Then follow steps 1 through 3 in the “Promote Content from One SAS Viya Environment to Another” section in [SAS Viya 3.2 Administration/ Promotion \(Import and Export\)](#). These are the steps to create a package file from the SAS Visual Analytics 8.1 environment, and download the package file to a JSON file on your local machine.
- 2 From the target machine, complete the prerequisite steps. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).

- 3 From a command window on the target SAS Viya machine, navigate to this directory: `/opt/sas/viya/home/bin`. Then, run the `sas-admin transfer upload` command to upload the package to the target SAS Viya environment. Make sure that the profile specifies the correct host information for the target environment.

For example, the following command uploads the file `MyPackage.json` to the Prod environment (that is, the SAS Viya environment that is specified in a profile called Prod).

```
sas-admin transfer --profile Prod upload --file MyPackage.json
```

Note: You can use the `--mapping` option here to create a mapping file that you can edit to specify substitution values. For more information, see the [upload step on page 264](#) in the “Promote Content from SAS Visual Analytics 7.x” section.


The ID of the created package is displayed at the end of the displayed output. Make a note of (or copy) the package ID. You will need it in the next step.

- 4 To complete the promotion, run the `sas-admin transfer import` command to import the uploaded package to SAS Viya. Make sure that the profile specifies the correct host information for the target environment. Specify the package ID of the uploaded package that you noted in step 5.

For example, the following command imports the package that was uploaded in step 5 to the Prod environment:

```
sas-admin transfer --profile Prod import --id 9d4ca052-1788-4e68-b558-a29069d22451
```


Note: You can use the `--mapping` option here to specify the mapping file with substitution values that you created on the upload step. For more information, see the [import step on page 264](#) in the “Promote Content from SAS Visual Analytics 7.x” section.

- 5 To see the uploaded content, open SAS Environment Manager in the target environment. In the side menu , under SAS Environment Manager, select **Content**.

Note: If you are promoting a geo map from SAS Visual Analytics 8.1 to SAS Visual Analytics 8.2, be aware that the colors of the geo map in SAS Visual Analytics 8.2 are different from the colors of the geo map in SAS Visual Analytics 8.1.

Promote Content from SAS Visual Analytics 8.2

Follow these steps to promote content from one SAS Viya environment to another (for example, from a test environment to a production environment). You can promote folders and reports.

- 1 Make sure that you have completed the prerequisite steps on the source machine. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).
- 2 Export content from the source environment to a package:
 - a Open SAS Environment Manager in the source environment. In the side menu , under SAS Environment Manager, select **Content**.
 - b In the navigation pane, navigate to each content item that you want to export. For each item to be exported, look at the **Basic Properties** in the right pane and make a note of the item’s ID and type.
 - c From a command window on the source SAS Viya machine, navigate to this directory: `/opt/sas/viya/home/bin`. Then, run the `sas-admin transfer export` command to export the content items to a package file.

If you are exporting only one item, you can specify its ID and type on the command line by using the `--resource-uri` option. To export multiple items, specify the items in a JSON file, and use the `--request` option to specify the file.

For example, the following command exports a single report from a Test environment (that is, the SAS Viya environment that is specified in a profile called Test).

```
sas-admin transfer --profile Test export --resource-uri "/reports/reports/faa7f5f2-0822-4ca0-9f92-23bda3e02738"
--name "Export Report"
```

- d The ID of the created package is displayed at the end of the displayed output. Here is an example:

```
Package created: faa7f5f2-0822-4ca0-9f92-23bda3e02738
```

- 3 From a command window on the source SAS Viya machine, navigate to this directory: `/opt/sas/viya/home/bin`. Then, run the `sas-admin transfer download` command to download the exported package to your local machine.

For example, the following command downloads a package from the Test environment and stores it in a package file called `MyPackage.json` on your local machine. If you do not specify a path, the file is created in the directory where the command was executed.

```
sas-admin transfer --profile Test download --id faa7f5f2-0822-4ca0-9f92-23bda3e02738 --file MyPackage.json
```

- 4 Complete the prerequisite steps on the target machine. See [“Command-Line Interface: Preliminary Instructions” on page 185](#).
- 5 From a command window on the target SAS Viya machine, navigate to this directory: `/opt/sas/viya/home/bin`. Then, run the `sas-admin transfer upload` command to upload the package to the target SAS Viya environment. Make sure that the profile specifies the correct host information for the target environment.

For example, the following command uploads the file, `MyPackage.json`, to the Prod environment. The file, `MyPackage.json`, was downloaded to your local machine in step 3. The Prod environment is the SAS Viya environment that is specified in a profile called Prod.

```
sas-admin transfer --profile Prod upload --file MyPackage.json
```

Note: You can use the `--mapping` option here to create a mapping file that you can edit to specify substitution values. For more information, see the [upload step on page 264](#) in the “Promote Content from SAS Visual Analytics 7.x” section.

The ID of the created package is displayed at the end of the displayed output. Make a note of (or copy) the package ID. You will need it in the next step.

- 6 To complete the promotion, run the `sas-admin transfer import` command to import the uploaded package to SAS Viya. Make sure that the profile specifies the correct host information for the target environment. Specify the package ID of the uploaded package that you noted in step 5.

For example, the following command imports the package that was uploaded in step 5 to the Prod environment.

```
sas-admin transfer --profile Prod import --id 9d4ca052-1788-4e68-b558-a29069d22451
```

Note: You can use the `--mapping` option here to specify a mapping file with substitution values that you created on the upload step. For more information, see the [import step on page 264](#) in the “Promote Content from SAS Visual Analytics 7.x” section.

- 7 To see the uploaded content, open SAS Environment Manager in the target environment. In the side menu , under SAS Environment Manager, select **Content**.

Edit the Mapping File

The mapping file contains a list of tables, each denoted by the following lines: `resourcename:`, `source:`, and `target:`. It can also contain the URL for a stored process denoted by these lines: `options:` and `storedProcessBaseUrl:`.

SAS 9.4 to SAS Viya Mapping File

You can use two types of connectors in a SAS 9.4 to SAS Viya mapping file, table connector and user connector.

Table Connector

The tables in the mapping file are preceded by a table connector that is denoted by the following lines: `connectors:` and `Table:`. The table connector maps information about a source table to a target table.

The first table in the mapping file that follows the table connector is a default source table denoted by the following line: `source: default`. Use the `target:` line of the default source table to specify the values that are inherited by the other tables.

Here is an example of the target information for a table in the mapping file:

```
1 target: "tablename, cas_library, cas_server_name, 2 data_source_locale"
```

- 1 Specify information about each table on a target line of the mapping file. Any values that are set in the target line of the default source table can be omitted from the target lines of the other tables.
- 2 `data_source_locale` is an optional parameter.

User Connector

The mapping file can also contain a list of users that are preceded by a user connector. The user connector is denoted by the following lines: `connectors:` and `user:`. Each user that follows the user connector is denoted by the following lines: `resourcename:`, `source:`, and `target:`. You can use the user connector to map SAS 9.4 reports whose paths begin with “User Folders” to the specified user’s SAS Viya “My Folder” location.

Use the `source:` line to specify the name of the user that you want to map in the SAS 9.4 “User Folders” location. Use the `target:` line to specify the user ID of the user to which you want to map the rest of the path.

Here is an example of a user connector:

```
user:
- resourcename: ""
  source: "Smith, John"
  target: "josmit"
```

Suppose that you want to promote the following report from the SAS 9.4 environment: `/User Folders/Smith, John/My Folder/My Reports/SomeReport`. If you edited the mapping file to contain the user connector in the preceding example, then after the import the report appears in the SAS Viya `/My Folder/My Reports` folder for the user with the “josmit” user ID. The import can create the user’s “My Folder” even if the user has not previously logged in. If you did not edit the mapping file with the user connector, then the source path is used as is. All needed folders are created in the SAS Viya environment.

Sample Mapping File

Here is a sample mapping file for a SAS 9.4 to SAS Viya promotion:

```
version: 1
connectors:
  Table:
  - resourcename: ""
    source: default
    1 target: ",PUBLIC,cas-shared-default"
  - resourcename: ""
    source: /<path to table>/MAILORDERDEMO(Table)
    2 target: MAILORDERDEMO
  - resourcename: ""
```

```

source: /<path to table>/HEARTCT(Table)
3 target: HEARTCT
- resourcename: ""
source: /<path to table>/CARS(Table)
4 target: CARS
substitutions: []
options: {}

```

- 1 Specifies the target information for the default source table. The table name has been omitted from this line. However, the CAS library and CAS server are present. These two values are passed on to each of the following tables in the file.
- 2 Specifies the target information for the MAILORDERDEMO source table. The only value specified is the table name. The CAS library and CAS server are inherited from the default source table.
- 3 Specifies the target information for the HEARTCT source table. The only value specified is the table name. The CAS library and CAS server are inherited from the default source table.
- 4 Specifies the target information for the CARS source table. The only value specified is the table name. The CAS library and CAS server are inherited from the default source table.

SAS Viya to SAS Viya Mapping File

You can use only a table connector in a SAS Viya to SAS Viya mapping file. The tables in the mapping file are preceded by a table connector that is denoted by the following lines: `connectors:` and `Table:`. The table connector maps information about a source table to a target table.

Here is a sample mapping file for a SAS Viya to SAS Viya promotion:

```

version: 1
connectors:
  table:
    - resourcename: ""
      source: server=<server>;library=<caslib>;table=CARS
      target: server=<server>;library=<caslib?>;table=CARS
substitutions: []
options: {}

```

Reference

Custom Themes

When you import a SAS Visual Analytics report from SAS 9.4 that contains custom themes, the report is promoted into SAS Viya like other types of content. However, if the custom theme is not found in the SAS Viya system, then a new theme is created from the default report theme. Then, the theme ID is set to `base`.

Explorations

When you import a SAS 9.4 exploration into SAS Viya, the exploration is converted to a report. Then, the string " (Exploration)" or the localized equivalent is appended to the name of the report. For example, if a SAS 9.4 exploration that is named `MyModel` is imported into SAS Viya, the exploration is converted to a report that is named `MyModel (Exploration)` in English locations. In other locations, the string that is appended is the localized equivalent of "(Exploration)".

Explorations use the default theme, **Marine**, for the created report content. This results in color differences in some objects.

When importing with the transfer CLI, the locale that is used is determined by the value of the CLI **locale** global option. The default value for the **locale** global option is **en**, which uses English as the locale.

Details and Tips

The following table shows the results of various promotion scenarios:

Action	Result
Promote a folder	Any supported objects that the folder contains (including reports, comments, files, and subfolders) are automatically included in the promotion.
Import a folder during a promotion	An underlying request is issued to create the folder. It is possible that a request to create another folder of the same name can occur. When this happens, you must address the duplicate folder issue and then retry the import.
Import a report or exploration from SAS 9.4 to SAS Viya 3.3	<p>The target value in the mapping file that is created is set to the metadata object name by default. In many instances, the target table name is the same as the source table name. You should review these target values to make sure that the value is set correctly.</p> <p>Note:</p> <p>If you use any auto-generated target values as is, ensure that the source named “default” specifies a target value with at least the CAS library and CAS server to use as defaults.</p>
Promote from SAS Viya to SAS Viya	Authorization rules are included as part of the export package.
Perform an upgrade in place or a promotion of a report from SAS Visual Analytics 8.1 to SAS Visual Analytics 8.2	<p>The report is converted every time you access it.</p> <p>In some cases, this might cause a delay when opening the report in SAS Visual Analytics 8.2. To prevent this delay in the future, follow these steps:</p> <ol style="list-style-type: none"> 1 Open the report from the reports editor on the SAS Visual Analytics 8.2 system. 2 Save the report in the SAS Visual Analytics 8.2 system. The report is saved in the SAS Visual Analytics 8.2 format, which eliminates the need to run the conversion process in the future.
Promote reports from SAS Visual Analytics 7.4 to SAS Visual Analytics 8.2	<p>Three-color ramps are converted to two-color ramps.</p> <p>Consequently, you might notice a difference in the SAS Visual Analytics 8.2 color gradients in heat maps.</p> <p>Note:</p> <p>As a result of the conversion to two-color ramps, the color gradient on some of the SAS sample reports also look different when promoted to SAS Visual Analytics 8.2.</p>

Themes

Overview

The Themes service is accessed through SAS Theme Designer 3.1. SAS Theme Designer enables a user with the appropriate authorization to create and manage SAS themes. A theme consists of the following components:

- style data in JSON format
- a display name
- a content type (application or report)
- a type (system, custom, or sample)
- the ID of the base theme

The base theme is the theme against which a theme is built. There are two types of base themes that a user can create new themes from:

Application theme

an application theme applies a look and feel to the application user interface.

Report theme

a report theme is a theme that applies a look and feel to report content. It typically includes styles for graphs and user interface controls that can be included in reports.

Users can save the theme data as a new theme resource and at that time a unique ID is assigned by the service. The administrator can then publish the theme. The theme is applied to all web applications or reports in the same deployment.

Each time a theme is edited and saved, it must be published again in order for users to see the latest changes. If the theme administrator no longer wants the theme to be visible by all users, the theme can be unpublished or deleted. All web applications supplied by SAS consume custom themes and a default application theme by means of the Themes service. SAS Visual Analytics also consumes report themes and a default report theme by means of the Themes service.

Preferences

Overview

The Preferences component enables clients to create, update, delete, and retrieve preference data for the currently authenticated user. This component is maintained and retrieved by applications in order to provide a custom experience for the end user. Some common uses include providing the settings window for web applications or providing an implicit user-specific state when there is no user interface.

Concepts

Preference data consists of a string identifier and a string value that can be used by an application to customize or enhance the user experience. String values can be simple strings, or more complex constructs like JSON documents.

Guidelines

The Preferences components provide access to preferences for a single person in a given request. A typical user sees only the preferences that belong to them. Preference identifiers are unique on a per-user basis. Applications are responsible for ensuring that the identifiers that they use do not collide with identifiers used by other applications. It is recommended that applications have a name that is applied to every identifier, followed by a dot, then the preference name. For example: *app1.preference*.

The identifier can be any valid printable ASCII character, except for the following characters, which will be rejected: %, &, /, !, #, ', (,), *, +, ,, :, ;, =, ?, @, [,], {, }, |, \$.

Data Administration

Data Administration: Overview

This document provides instructions for administrative tasks such as adding caslibs and loading data. This document assumes that you are familiar with the data and caslib concepts that are explained in [SAS Cloud Analytic Services: Fundamentals](#).

Use the [interface](#) that best meets your needs. Here are suggestions:

- To manage caslibs and CAS tables interactively, use the Data area in SAS Environment Manager. See [“Data Administration: How to \(SAS Environment Manager\)” on page 277](#).
- If SAS Environment Manager is not deployed, use CAS Server Monitor to manage caslibs interactively. See [“Data Administration: How to \(CAS Server Monitor\)” on page 286](#).
- To programmatically manage CAS data, use the [Tables Action Set](#). To get started, see [SAS Viya Quick Start](#).

An in-memory table is loaded from the physical source that is associated with a caslib. Each in-memory table should always be loaded from the same corresponding physical source file, in the same caslib. Loading data from one caslib into another can introduce ambiguity about the physical source for an in-memory table, yielding unexpected results in SAS Visual Analytics.

Note: If you use SAS Environment Manager exclusively to load data, this requirement is met automatically.

Data Administration: How to (SAS Environment Manager)

Introduction

The SAS Environment Manager Data area enables you to view and manage tables, caslibs, user-defined formats, and SAS Viya servers. With the Data area you can perform the following tasks:

- examine available caslibs, loaded tables, and the unloaded tables that are assigned to each caslib
- add new caslibs
- load and unload tables
- delete loaded and unloaded tables and caslibs
- examine the properties for a server, stop a server, and view and terminate sessions
- manage the libraries that are assigned to a server, including the library tables and columns
- import, add, and administer global user-defined formats

Depending on user permissions and privileges, assignments can be viewed and managed for the different tables, caslibs, and servers.

The following instructions explain how to manage caslibs and tables using [SAS Environment Manager on page 466](#).

Navigation

From **SAS Environment Manager**, select **Data**.

At the top of the **Data** page, from the **View** drop-down list, select one of the following views:

Loaded tables

lists all in-memory, global-scope tables that you are authorized to see. This view of tables does not include unloaded tables. In this area you can view the column information for each table, view and edit authorizations, unload a table, delete a table, or view the table properties. Specific actions are granted based on permission settings. This is the default view.

Libraries

lists all global and personal caslibs that you are authorized to see. Use this view to add and manage caslibs.

In this area you can create a new library or edit an existing library. You can view tables, import data or tables into the library, delete a library, and view library properties. You can also view and edit authorization settings for the library.

You can drill into a caslib to see its tables, both loaded and unloaded. You can perform table-specific tasks, such as view column information for a loaded table, import data, load and unload tables, delete a table, or view the tables properties. You can also view and edit authorization settings for a table. Specific actions are granted based on permission settings.

For information about the different types of predefined caslibs, see [“Predefined Caslibs” on page 298](#).

Servers

lists all CAS servers. In this area you can view the running state of a server, content with respect to libraries and tables, and server properties. You can also view and terminate sessions for the server.

Properties for a server enable you to view and modify server settings including: the basic properties and state of a server, nodes for the server, paths list, Superuser role membership, and caslib management privileges.

This view is relevant to data administration if you need to assume Superuser status. When you assume Superuser status, you can stop the server.

User-defined formats


lists SAS provided format libraries and global user-defined formats in each library. In this area you can add new user-defined formats to one of the SAS provided format libraries. You can then edit, copy, delete, and view properties for the user-defined formats. You can also import formats from item stores that are created with the FMTC2ITM procedure.

Note: Session formats are not shown in the **User-defined formats** area.

When you select a table, caslib, server, or user-defined format, available options for that item are accessible from a pop-up menu or from the icon task menu. You can also double-click to open a table, caslib, or server.


Here are some additional navigation features:

- You can modify the display and sort order of columns in the display by right-clicking a column heading and selecting from the available sort options. You can also reorder columns by selecting and dragging a column to the left or right.

You can customize which columns are displayed in the Data area by selecting  and selecting **Manage Columns**. The Columns window contains all available columns for the currently selected table, caslib,


format, or server. From here you can choose to display or hide columns in the Data area. On this window you assign table columns to the **Hidden columns** list or to the **Displayed columns** list.

These preferences persist until the end of your SAS Environment Manager login session.

- For loaded tables, libraries, and servers, you can filter the current displayed list. On the **Filter By** drop-down list, select from the list of available options. You can also enter filter text in the search field.
- You can refresh the current view by selecting . **Refresh** updates the display of your current user session. Changes made by other users are not dynamically updated.

Import Data

In SAS Environment Manager 3.2 you can import data on the Import Data To Caslib window. You can import from a database server, remote file system, local files, and social media feeds. In the Data area, you can import data from the **Libraries** or **Servers** view.

- 1 In the **Libraries** or **Servers** view, select a library or server and then select  from the taskbar.
- 2 In the Import Data To Caslib window, select either of the following tabs:
 - **Data Sources** enables you to create a connection between a caslib and a database server or a remote file system. For more information about the **Data Sources** tab see [“Data Sources Tab: Access Databases or Remote File Systems” on page 303](#).
 - **Import** enables you to create a connection between a caslib and a local file or other data source. For more information about the **Import** tab see [“Import Tab: Access Local Files, Social Media Content, or Esri Data” on page 304](#).



The ability to import data or tables in SAS Environment Manager is affected by two factors:

- Whether you have Read permission on the `/casManagement_capabilities/importData` object URI in the **Security** ⇌ **Rules** area of SAS Environment Manager. You must have Read permission granted in order to import data or tables. If you do not have Read permission granted, the **Import** function is disabled. In the initial configuration, all authenticated users have the necessary access. For information about restricting the ability to import tables in SAS Environment Manager or SAS Visual Analytics, see [Adjust Rules for Access to Functionality](#).
- Whether you have the necessary access to the target caslib. For information about specific requirements, see [Table 9.3 on page 94](#).

Manage Tables


In the Data area, you can load, unload, and delete tables when needed. You can also view table properties, the different columns for a table, and authorization settings for users.

Load a Table

- 1 In the **Libraries** view, select a caslib.
- 2 Right-click, and select **Tables**. Or select  from the taskbar.
- 3 Select the table that you want to load.
- 4 Right-click, and select **Load**. Or select  from the taskbar.


Note: In the **Tables** view for a caslib, loaded tables are identified with a green circle in the **State** column.

Unload a Table


- 1 In the **Loaded tables** or **Libraries** ⇒ **Table** view, select a loaded table. Loaded tables are identified by a green circle in the **State** column.
- 2 Right-click, and select **Unload**. Or select  from the taskbar.
- 3 In the Confirmation window, click **OK**.

Note: In the **Tables** view for a caslib, unloaded tables are identified by a red square in the **State** column.


Delete a Table

- 1 In the **Loaded tables** or **Libraries** ⇒ **Table** view, select a table.
- 2 Right-click, and select **Delete**. Or select  from the taskbar.
- 3 A confirmation window appears. Select **Yes** to delete the table. You can also select whether you want to delete any direct access controls for the table.

View Table Properties

- 1 In the **Loaded tables** or **Libraries** ⇒ **Table** view, select a table.
- 2 Right-click, and select **Properties**. Or select  from the taskbar. Read-only information is displayed in the Table Properties window.

View Table Columns

- 1 In the **Loaded tables** or **Libraries** view, select a loaded table.
- 2 Right-click, and select **Columns**. You can also double click on a table row or select  from the taskbar. A list of columns is displayed.
- 3 To view the properties of a column, select a column, right-click, and select **Properties**. Read-only information is displayed in the Column Properties window.

Note: The table must be loaded to view columns.

Manage Caslibs

In the Data area, you can examine available caslibs and the assigned tables for each caslib. You can add a new caslib, modify path and description settings for a caslib, and delete a caslib. You can also view caslib properties.

Add a Caslib

- 1 In the **Libraries** view, click .
- 2 In the New Caslib window, specify general settings as follows:

Server

Select a server. Only servers to which you are authorized to add a global caslib are listed. See [“Caslib Management Privileges” on page 621](#).

Data source type	Select the type of data source. The Data Source area automatically displays the settings for the selected data source.
Path	Specify data source-specific information for the caslib.
Name	Specify a name for the caslib.


3 Depending on the data source type that you selected, different settings are available on the **Data Source** area. Below are the different data sources that you can select.

- **PATH**
- **HDFS**
- **DNFS**
- **LASR**
- **Oracle**
- **Teradata**
- **Hadoop**
- **Postgres**
- **Impala**
- **ODBC**
- **DB2**

For further information about these data sources and the specific parameters for each data source, see the [addCaslib Action](#).

4 After you have entered all of the parameter settings, click **Save**. The new caslib will be listed in the **Libraries** display.


Modify a Caslib

- 1** In the **Libraries** view, select a global caslib.
- 2** Right-click, and select **Edit**. Or select  from the taskbar.
- 3** In the **Edit Caslib** window, change the caslib path or description as needed.
- 4** Click **Save**.


When editing a caslib, the following restrictions apply:

- Only **Path** and **Description** fields can be edited for PATH, HDFS, and DNFS caslibs.
- Personal caslibs are created for each user and have a specific meaning that does not allow them to be edited.

View Caslib Properties

- 1** In the **Libraries** view, select a caslib.
- 2** Right-click, and select **Properties**. Or select  from the taskbar. Read-only information is displayed in the Library Properties window.

View Tables for a Caslib

- 1 In the **Libraries** view, select a caslib.
- 2 Right-click, and select **Tables**. You can also double-click a caslib row or select  from the taskbar.


A list of the tables that are assigned to the caslib is displayed.

Delete a Caslib

CAUTION! When you delete a caslib, all associated in-memory tables are immediately unloaded.

Note: Deleting a caslib does not affect persisted files in the corresponding data source.


Note: You cannot delete a personal caslib.

- 1 In the **Libraries** view, select a caslib.
- 2 Right-click, and select **Delete**. Or select  from the taskbar.
- 3 In the confirmation window, click **Yes**.

Manage Servers




In the Data area, you can view servers and the caslibs that are assigned to an individual server. You can view and manage sessions for a server and you can also view and manage various properties for a server.

View Caslibs for a Server

- 1 In the **Servers** view, select a server.
- 2 Right-click, and select **Libraries**. You can also double click a server row or select  from the taskbar.

A list of the libraries that are assigned to the server is displayed. You can continue to drill down into tables and columns.

Manage Sessions for a Server

- 1 In the **Servers** view, select a server.
- 2 Right-click, and select **Sessions**. Or select  from the taskbar. On the Sessions window, you can view sessions for the current server. You can refresh the session view by selecting .
- 3 You can also select a session and terminate the session. On the Sessions window, select the check-box for the session or sessions that you want to terminate. Click . The session is terminated.

For more information see [“Terminate a CAS Server Session” on page 613](#).

Manage Properties for a Server


- 1 In the **Servers** view, select a server.
- 2 Right-click, and select **Properties**. Or select  from the taskbar. The following properties are displayed in the Server Properties window:

- **Basic Properties** displays information such as the type, host, and port.
- **Nodes** displays the node information for the selected server including the **Name**, **Role**, **Connected** status, and **IP Address**.

For more information see [“Manage CAS Nodes” on page 614](#).

- **Superuser Role Membership** displays users who are members of the Superuser role. For more information about the Superuser role, see [CAS Server Roles](#).
- **Caslib Management Privileges** displays directly granted privileges for different users. For more information, see [Adjust Caslib Management Privileges](#).

If you are a member of the SAS Administrators group, you can assume the Superuser role and see additional server properties.

- 1 In the **Servers** view, select a server.
- 2 Right-click, and select **Assume the Superuser role**.
- 3 Right-click, and select **Properties**. Or select  from the taskbar.

You can now edit property settings on the Server Properties window. You can edit settings for **Nodes**, **Superuser Role Membership**, and **Caslib Management Privileges**.

In addition to the standard properties, the **Paths List** properties are displayed. The **Paths List** displays a list of unavailable paths also known as **Black List** paths.

Only users who assume the Superuser role for a server can see and manage that server’s paths list.

To manage Paths List properties, see [“Manage Whitelists and Blacklists” on page 612](#).

Stop a Server

If you are a member of the SAS Administrators group, you can assume the Superuser role and stop a Server.

CAUTION! You cannot restart the server from SAS Environment Manager.

For more information see [Stop a CAS Server](#).

Manage User-Defined Formats

In the Data area, you can view and manage user-defined formats. The **Format Filter** pane displays the available format libraries and user-defined formats. You can select the CAS server that you want to work with from the **Server** list. And you can select specific format libraries or specific formats. Items that you select on the **Format Filter** pane are displayed on the **Format name** table.


You can also search for specific formats in the **Format name** table. From this view you can import formats contained in item stores that you create with the FMTC2ITM procedure. You can also add new user-defined formats. You can then edit, copy, and delete user-defined formats. You can also view properties for a user-defined format.

Format Libraries

There are five format libraries that are included with SAS Viya. These format libraries are: USERFORMATS1, USERFORMATS2, USERFORMATS3, USERFORMATS4, and USERFORMATS5. The format libraries initially do not contain any formats. Some application products add formats during installation.

Import an Item Store

You can import SAS client user-defined formats from a SAS item store that was created with the FMTC2ITM procedure. To import formats from an item store:

- 1 Click  to open the Import Formats window.
- 2 Enter a directory path for the item store on the **Import itemstore** field.
- 3 Select a format library from the **Target format library** drop-down list.
- 4 Click **Compare**. Formats that are found in the item store are populated in the **Comparison Results** list. The displayed list of format libraries is derived from the CAS server option FMTSEARCH.

The number of formats that are found is displayed. The status of the formats that are located is also indicated. The following information values are updated:


- **Information** ⓘ - This value shows the number of formats that are found in the item store that can be loaded without conflict.
 - **Warning** ⚠ - This value indicates that there is a possible conflict with one or more of the formats that are listed in the **Comparison Results** list. For example, a duplicate format has been found with an identical range as an existing format.
 - **Error** ❌ - This value indicates that there is a conflict with one or more of the formats that are listed in the **Comparison Results** list. For example, a duplicate format of an existing format has been found. However, the range of values is different between the two formats.
- 5 In the **Comparison Results** list, select the formats that you want to import.
 For any formats that have a warning or error conflict, you can analyze the conflict on the **Format Properties** panel. Select a format that has a conflict by clicking on the format name. The problem range values for both the new format and the existing format are displayed.
 For formats that are in conflict with existing formats, you can assign the new format a different name than the existing format. In the **New Format** field, enter a name for the new format. Click **Apply**. When you select **Import**, the new format is imported with the new name.
 - 6 You can also click **Select all** to select all of the found imports. Select **Import**. The new formats are now available.

Note: Update the **FMTSEARCH** server option list of format libraries to change the displayed list of format libraries. For more information about **FMTSEARCH** see [“Converting a SAS Catalog to an Item Store with the FMTC2ITM Procedure” on page 296](#).

Add a User-Defined Format

To add a new user-defined format:

- 1 Select **+** to open the New Format window.
- 2 Select one of the SAS provided format libraries from the **Format Library** drop-down list. The new format will be assigned to this library.
- 3 Select either **Character** or **Numeric** from the **Type** drop-down list.
- 4 Enter a name for the format in the **Name** field.
- 5 Check the **Save without locale** option if you do not want to include a default locale. By default a format is created with a locale.


- 6 Add rows to the **Range** table. Select **+**. Enter values in the **Name** or **Value** columns for each row. You can delete a row by selecting .


You can also select a range of multiple rows to delete.

- 7 After you have added any needed rows, click **Save**. The new format is listed in the **Format name** table.

Edit a User-Defined Format

To edit a user-defined format:


- 1 Select a format and click .
- 2 On the Edit Format window, make any needed changes to the **Range** table of rows. You can edit values in the **Name** or **Value** columns for existing rows.


Click **+** to add rows and  to delete rows.

- 3 Click **Save** when finished.

Copy a User-Defined Format

To copy a user-defined format:

- 1 Select a format and click .
- 2 On the Copy Format window, enter a name for the copied format in the **Name** field. Make any needed changes to the **Range** table of rows. You can edit values in the **Name** or **Value** columns for existing rows.

Click **+** to add rows and  to delete rows.

- 3 Click **Save** when finished. The new format is listed in the **Format name** table.


Delete a User-Defined Format

To delete a user-defined format:

- 1 Select a format and click . The Delete pane opens.
- 2 Select **Delete**.

View Properties for a User-Defined Format


To view properties for a user-defined format:

- 1 Select a format and click . The Format window appears.
- 2 You can now view properties and row values for the format.

Authorization

You can view and modify authorization settings for tables or libraries. Select one of the following options:


View Authorization

Right-click, and select **View Authorization** or select  from the taskbar. The View Authorization window appears. You can view different users and access levels. If you select **Show individual permissions**, different permission settings are available to view.

You can also select **+** to open the Select Identities window.

If you select **Edit**, the Edit Authorization view for the selected item is displayed.

Edit Authorization


Right-click, and select **Edit Authorization** or select  from the taskbar. The Edit Authorization window appears. You can view different users and modify access levels. If you select **Show individual permissions**, different permission settings are available for edit.


You can also select **+** to open the Select Identities window.

You can also right-click on a table or library and select **View Authorization** or **Edit Authorization**.

For further information about managing access, see [SAS Viya Administration: General Authorization](#) and [SAS Viya Administration: Cloud Analytic Services Authorization](#).

Data Encryption

When you add a caslib with the **New CAS Library** function, you can choose to encrypt the data for that caslib. Depending on the data source type that you select, the **Enable encryption** option is available on the **New CAS Library** window. Select **Enable encryption** and select a domain from the list of available domains or create a new domain by selecting .

You can also view the encryption status for individual tables in a caslib. Select a table and select . The encryption status is listed on the Table Properties window.

Note: In the Data area, the **Encryption** column is hidden by default.

For more information about encrypting data, see [Encryption in SAS Viya: Data at Rest](#).

Data Administration: How to (CAS Server Monitor)

Introduction

CAS Server Monitor enables you to monitor and administer your CAS server. Within CAS Server Monitor, the **System State** view contains various CAS server properties and settings, including the **Global Caslibs** table. This table displays the global caslibs for your environment. From here you can add and delete global caslibs and modify access controls for users and groups.

These instructions explain how to manage global caslibs using [CAS Server Monitor on page 625](#).

Add a Global Caslib

- 1 On the **System State** page, select **Global Caslibs**.
- 2 Click **Add**.

TIP If the **Add** button is disabled, you are not authorized to add a global caslib. See “[Caslib Management Privileges](#)” on page 621.

- 3 On the Add Global Caslib pane, specify general settings as follows:


Caslib	Enter a caslib name.
Description	Enter a description for the caslib.
Path	Enter data source-specific information.
Subdirectories	For a path-based caslib, specifies whether tables and files in subdirectories of the specified path are accessible from the caslib.
Create directory	For a path-based caslib, creates the host directory that you specify in the Path field, if that directory does not already exist.
Permission	For a path-based caslib, sets host-layer permissions on the directory. See “ Using CAS to Modify Host Access ” on page 106.
Active on add	Specifies whether the new caslib becomes the active caslib in your current session.
Hidden	Makes the caslib and its tables unlisted in certain contexts. See “ Reduced Visibility: Hidden Caslibs ” on page 101.
Transient	Specifies that the caslib is scoped to the current session only.
Data source	Specifies the type of source data for the caslib.
Data encryption password	Specifies the encryption password for the caslib.
Encryption domain	Specifies the encryption domain for the caslib.

- 4 Specify additional settings as needed. For information about caslib properties, see [addCaslib Action](#).
- 5 Make sure your settings are as intended. In CAS Server Monitor, caslib properties are not editable.
- 6 Click **OK**.

Delete a Global Caslib


CAUTION! When you delete a caslib, all associated in-memory tables are immediately dropped.

Note: Deleting a caslib does not affect persisted files in the corresponding data source.

- 1 On the **System State** page, click **Global Caslibs**.
- 2 At the end of the row for the caslib, click , and select **Drop Caslib**. On the Drop Global Caslib pane click **OK**.

Manage Access to a Global Caslib

- 1 On the **System State** page, click **Global Caslibs**.

- 2 At the end of the row for the caslib, click , and select **Edit Access Controls**. The Edit Access Controls window appears. From here you can grant or deny permission settings to different users.

See [SAS Viya Administration: Cloud Analytic Services Authorization](#).

Loading Geographic Polygon Data as a CAS Table

Overview of Loading Geographic Polygon Data

Some SAS Viya applications such as SAS Visual Analytics can display geographic maps with colored map regions. By default, countries and their first-level subdivisions can be displayed as a region map. To display other types of map regions, such as postal codes or sales regions, you must define a custom polygon provider that contains the polygons (geographic region shapes).

You can load two types of polygon data into CAS for use in a polygon provider: Esri shapefiles, and SAS map data sets.

After you have loaded the polygon data into CAS, you must define a polygon provider that specifies the parameters for the polygon data. For details about defining a polygon provider in SAS Visual Analytics, see [“Create a Geography Data Item By Using Custom Polygonal Shapes” in SAS Visual Analytics: Working with Report Data](#).

Note: By default, SAS Visual Analytics can retrieve up to 250,000 polygon vertices at a time. If you encounter an error message in a geo map object about the number of polygon vertices, then you might need to reduce the density of your polygon data or filter the data query for your geo map object. In some cases, a very wide ID column in your polygon data can further limit the number of polygon vertices that are retrieved. Check the width of your ID column in SAS Data Explorer if you encounter this message.

Loading Polygon Data from Esri Shapefiles

Overview

To load Esri shapefile data into CAS, you must first convert the shapefile into a SAS data set.

SAS provides two autocall macros to help you inspect and load Esri shapefiles:

`%SHPCNTNT`
displays the contents of the specified shapefile.

`%SHPIMPRT`
converts a shapefile into a SAS data set and loads it into CAS.

TIP Where possible, use shapefiles with unprojected latitude and longitude values. Configuring a polygon provider for projected data can be difficult for users who are inexperienced with map data.

`%SHPCNTNT` Autocall Macro

The `%SHPCNTNT` macro displays the contents of the specified shapefile. You can use the `%SHPCNTNT` macro to identify which variable in the shapefile should be used as an ID variable.

The syntax for the `%SHPCNTNT` macro is as follows:

```
%SHPCNTNT(SHAPEFILEPATH=path-to-shapefile)
```


SHAPEFILEPATH=*path-to-shapefile*

specifies the full path to the shapefile with the .SHP extension. Do not enclose the file path in quotation marks.

%SHPIMPRT Autocall Macro

The %SHPIMPRT macro converts the shapefile into a SAS data set and then loads it into CAS.

Note: To load tables into CAS, you must configure an authentication file. See [Client Authentication Using an Authinfo File](#),

The syntax for the %SHPIMPRT macro is as follows:

%SHPIMPRT(*options*)

SHAPEFILEPATH=*path-to-file*

specifies the full path to the shapefile with the .SHP extension.

ID=*id-column*

specifies a field in the shapefile that identifies the polygons in the map.

Requirement The ID column must contain character data, and cannot contain special characters or double-byte characters.

OUTTABLE=*table-name*

specifies the name of the output table that is loaded into CAS.

CASHOST=*machine-name*

specifies the machine name of the CAS server.

CASPORT=*port-number*

specifies the port for the CAS server.

CASLIB=*library-name*

specifies the library on the CAS server where the output table is loaded.

REDUCE=0|1

(Optional) specifies whether to reduce the density of the polygon data. A value of 1 specifies that the data density is reduced, and a value of 0 specifies that the data density is not reduced.

Reducing the density of your polygon data can improve performance and might enable a greater number of map regions to be displayed at one time.

Default 0

Requirement A license for SAS/GRAPH software is required to reduce the density.

The following example loads a shapefile without reducing the polygon density:

```
%shpimprt(shapefilepath=/tmp/myfile.shp, id=GEOID, outtable=mytable, cashost=cloud.example.com, casport=5570, caslib='casuser');
```

The following example loads a shapefile and reduces the polygon density:

```
%shpimprt(shapefilepath=/tmp/myfile.shp, id=GEOID, outtable=mytable, cashost=cloud.example.com, casport=5570, caslib='casuser' reduce=1);
```

Loading Polygon Data from SAS Map Data Sets

To use a SAS map data set as a polygon provider, you must perform the following steps:

- 1 Create a sequence variable to enable the polygon segments to be read in the correct order. In a SAS DATA step, you can use the `_n_` automatic variable to store the observation number as a sequence variable. For example, the following DATA step creates a sequence variable for the MYMAP data set:

```
data mymap;  
  set mymap;  
  sequence = _n_;  
run;
```

- 2 (Optional) Subset your polygon data to decrease the level of detail and improve performance. Reducing the level of detail might also enable you to display a greater number of map regions at one time.

If you have a license for SAS/GRAPH, then you can use the GREduce procedure to create a DENSITY variable that enables you to reduce the density of your polygon data. Depending on the source of your map data sets, a DENSITY variable might already be present. For more information about the DENSITY variable and the GREduce procedure, see [SAS/GRAPH and Base SAS: Mapping Reference](#).

You can use the DENSITY variable in a WHERE statement in a DATA step to reduce the detail in your polygon data. For example, the following DATA step reduces the MYMAP data set to exclude segments that are density level 4 or greater:

```
data mymap;  
  set mymap;  
  where (density<4);  
run;
```

- 3 Load the data set in your SAS Cloud Analytic Services environment.

CAS Table State Management

Overview of CAS Table State Management

CAS table state management enables you to manage the import, load, and unload of source files in CAS. CAS table state management is performed through the use of jobs that are created from sample jobs that are provided by SAS.

For SAS Viya 3.3, you can import batch data that is directly accessible using a caslib, but might not be in the desired caslib or format. This type of import is used to take source data and make a copy in SASHDAT format.

For example, business processes might produce new data each night in CSV or SAS7BDAT format. It is possible to access the data directly using a global caslib that points to the source of the data. However, for performance reasons, it might be desirable to make a copy of the data in SASHDAT format.

Sample jobs should be used as a starting point. These sample jobs can be used as is for the Public caslib that is associated with the cas-shared-default CAS server. The jobs can be copied, and the copies can be edited or deleted. A job includes the specific options required by the job. In the context of CAS table state management, a job performs an import, load, or unload operation on input files, tables, or loaded tables. Jobs can be listed, copied, updated, and deleted on the SAS Environment Manager Scheduling page. For CAS table state management sample jobs, this page also enables you to make copies that are used to import, load, and unload batch data. Each job can be submitted manually, or scheduled for later execution.

Sample Jobs in SAS Environment Manager

For SAS Viya 3.3, there are three sample jobs provided by SAS for managing table state. These jobs are available on the Scheduling page of SAS Environment Manager. Below are the sample jobs:

Sample: Import cas-shared-default Public data

This job demonstrates settings that import all CSV, SAS7BDAT, and EXCEL files in the Public caslib to SASHDAT files in the same caslib.



Sample: Load cas-shared-default Public data

This job demonstrates how to load all SASHDAT files found in the Public caslib.

Sample: Unload cas-shared-default Public data


This job demonstrates how to unload all loaded CAS tables in the Public caslib that have not been accessed within the past 7 days.

On the Scheduling page of SAS Environment Manager, the sample jobs are listed on the **Jobs** pane. The sample jobs operate on a CAS server named `cas shared default`. You cannot edit or delete the sample jobs. However, you can copy the sample jobs to create unique jobs that you can further customize. Copied jobs contain the options that you can define and update as needed.

To create a new job, select **Copy**  on one of the sample jobs. You can now customize the options for the new job by selecting .



For further information about the Scheduling page in SAS Environment Manager, see [“Scheduling: How to \(SAS Environment Manager\)” on page 587](#).

Viewing Properties for a Job

You can view properties for copied jobs on the Scheduling page. You must select the job and then select . The Job Properties window opens.

On the Job Properties window, select the **Properties and Arguments** tab. On the **View** drop-down list, select **Arguments**. You can now view the settings for the job under the **Value** column. Holding your pointer over the value shows all of the options in a tooltip.

Editing Options for a Job

You can edit options for copied jobs on the Scheduling page. Select the copied job and then select . The Job Properties window appears. On the Job Properties window, select the **Properties and Arguments** tab. On the **View** drop-down list, select **Arguments**. You can now select .

The Edit Job window appears. The different options for the job are listed in the **Options** pane. The job options panel contains the full set of job options as a JSON-formatted string. You can manually edit the settings and simple edits can be made in place.

TIP For more complex edits, copying the options to a JSON editor, making the changes, and copying the result back into options might be helpful.

The following table contains options that are common to the import, load, and unload jobs:

Table A.16 Common Options

Option	Description
serverName	The name of the CAS server on which the operation will be performed.
inputCaslib	The caslib name used as input for the job. For import and load jobs, this is the caslib that contains source files or tables. For an unload job, this is the caslib that contains potential tables to unload.

Option	Description
outputCaslib	The caslib name used for output of the job. For an import job, this is the caslib where output files are written. For a load job, this is the caslib where CAS tables are loaded. outputCaslib is not applicable for an unload job.
filter	The filter is used to subset the list of items from the inputCaslib upon which job operations are performed. See “ Filter Syntax ” on page 292 for more details and example filters.

Filter Syntax

Job options can also contain filters. In its simplest form, a filter selects an item based on whether a condition passes. For example, to select an in-memory table whose name is exactly MYDATA, the following example filter could be used:

```
eq (name, 'MYDATA')
```

In the next example, the filter is used to select a source table name ending in lowercase SASHDAT:

```
endsWith(sourceTableName, '.sashdat')
```

There are several operators that can be used in a filter. The following table contains these operators:

Table A.17 Filter Operators

Operator	Description	Example	Example Result
eq	True if the parameters specified are equal.	eq (name, 'MYDATA')	Only the table named MYDATA is selected.
startsWith	True if the value of the first parameter begins with the value of the second parameter.	startsWith (sourceTableName, 'DEPTA_')	Only source tables beginning with 'DEPTA_' are selected. For example: DEPTA_CUSTOMERS, DEPTA_ADDRESSES.
endsWith	True if the first value of the parameter ends with the value of the second parameter.	endsWith (sourceTableName, '.sashdat')	Only source file or table names ending in lowercase .SASHDAT are selected.
contains	True if the value of the first parameter contains the value of the second parameter.	contains (name, 'SPECIAL')	Only tables whose name contains SPECIAL are selected. For example: MYSPECIALDATA, SPECIALDATA, THISSPECIALDATA.
in	True if the value of the first parameter contains any following value.	in (name, 'TABLE1', 'TABLE2', 'TABLE3')	Only tables TABLE1, TABLE2, or TABLE3 are selected.

The following table contains filter fields that can be used in expressions:

Table A.18 Filter Fields

Field Name	Content
name	This field represents the CAS table name (whether loaded or unloaded).
sourceTableName	This field represents the name of the source file in the input caslib.
tableReference.sourceTableName	This field is an alias for the sourceTableName field and can be used in place of it.

The following table contains filter examples:

Table A.19 Filter Examples

Example	Filter
by file extension (.SASHDAT, .CSV, .SAS7BDAT)	<code>or(endsWith(sourceTableName, '.sashdat'), endsWith(sourceTableName, '.csv'), endsWith(sourceTableName, '.sas7bdat'))</code>
by substring (contains some string)	<code>contains(name, 'DATA')</code>
by exact match	<code>eq(name, 'MAILORDER')</code>
by list of inputs	<code>in(name, 'AIRLINE', 'CUSTOMERS', 'WORLD BANK')</code>
using multiple conditions where either are true	<code>or(eq(name, 'MYDATA'), endsWith(name, 'YOURDATA'))</code>
using multiple conditions where both are true	<code>and(contains(tableReference.sourceTableName, 'DEPTA_'), endsWith(tableReference.sourceTableName, '.sashdat'))</code>

Importing Data

The **Sample: Import cas-shared-default Public data** job imports CSV, SAS7BDAT, and EXCEL files to SASHDAT files. It imports to the `Public` library on an example CAS server named `cas-shared-default`.

This job enables an import from the caslib source defined for the CAS server. By default, the import job imports CSV, SAS7BDAT, and XLS, XLSX (EXCEL) files. It imports those files to the target caslib's source location as SASHDAT files of the same name. Source files can therefore be placed in a path-based caslib (PATH, and DNFS for example) that is accessible by the CAS server controller. The default path for imported files is `/opt/sas/viya/config/data/cas/default/public/`.

Note: For situations where the SASHDAT copy is not required, the load job can be used to load the file directly into memory as a CAS table.

On the **Scheduling** page of SAS Environment Manager, the sample job *Sample: Import cas-shared-default Public data* is available.

Loading Data

The **Sample: Load cas-shared-default Public data** job performs a load operation on managed files or tables in the target caslib. It then creates an in-memory CAS table of the same name in the target caslib.

This job enables you to preload tables for which there is a high user demand. Or, for scenarios where the amount of time needed to load the table is too long due to data size.

On the **Scheduling** page of SAS Environment Manager, the sample job **Sample: Load cas-shared-default Public data** is available. You can modify the following settings for copies of this job:

Table A.20 Load Settings

Setting	Value	Default Value	Sample Values
refresh	Boolean	false	true, false

Note: When the refresh option is set to `true`, each table selected by the filter is unloaded first. If the table is not sourced from the input caslib, it is not reloaded. Therefore, it is important to ensure that the filter is properly set to select only the tables for which you want a refresh. Tables are refreshed only if they are sourced from the caslib that is specified with the `inputCaslib` setting.

Unloading Data

The **Sample: Unload cas-shared-default Public data** job unloads tables in the target caslib either immediately, or based on recent access. This enables you to schedule forced unloads of tables on a routine basis. Or you can schedule an unload request that is based on how often a table is used. The sample job unloads infrequently accessed data in the `Public` table on the `cas-shared-default` server.

The setting `unloadAccessThreshold` is available in the settings for this job. If `unloadAccessThreshold` is set to `PT0D`, all tables in the target caslib are unloaded when the job is run. However, if it is set to a specific time period, those tables that are not accessed within the set time period are unloaded. The sample job is **Sample: Unload cas-shared-default Public data**, and by default, uses an `unloadAccessThreshold` setting of `P7D` (7 days).

On the **Scheduling** page of SAS Environment Manager, the sample job **Sample: Unload cas-shared-default Public data** is available. For copies of this job, you can modify the `unloadAccessThreshold` setting. The following example time threshold values are possible:

Table A.21 Unload Settings

Value	Description
P0D	zero days. This setting results in an immediate unload. There is no threshold.
P7D	period of 7 days
P5M	period of 5 months
PT4H	period of time of 4 hours
PT5M	period of time of 5 minutes

Value	Description
PT45S	period of time of 45 seconds

Source-Data-Specific Settings

Currently you can import CSV, SAS7BDAT, and EXCEL files with the *Sample: Import cas-shared-default Public data* sample job. Settings for these import file types are listed in the following tables.

Table A.22 CSV

Setting	Value Type	Default Value	Sample Values
delimiter	character	,	,
guessRows	integer	200	20,50,500
allowTruncation	Boolean	true	true,false
encoding	string	utf-8	utf-8
getNames	Boolean	true	true,false

Table A.23 SAS7BDAT

Setting	Value Type	Default Value	Sample Values
charMultiplier	decimal	2	1,2,2.5,3,4

Table A.24 EXCEL and XLS

Setting	Value Type	Default Value	Sample Values
getNames	Boolean	true	true,false

Execution and Monitoring of Jobs

The Scheduling page of SAS Environment Manager enables you to schedule and execute the jobs that you define and customize. You can choose to run jobs as the SAS Administrator or as a different user. You can also schedule or unschedule a job.

In the **Jobs** pane of the Scheduling page, right-click on a job. Available functions for that job are listed. You can also execute functions from the icon menu on the **Jobs** pane.

If you select **Run** for a job, you can check the execution of that job by accessing the Data page of SAS Environment Manager. On the Data page, select **Libraries** from the **View** menu. Right-click on the **Public** library for the `cas-shared-default` server. Select **Tables**. From here you can check for the source tables that you are importing or loading.

You can also check the execution of that job from SAS Job Monitor. On the **Jobs** pane of the Scheduling page, select **Monitor Jobs**. This opens SAS Job Monitor. From here you can view the different jobs that have been executed and download the log file that contains details of the execution.

Note: The log file is updated as progress is made. So downloading the log file while the job is running shows progress until that point only. To see later progress, you must download the log again for those jobs.

TIP Open SAS Job Monitor in a separate browser tab or browser instance. You will be able to see job progress while still keeping Tables open in the Libraries view of the Data page. You can watch tables populate, load, or unload.

Preliminary Tasks for User-Defined Formats in SAS Viya 3.3

Overview

For SAS Viya 3.3, CAS does not read a SAS catalog directly. One way to move formats stored in a SAS catalog to a CAS server is to use the FMTC2ITM procedure. This procedure copies the formats in the catalog to a physical file. The physical file is often referred to as an item store. The physical file must be accessible from the CAS server controller and the user must have permission to read the file.

SAS Environment Manager can then import formats from an item store. To create an item store the FMTC2ITM procedure is used. The FMTC2ITM procedure is available in SAS 9.4M3 and later.

Converting a SAS Catalog to an Item Store with the FMTC2ITM Procedure

The FMTC2ITM procedure uses a SAS format catalog as input and as output, and produces an item store that is read in CAS. The FMTC2ITM procedure is used to convert one or more format catalogs into a single item store.

Syntax

```
FMTC2ITM <options>;
SELECT<member-list>;
RUN;
```

Options

PRINT

displays information about each member that is being written.

DEBUG

debugs information about the records that are being written.

CATALOG=memname | libname.memname | (list)

specifies catalog(s) that will be converted to an item store.

ITEMSTORE= fileref | 'filename'

specifies the item store file that is being created.

XMLFILE= fileref

specifies an XML file that is created to accompany the item store file.

PAGESIZE=n

specifies the page size.

LOCALE

specifies locale-sensitive prefixes that are added to item store memnames.

FMTC2ITM Procedure Example

The following example converts the catalogs *formats.orionfmt* and *formats.siriusfmt* to the item store *format1*.

```
proc fmtc2itm catalog=(formats.orionfmt, formats.siriusfmt)
  print locale itemstore="/users/dsmith/formats/format1";
run ;
```

About the FMTC2ITM Procedure

Note that the item store is always written as new, so if the **ITEMSTORE=** option refers to an existing item store, it will be completely overwritten. For a file-based item store, the **XMLFILE=** option can also be provided. It is populated with a small XML stream that accompanies the item store file.

If the **CATALOG=** option is not given, then the default value is **WORK.FORMATS**. If the **CATALOG=** option is given, it can be a single-level name, which is interpreted as a catalog name in **WORK**. It can also be a two-level name, which is interpreted as a libname.memname for a catalog. It can also be a list of catalog names that is enclosed in parentheses.

For an item store that is a file, either a fileref or a quoted string pathname is given. For a list of catalog names that is enclosed in parentheses, each catalog is opened in order and the members of the catalogs are written to the item store. Only the first occurrence of the member is written out.

The **SELECT** statement is optional. If specified, it lists the formats that are selected to be placed in the item store. If a **SELECT** statement is not given, all formats are written to the item store.

Using the Casstartup.lua File

The **casstartup.lua** file is a file that is processed as a LUA client session into the CAS server. It is used to perform static, default deployment tasks. **Casstartup.lua** comes pre-configured with default settings as part of the deployment process. **Casstartup.lua** is included in the start-up processing hierarchy and is used during CAS server start-up. The command line default start-up value is **-startup casstartup.lua**.

During server start-up, you can establish default and custom user-defined format libraries for use in SAS Viya. In a default SAS Viya installation, the caslib **Formats** is created. The **casstartup.lua** file contains the default **addFmtLib** actions. These actions add format libraries **userformats1** through **userformats5**. The **casstartup.lua** file also contains the **setServOpt** action that is used to establish the format search list that each session starts with. Currently, **userformats1** through **userformats5** are placeholders for use with SAS Environment Manager.

Below is an example **casstartup.lua** file:

```
s:sessionProp_addFmtLib{caslib="Formats",fmtLibName="userFormats1",name="userformats1.sashdat",promote=true}
s:sessionProp_addFmtLib{caslib="Formats",fmtLibName="userFormats2",name="userformats2.sashdat",promote=true}
s:sessionProp_addFmtLib{caslib="Formats",fmtLibName="userFormats3",name="userformats3.sashdat",promote=true}
s:sessionProp_addFmtLib{caslib="Formats",fmtLibName="userFormats4",name="userformats4.sashdat",promote=true}
s:sessionProp_addFmtLib{caslib="Formats",fmtLibName="userFormats5",name="userformats5.sashdat",promote=true}
s:configuration_setservopt{fmtsearch="userformats1 userformats2 userformats3 userformats4 userformats5"}
```

The recommended deployment best practice is to target a configuration path for all default filenames. If present, the default names will be found. For example, the following **cfgpath** option

```
-cfgpath /my/config/path
```

will find

```
/my/config/path/casstartup.lua
```

After the deployment tasks have been accomplished, `casstartup.lua` invokes `casstartup_usermods.lua`. Site specific format libraries associated with the `addFmtLib` action should be placed in `casstartup_usermods.lua`. The `casstartup_usermods.lua` file is used for site customization and is automatically applied when present. The local system administrator can add any site-specific start-up processes, such as table loading, to `casstartup_usermods.lua`.

Note: `Casstartup_usermods.lua` is not replaced when software is updated.

Data Administration: Reference





Data Administration: Interfaces

Interfaces

All CAS data management requirements and constraints are always fully enforced. Not all interfaces enable you to see and interact with all CAS data management features.

In the following table, the shaded part of each circle is an approximation of the amount of CAS data management functionality that a particular interface exposes.

Table A.25 Interfaces to Data Administration

	Tables Action Set	A programmatic interface for CASL (the CAS procedure), Python, and Lua.
	SAS Environment Manager	The enterprise graphical web application for administration.
	CAS Server Monitor	A graphical web application that is embedded in the CAS server. Supports adding and deleting global caslibs.
	CASLIB statement	A programmatic interface for adding caslibs. See CASLIB statement .

Predefined Caslibs

The following caslibs are automatically created during deployment. Each caslib has a default assignment and specifications.

AppData*	<code>/opt/sas/viya/config/data/cas/default/appData/</code> Stores data that specific applications use for internal purposes.
Formats	<code>/opt/sas/viya/config/data/cas/default/formats/</code> A shared location for user-defined formats. All users can read. Administrators can read and write.

Models*	<code>/opt/sas/viya/config/data/cas/default/models/</code> Stores models created by SAS Visual Analytics for use in SAS Studio.
Public	<code>/opt/sas/viya/config/data/cas/default/public/</code> A shared location for data. All users can read and write. See “Protecting Files in the Public Caslib” on page 106 .
ReferenceData*	<code>/opt/sas/viya/config/data/cas/default/referenceData/</code> Stores per-server data that specific applications use for internal purposes.
Samples	<code>/opt/sas/viya/config/data/cas/default/samples/</code> Stores sample data, supplied by SAS.
SystemData*	<code>/opt/sas/viya/config/data/cas/default/sysData/</code> Stores application-generated data that is used for general reporting.
VAModels	<code>/opt/sas/viya/config/data/cas/default/vamodels/</code> This is a library for ASTORE objects that are used within a SAS Visual Analytics report.
ProductData	<code>/opt/sas/viya/home/share/productData/</code> Stores product data supplied by SAS.

* Not included in a programming-only deployment.

Note: Some predefined caslibs are hidden or have limited access. For more information about hidden caslibs, see [“Reduced Visibility: Hidden Caslibs” on page 101](#)

Access to SAS 9.4 Data

If you are moving data from SAS 9.4 to SAS Viya, you will need to consider some preliminary information:

- You can move and you can share data between SAS 9 and SAS Viya environments using SAS/CONNECT.
- SAS Viya operates with UTF-8 encoded data. If your SAS 9 installation is not UTF-8 compliant, you might need to re-create your data sets.

See the following topics for more information:

- [Comparing SAS 9 and SAS Viya](#)
- [SAS 9 and SAS Viya](#)
- [Sharing Data Between SAS 9 and SAS Viya using SAS/CONNECT](#)
- [Migrating Data to UTF-8 for SAS Viya 3.3](#)

SAS Data Explorer

Working with SAS Data Explorer

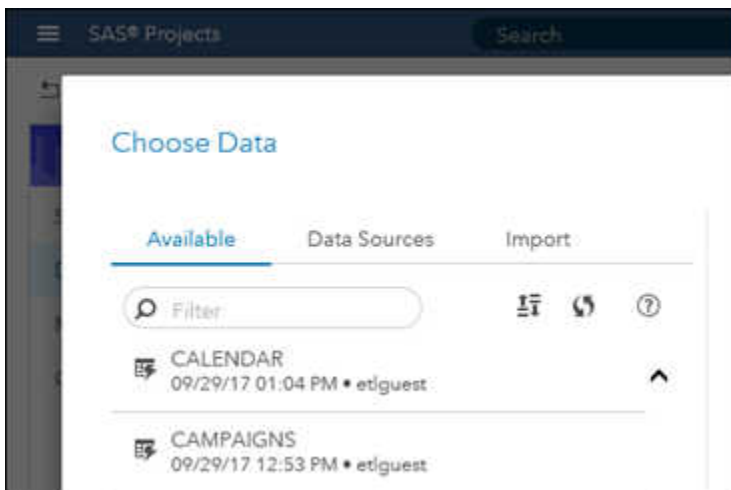
Understanding SAS Data Explorer

SAS Data Explorer and the Choose Data Window

SAS Data Explorer enables you to copy data to memory on SAS Cloud Analytic Services (CAS) server and to perform related tasks. There are two versions of SAS Data Explorer: a stand-alone web application and a window that can be displayed from SAS Viya applications.

SAS Viya applications such as SAS Environment Manager, SAS Visual Analytics, Model Studio, and SAS Decision Manager include a window similar to the Choose Data window shown in the next figure.

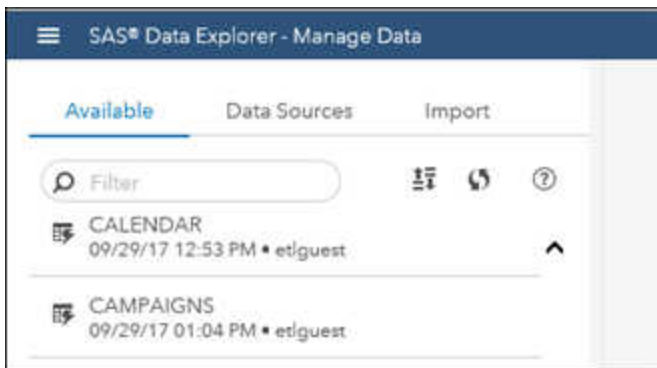
Figure A.1 Choose Data Window Displayed in SAS Projects



In some applications, the Choose Data window has a different name, but this window typically includes the **Data Sources** tab and the **Import** tab. These tabs enable you to copy data to memory on a CAS server and perform related tasks. For more information about this window, see [“Getting Started with the Choose Data Window” on page 302](#).

The SAS Data Explorer web application is available only if your site has licensed SAS Data Preparation. If you have licensed that offering, you can display SAS Data Explorer by selecting **Manage Data** from SAS Home. The next figure shows the SAS Data Explorer application.

Figure A.2 SAS Data Explorer Web Application



The licensed SAS Data Explorer web application has two features that the Choose Data window does not have:

- You can set preferences for the application, as described in [“Modify SAS Data Explorer Settings” on page 304](#).
- You can display advanced data profile metrics, as described in [“Profiling Data” on page 328](#).

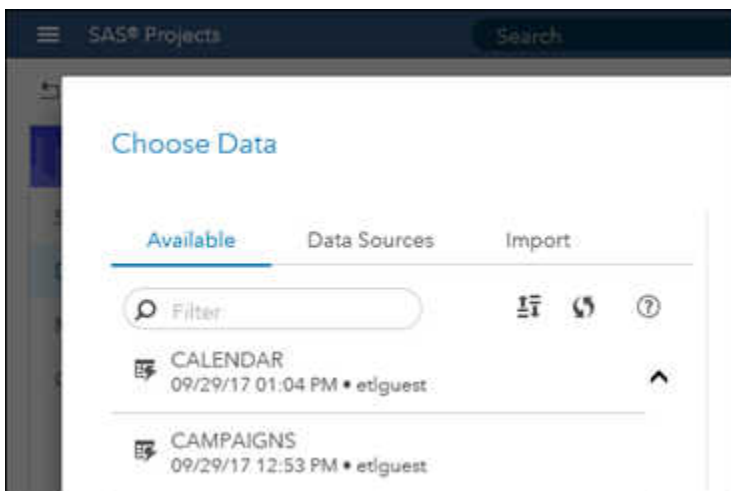
Otherwise, the licensed SAS Data Explorer web application and the Choose Data window are identical.

Getting Started with the Choose Data Window

Overview of the Choose Data Window


SAS Viya applications such as SAS Environment Manager, SAS Visual Analytics, Model Studio, and SAS Decision Manager include a window similar to the Choose Data window shown in the next figure.

Figure A.3 Choose Data Window Displayed in SAS Projects



In some applications, the Choose Data window has a different name, but this window typically includes the **Data Sources** tab and the **Import** tab. In some applications, the **Available** tab is included as well. These tabs enable you to copy data to memory on SAS Cloud Analytic Services (CAS) server and perform related tasks. To display the Choose Data window in your application, see the documentation for that application. Look for topics about importing data to CAS.

Available Tab: Work with In-Memory Tables and Files

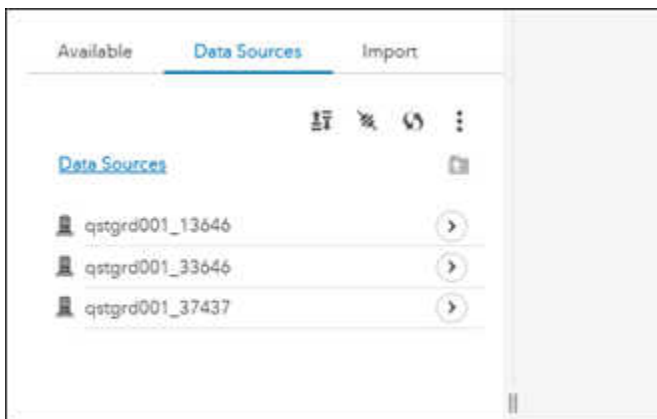
If present, the **Available** tab displays all tables and files that have been loaded to memory from any CAS server to which you have access. The items with  beside them are tables or files that have been loaded to memory on a CAS server. You can access only those tables or files that are permitted by your login. If you right-click a table or file, you can perform the tasks that are described in [“Working with Tables” on page 327](#).

Data Sources Tab: Access Databases or Remote File Systems

The **Data Sources** tab enables you to create a connection to a database server or a remote file system, such as a Hadoop Distributed File System (HDFS). If the connection is successful, tables that you are authorized to access will be available on the **Data Sources** tab. For more information about these tasks, see [“Connecting to Databases” on page 308](#) and [“Connecting to Remote File Systems” on page 310](#).

However, before you define any connections, you might want to become familiar with how the **Data Sources** tab displays CAS servers, caslibs, and tables.

Figure A.4 Choose Data Window with the Data Sources Tab Selected





The items with  beside them are the CAS servers that your login enables you to access. Click  to access the caslibs on a given server, as shown in the next figure.

Figure A.5 Caslibs on a Selected Server






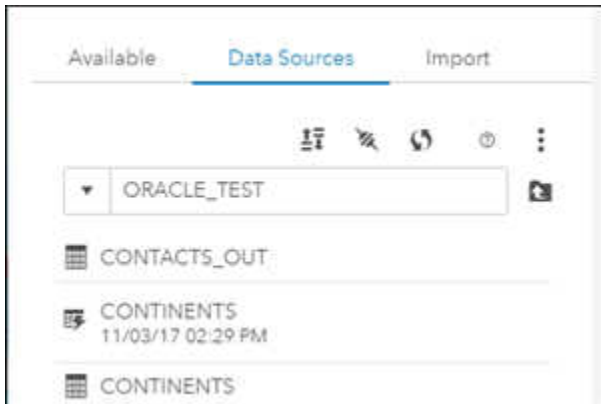


Items with  beside them are global caslibs on the selected CAS server. Items with  beside them (and with the **session** label) are session-based caslibs on the selected CAS server. For more information about global and session-based caslibs, see [“Caslibs on the Data Sources Tab and Import Tab” on page 307](#). If you right-click a caslib, you can perform the tasks that are described in [“Working with Caslibs” on page 327](#). Click  to access the tables in a selected caslib, as shown in the next figure.

Figure A.6 Tables in a Selected Caslib



The items with  beside them are physical tables that are accessible to the CAS server that have not been loaded to memory. The items with  beside them are tables that have been loaded to memory. If you right-click a table, you can perform the tasks that are described in [“Working with Tables” on page 327](#).

Import Tab: Access Local Files, Social Media Content, or Esri Data

The **Import** tab enables you to create a connection to a local file, social media content, or Environmental Systems Research Institute (Esri) data. If the connection is successful, the data will be loaded to memory on the CAS server that is specified in the connection.

The **Import** tab also enables you to copy a table or file on the **Data Sources** tab or the **Available** tab. If the connection is successful, the copy will be loaded to memory on the CAS server that is specified in the connection.

For more information, see the following topics:


- [“Importing Local Files” on page 312](#)
- [“Importing Data from Social Media” on page 316](#)
- [“Importing Esri Data” on page 324](#)
- [“Copying Data from the Available Tab or Data Sources Tab” on page 330](#)

Administrators can change the default rules and deny access to the **Import** tab or to individual social media feeds on this tab. For more information, see [“Limit Access to the Import Tab” on page 305](#).

Modify SAS Data Explorer Settings

To modify settings for the SAS Data Explorer web application:

- 1 In the application bar, click your name, and then click **Settings**. The Settings window is displayed.
- 2 The following settings are available:


Global settings. These settings are applied to all SAS web applications. Click  for details about each setting.

Data Explorer settings. These settings are saved on a per-user basis. All of your settings persist between sessions.

General settings for SAS Data Explorer.

There are two user preferences in this section.

Set default target location

Click  to select a caslib where target tables will be stored by default. This caslib will serve as the default for various operations, including operations on the **Import** tab. A change to this setting will be applied to the next table that uses the default target location. A change is not applied retroactively.

If a default target location is not set in this field, the default CAS server and caslib for the CAS Management Service is used. For more information, administrators can refer to [“CAS Management Service” on page 227](#).

Apply formats to variables when profiling data

Select this check box to apply formats to the output data for the **Run profile** option. Some data is more meaningful when it is formatted. For example, currency values might be more meaningful if they are formatted as currency rather than as integers. The impact on data profiling performance is usually acceptable. For more information about profiling, see [“Profiling Data” on page 328](#).

Geographic Mapping settings for SAS Data Explorer.

Use the options in this section to accept the terms and conditions for Esri ArcGIS Online Services. For more information about these options, see [“Importing Esri Data” on page 324](#).

Administrative Tasks for SAS Data Explorer

Overview

The tasks described in this section must be performed by an administrator. They apply to the SAS Data Explorer application and the embedded Choose Data window unless stated otherwise.

Controlling Access to Features

Limit Access to the Import Tab

By default, all authenticated users can access the **Import** tab. Administrators can limit access to the **Import** tab or to individual social media feeds on this tab. You might want to do this for two reasons:

- You do not want all users to be able to import data from Twitter, Facebook, Google Analytics, YouTube, or Google Drive.
- The **Import** tab requires users to add persistent data tables on a CAS server. You might want to limit who can do this.

To limit access to the **Import** tab or to individual social media feeds on this tab, an administrator will find the relevant rule in SAS Environment Manager. The administrator will then change the initial principal (Authenticated Users) to the ID of the group that should be able to perform the controlled function. The rules are as follows:

Import tab

/casManagement_capabilities/importData

Twitter Feed

/webDataAccess_capabilities/twitterImport

Facebook Feed

/webDataAccess_capabilities/facebookImport

Google Analytics Feed

/webDataAccess_capabilities/googleanalyticsImport

YouTube Feed

/webDataAccess_capabilities/youtubelImport

Google Drive Feed

/webDataAccess_capabilities/googledriveImport

For more information about updating rules, administrators can refer to [“Adjust Rules for Access to Functionality” on page 481](#).

Licensing

The following features are available only when your site licenses the SAS Data Preparation offering:

- The stand-alone web application SAS Data Explorer. For more information, see [“SAS Data Explorer and the Choose Data Window” on page 301](#).
- Advanced column statistics for data profiles. For more information, see [“Profiling Data” on page 328](#).

CAS Management Service Options**How to Set Service Options**

Administrators can change how some features work in a SAS Viya application by changing the service options for the application. See [“Configuration Properties: How To Configure Services” on page 215](#).

To update service options for SAS Data Explorer, administrators would enter *Explorer* in the **Filter** window for the **Environment** section of SAS Environment Manager. They would then open the SAS Data Explorer service and find the option that they want to update.

Service Options for SAS Data Explorer

The following table lists service options for SAS Data Explorer.

Table A.26 CAS Management Service Options for SAS Data Explorer

Option	Description
filterAvailableTab	<p>Turn the toggle switch on to automatically populate the Available tab with tables only from the default caslib for your site. The default is off.</p> <p>By default, the Available tab displays all tables that have been loaded to memory, from any CAS server to which you have access. If some nodes are slow to respond to queries from the Available tab, your browser might appear to hang while it is waiting for a response. If slow performance persists, an administrator can set an option so that the Available tab is automatically populated with tables only from the default caslib for your site.</p>
maxImportQueueSize	<p>The maximum number of items that can be imported with the Import All option on the Import tab. The default is 100.</p> <p>The Import All option imports all items in the import queue. If the number of items exceeds the value for maxImportQueueSize, the import will fail with an error. The value for maxImportQueueSize can be increased, if desired.</p>

Support for Third-Party Software

Unless otherwise noted, SAS Data Explorer supports the databases, the browsers, and other third-party software that is supported by SAS Viya. For more information, see [Third-Party Software Requirements for Use with SAS Viya](#).

General Usage Notes

Caslibs on the Data Sources Tab and Import Tab

Overview



A caslib is an in-memory space to hold tables, access control lists, and data source information. One property of a caslib is scope. A caslib can have one of two scopes: session scope or global scope.

Session-based caslibs enable you to work with data during the current session only. They are dropped when you sign out of the current session. Session-based caslibs do not enable you to share in-memory data with other CAS users or sessions. Global caslibs persist as long as the CAS server is running. They enable you to share in-memory data with other CAS users or sessions.

The **Data Sources** tab displays the caslibs that have been defined on a CAS server, as shown in the next figure.

Figure A.7 Caslibs on a Selected Server




Items with  beside them are global caslibs on the selected CAS server. Items with  beside them (and with the **session** label) are session-based caslibs on the selected CAS server.

Some features on the **Data Sources** tab or the **Import** tab are valid only for tables or files in global caslibs:

- **Load.** This option is used to load a table or file to memory on a CAS server.
- **Run profile.** A data profile report enables you to recognize data patterns, identify scarcity in the data, and review basic statistics for the selected table or file.
- **Create job** on the **Import** tab. This option creates import jobs that can be scheduled in SAS Environment Manager.
- Only global caslibs can be selected in the **Target destination** field when you are importing data from the **Import** tab.

Caslibs and the Data Sources Tab

The **Data Sources** tab enables you to create a new caslib that connects to a data source. The privileges associated with your login determine what type of caslib you can create. Most users have the privilege that is required to create new session-based caslibs. Most users do not have the privilege that is required to create new global caslibs.

If you click  on the **Data Sources** tab, the **Connection Settings** dialog box displays. This dialog box enables you to create a new caslib that connects to a data source.

The **Connection Settings** dialog box has an option that is called **Persist this connection beyond current session**. This check box is deselected by default. If you do not select this check box, and you specify a connection, the caslib that is created is a session-based caslib.

If you select the **Persist this connection beyond current session** option, and you specify a connection, the caslib that is created will be a global caslib. Your login must have the privilege required to create a global caslib, or the connection will fail.

You can check to see what options are available for the tables or files in the new caslib. If the saved connection uses a global caslib, the following options are available when you right-click a table in the caslib: **Load**, **Delete**, **Add to import**, and **Run profile**. For information about these options, see [“Working with Tables” on page 327](#).

If the saved connection uses a session-based caslib, the following options are valid when you right-click a table in the caslib: **Delete** and **Add to import**. You not have access to the **Load** option which is used to load a table or file to memory on a CAS server. The **Run profile** is available, but it is not valid for tables in a session-based caslib.

However, if you can access an appropriate global caslib, you can use the **Add to import** option to copy the table to that global caslib. From there, you have access to the **Load** option and the **Run profile** option. For information about the **Add to import** option, see [“Copying Data from the Available Tab or Data Sources Tab” on page 330](#).


If Data Access Fails

If you fail to access data from the **Data Sources** tab or the **Import** tab, an error message is displayed. Reasons for failure include:

- A hardware or software resource that the caslib requires is not available.
- You do not have permission to access a resource that the caslib requires.
- The caslib or the CAS server is incorrectly configured.

Contact your data administrator if the problem persists.

Refresh to Update Information about Caslibs and Tables

Information about caslibs and tables is stored in the cache for your web browser. If you think this information does not reflect the current state of your system, click  (Refresh) in the nearest toolbar.

Restart Your Application After Restarting a CAS Server

If a CAS server is restarted, log off from your application and log on to refresh the session. Otherwise, tables loaded on that CAS server will not be available, and new caslibs cannot be created on that CAS server.

Making Data Available to CAS

Connecting to Databases

Overview of Database Connections

The **Data Sources** tab enables you to create a new caslib that connects to a database server. If the connection is successful, tables that you are authorized to access in the database will be available on the **Data Sources** tab. The tables are not automatically loaded to memory.



By default, a database connection makes data available in CAS during the current session only. You must have appropriate privilege to create a connection that persists beyond the current session. For more information, see [“Caslibs on the Data Sources Tab and Import Tab” on page 307](#).



Gather the following information to create a new caslib that connects to a database server:

- Identify a name for the new caslib and the CAS server to be used in the connection.
- Identify the data source: the database that you want to access. In SAS Data Explorer, you can create a connection between a caslib and any of these supported databases: DB2, Hadoop Hive, Impala, SAS LASR Analytic Server, ODBC, Oracle, PostgreSQL, and Teradata. For information about the versions that are supported, see [SAS Viya Support for Databases](#).
- Obtain any access credentials, physical pathnames, and basic connection options that are required to make the connection to the desired database server.
- Review the documentation for any special connection options that you want to use. If you have a question about a connection option for a database, see the documentation for the database connector in [Data Connectors](#).

Create a Database Connection

Use information that you previously gathered to create a new caslib that connects to a database server:

- 1 Display the window that contains the **Data Sources** tab. See [“SAS Data Explorer and the Choose Data Window” on page 301](#).
- 2 Click the **Data Sources** tab.
- 3 Click  on the **Data Sources** tab.
The Connection Settings dialog box is displayed.
- 4 Enter a name for the caslib in the **Name** field.
Follow the name conventions described in [Variable Names and Data Set Names in CAS Engine](#).
- 5 Accept the default CAS server or select another CAS server in the **Server** field.
- 6 Select **Database** in the connection **Type** field.
- 7 Select the database that you want to access in the **Select source type** field.
- 8 Accept or change the **Persist this connection beyond the current session** check box.
The check box is deselected by default. The default setting is appropriate for most users. For more information about this option, see [“Caslibs on the Data Sources Tab and Import Tab” on page 307](#).
- 9 Using information that you gathered about this connection, enter access credentials, physical pathnames, and other connection details into the fields on the **Settings** tab.
- 10 Specify options on the **Advanced** tab, if desired.
Click  for details about each option. If you have a question about a connection option for a database, see the documentation for the database connector in [Data Connectors](#).
- 11 Click **Test Connection** to test your connection.
- 12 When ready, click **Save** to save your connection.
If the connection succeeds, tables that you are authorized to access in the database will be available from the caslib that you specified in Step 4.
If the connection fails, see [“General Usage Notes” on page 307](#).

13 If the target caslib is not visible from your current view, click  (up-one-level) or scroll (such as with ) to find the caslib and its tables on the specified CAS server.

14 Check to see what options are available for the tables or files in the new caslib.

If the saved connection uses a global caslib, the following options are available when you right-click a table in the caslib: **Load**, **Delete**, **Add to import**, and **Run profile**. For information about these options, see [“Working with Tables” on page 327](#).

If the saved connection uses a session-based caslib, the following options are valid when you right-click a table in the caslib: **Delete** and **Add to import**. You not have access to the **Load** option which is used to load a table or file to memory on a CAS server. The **Run profile** is available, but it is not valid for tables in a session-based caslib.

However, if you can access an appropriate global caslib, you can use the **Add to import** option to copy the table to that global caslib. From there, you have access to the **Load** option and the **Run profile** option. For information about the **Add to import** option, see [“Copying Data from the Available Tab or Data Sources Tab” on page 330](#).

Note: If you change an existing connection that requires a password, you must re-enter the password before saving the updated connection. The password is not retained for an updated connection.

Connecting to Remote File Systems

Overview of Remote File System Connections

The **Data Sources** tab enables you to create a new caslib that connects to one of the following remote file systems:

- DNFS (distributed network file system). Examples include MapR-FS and EMC Isilon, which are alternatives to Hadoop.
- HDFS (Hadoop Distributed File System).
- A server-side directory that a CAS controller can access.

If the connection is successful, the tables that you are authorized to access will be available on the **Data Sources** tab. The tables are not automatically loaded to memory.

By default, a remote file system connection makes data available in CAS during the current session only. You must have appropriate privilege to create a connection that persists beyond the current session. For more information, see [“Caslibs on the Data Sources Tab and Import Tab” on page 307](#).

Gather the following information to create a new caslib that connects to a remote file system:

- Identify a name for the new caslib and the CAS server to be used in the connection.
- Identify the type of remote file system that you want to access: a DNFS, an HDFS, or a server-side directory.
- Identify the physical paths, the access credentials, and the basic connection options that are required to make the connection to the data source.
- To connect to a directory, plan to create a connection for each directory that you want to access. Each **File System** connection to a directory provides access to data in that directory only, not to data in any subdirectories.
- Review the following SAS documentation for any special connection options that you want to use:

[DNFS Data Source](#)





[HDFS Data Source](#)

[Path-Based Data](#)

Note: When you use a path-based caslib to import data from a remote file system, the CAS server does not maintain pre-existing sort orders. You must re-sort the data after you import it.

Connect to a Remote File System

Use information that you previously gathered to create a new caslib that connects to a remote file system:

- 1 Display the window that contains the **Data Sources** tab. See [“SAS Data Explorer and the Choose Data Window” on page 301](#).
- 2 Click the **Data Sources** tab.
- 3 Click  on the **Data Sources** tab. The Connection Settings dialog box is displayed.
- 4 Enter a name for the caslib in the **Name** field.
Follow the name conventions described in [Variable Names and Data Set Names in CAS Engine](#).
- 5 Accept the default CAS server or select another CAS server in the **Server** field.
- 6 Select **File System** in the connection **Type** field.
- 7 Select the remote file system that you want to access in the **Select source type** field: **DNFS**, **HDFS**, or **PATH**.
- 8 Accept or change the **Persist this connection beyond the current session** check box.
The check box is deselected by default. The default setting is appropriate for most users. For more information about this option, see [“Caslibs on the Data Sources Tab and Import Tab” on page 307](#).
- 9 Enter a pathname to the remote file system in the **Path** field.
- 10 Enter a **Description** for the connection, if desired.
- 11 Do not select the **Include subdirectories** option. Each **File System** connection to a directory provides access to data in that directory only, not to data in any subdirectories.
- 12 Specify options on the **Advanced** tab, if desired. Click  for details about each option.
- 13 Click **Test Connection** to test your connection.
- 14 When ready, click **Save** to save your connection.
If the connection succeeds, tables that you are authorized to access in the remote directory will be available from the caslib that you specified in Step 4.
If the connection fails, see [“General Usage Notes” on page 307](#).
- 15 If the target caslib is not visible from your current view, click  (up-one-level) or scroll (such as with ) to find the caslib and its tables on the specified CAS server.
- 16 Check to see what options are available for the tables or files in the new caslib.
If the saved connection uses a global caslib, the following options are available when you right-click a table in the caslib: **Load**, **Delete**, **Add to import**, and **Run profile**. For information about these options, see [“Working with Tables” on page 327](#).
If the saved connection uses a session-based caslib, the following options are valid when you right-click a table in the caslib: **Delete** and **Add to import**. You not have access to the **Load** option which is used to load a table or file to memory on a CAS server. The **Run profile** is available, but it is not valid for tables in a session-based caslib.
However, if you can access an appropriate global caslib, you can use the **Add to import** option to copy the table to that global caslib. From there, you have access to the **Load** option and the **Run profile** option. For

information about the **Add to import** option, see “Copying Data from the Available Tab or Data Sources Tab” on page 330.

Importing Local Files

Overview of Importing Local Data Files

Local files are available from the file system on your PC. This includes local file systems such as `c:\` on Windows machines. Network file systems and shared folders are also included, such as UNC paths like `\\nas\spreadsheets`.

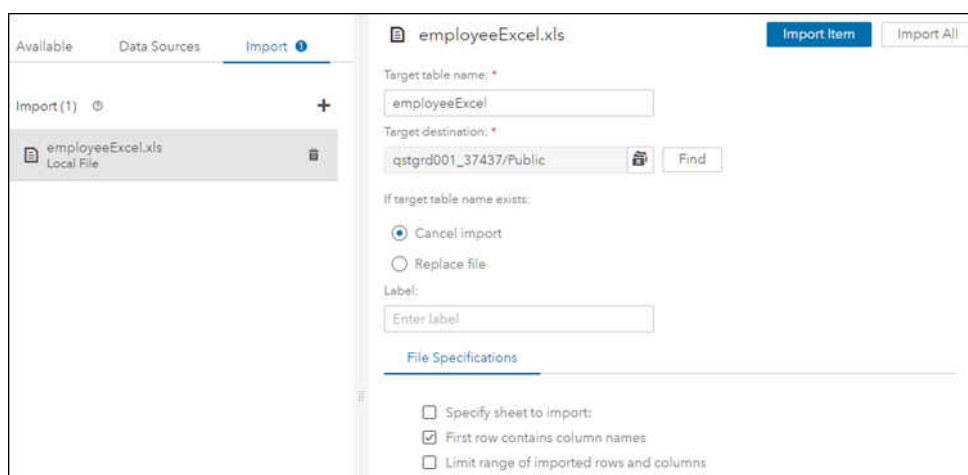
The **Local File** option on the **Import** tab enables you to copy a local file and load it to memory on a CAS server.

Here are the supported file formats:

- Comma-delimited (CSV) text files or TXT files.
- SAS data sets (SASHDAT or SAS7BDAT). SAS data set views (SAS7BVEW) cannot be loaded into CAS tables.
- Microsoft Excel workbook (XLSX) files and Excel 97-2003 workbook (XLS) files. You cannot import XLST, XLSB, XLSM, or other Excel file types. You cannot import pivot tables.

If you drag and drop a local file onto the **Import** tab, it is added to the queue on the left side, as shown in the following figure.

Figure A.8 Excel Spreadsheet in the Queue on the Import Tab



Import properties for the selected file appear on the right. A default table name and caslib for the copy that you will create are provided in the **Target table name** field and the **Target destination** field. Several file specifications for the type of file that you are copying are provided at the bottom of the import properties.

When ready, you can right-click the file in the import queue and select **Import Item**. A copy of the file is loaded to memory on the CAS server that is specified in the **Target destination** field.

Gather the following information to import a local file:

- Identify the local file that you want to load to memory. Verify that the file is in a supported file format.
- Identify the physical path to the local file. Note the size of the file that you want to import. Importing a large table or file can be time-consuming. See “Notes for All Local Data Files” on page 315.
- Gather information about any special options that might be needed to accurately copy the contents of a file. For example, to read a delimited text file, you might have to specify a delimiter value other than the default

delimiter, which is a comma. To review the import options for a file type, see Step 8 in [“Import Local Microsoft Excel, Text, or SAS Data Set Files”](#) on page 313.

Note: Any items in the queue on the **Import** tab will be dropped from the queue when you log off. The target files that you load to memory will persist beyond the current session. The targets are associated with a global caslib in the import properties. For more information about global caslibs, see [“Caslibs on the Data Sources Tab and Import Tab”](#) on page 307.

Import Local Microsoft Excel, Text, or SAS Data Set Files

Use information that you previously gathered to copy a local file and load it to memory on a CAS server:

- 1 Display the window that contains the **Import** tab.

See [“SAS Data Explorer and the Choose Data Window”](#) on page 301.

- 2 Click the **Import** tab.

- 3 Drag and drop a local file onto the **Import** tab. Alternatively, click **Local File** on the **Import** tab. Use the file selection window to navigate to a local file and select it.

The file is added to the queue on the **Import** tab. Import properties for the selected file appear on the right.

- 4 The name of the source was copied into the **Target table name** field on the right. Accept or change this name as appropriate for the copy.

If you change the name, follow the conventions described in [Variable Names and Data Set Names in CAS Engine](#). The target table name can be a maximum of 247 characters long. However, the name might have to be shorter, due to the data source settings for the caslib that is specified in the **Target destination** field. Contact your data administrator for more information about the permitted length for target table names.

- 5 A default caslib is provided in the **Target destination** field. If this is not where you want to store the file, you can select an existing global caslib or create a new one.

If you have the privilege required to create a new global caslib, follow the name conventions described in [Variable Names and Data Set Names in CAS Engine](#). For more information about global caslibs, see [“Caslibs on the Data Sources Tab and Import Tab”](#) on page 307.

- 6 Specify what action the import operation should take if the target filename exists in the caslib that was specified in the **Target destination** field. The options are to cancel the import or to replace the existing item that has the same name.

- 7 Specify a label that will help identify the copy, if desired.

- 8 Specify any special options for the file that you are importing.

Microsoft Excel spreadsheet options:

The following options are available for Microsoft Excel spreadsheets:

Specify sheet to import

For spreadsheets that contain multiple worksheets, indicate the name of the worksheet that you want to import. If you do not specify a worksheet name, then only the first worksheet in the spreadsheet is imported.

First row contains column names

Select this check box when the first row in the file contains column names.

Limit range of imported rows and columns

Select this check box to limit the import operation to a range of cells in the spreadsheet. To use this field, you must also indicate the name of the worksheet in the **Specify sheet to import** field, even if the spreadsheet contains only one worksheet.

Note: If you select both **Limit range of imported rows and columns** and **First row contains column names**, the first row will be the first row in the specified range.

Text file options:

The following options are available for text files:

Input file delimiter

Select the delimiter that is used in the file that you want to import. The default value is a comma. If you select **Custom**, you can specify a single character to use as a user-defined delimiter.

Scanned rows

Select the number of rows to scan in order to determine column data types and lengths. A smaller value causes the import to complete quickly, but you increase the possibility of obtaining a value that is too small to accommodate character columns. A larger value reduces the possibility of truncating character columns, but it increases processing time.

Locale

Enter the locale code for the file. An example is **fr-FR**. Specifying the locale code is important so that special characters (for example, commas) in the file are interpreted correctly based on locale. If you leave this field blank, it defaults to **en-US**.

For examples of locale code values, see [Table of Language Culture Names, Codes, and ISO Values](#).

Encoding

Enter the encoding of the file. When importing UTF-8 or UTF-16 data, make sure that SAS Web Application Server is a Unicode server or that the file contents can be transcoded to the encoding of SAS Web Application Server. Examples of valid values include **utf8**, **utf16le**, and **euc-cn**. The default value is **utf8**.

First row contains column names

Select this check box when the first row in the file contains column names.

Convert character columns to variable size

Select this check box if the file includes columns that contain varying-length character strings. When the file is imported, this option assigns the varying-length character string (VARCHAR) format to all columns that contain character data.

CAUTION! If you select this check box, the VARCHAR format is applied to all columns that contain character data, rather than to only those with varying-length character strings.

SAS data set, SAS7BDAT options:

The following options are available for SAS7BDAT files:

Password

If the file is password-protected, enter the password for the file. If the file is password-protected and you do not enter the password in this field, then the file will fail to import.

Encryption key

If the file is encrypted, enter the encryption key for the file. Otherwise, the file will fail to import.

Character multiplier

Modify this option to increase the number of characters that can fit in each cell so that character data truncation does not occur. The lengths for character variables can be increased by multiplying the current length by the value that you specify. You can specify a multiplier value from 1 to 5. The default value is 2.

SAS data set, SASHDAT options:

The following options are available for SASHDAT files:

Encryption key

If the file is encrypted, enter the encryption key for the file. Otherwise, the file will fail to import.

- 9 When ready, you can right-click the file to be copied and select **Import Item**.

If the import succeeds, a copy of the file is loaded to memory on the CAS server that is specified in the caslib. The copy of the file can be selected from the **Available** tab or the **Data Sources** tab.

If the import fails, see [“Usage Notes for Importing Local Data Files”](#) on page 315. See also [“General Usage Notes”](#) on page 307.

Usage Notes for Importing Local Data Files

Notes for All Local Data Files

You can experience long wait times when you import large local files through a web browser. Accordingly, the CAS Management service option `maxFileUploadSize` is set to 4 Gb by default. If the import of a local file fails because the file size is too large, it might be because you have exceeded the `maxFileUploadSize` value for the CAS server. Your administrator can change the value of this option as appropriate for your site.

To update CAS Management service options, administrators would enter *CAS Management* in the **Filter** window for the **Environment** section of SAS Environment Manager. They would then open the CAS Management service and find the `maxFileUploadSize` option. For more information, see [“Configuration Properties: How To Configure Services”](#) on page 215.

Notes for Importing Text Files

Here are some key points about importing text files:

- When you import a text file, any data that is enclosed in quotation marks will be identified as the VARCHAR type.
- To represent a missing value in the table, you must leave the field empty instead of entering a period (.). Fields that contain a period (.) will be imported as a character type.

Notes for Importing SAS Data Sets

Here are some key points about importing SAS data sets:

- Before importing a SAS data set that uses user-defined formats, ensure that the custom format catalog is available to the CAS server. For more information, see [Manage User-Defined Formats](#).
- Importing indexed SAS data sets is not supported.

Table Names, Column Names, and Special Characters

In general, you can import files that use blanks and special characters in the filenames and column names. The following list identifies how table names are handled:

- For text files or SAS data sets, the table name is initially set from the filename.
- When you import a spreadsheet, table names are handled as follows:
 - If the spreadsheet contains a single worksheet, then the output table name is initially set as the filename.
 - If the spreadsheet contains multiple worksheets, then the output table name is initially set as the filename for the first worksheet. To indicate a different worksheet, enter the worksheet name in the **Sheet name** field in the Options window for the file.
 - Some special characters can be used, including spaces. Unsupported special characters include / \ * ? " < > | : - , and period (.). If you include unsupported special characters in the output name for a table, the import fails.

If you clear the **Include column names** check box or the **First row contains column names** check box, then the column names are generated as follows:

- **Spreadsheets** Column names are assigned A, B, C, and so on.
- **Text files** Column names are assigned VAR1, VAR2, and so on.

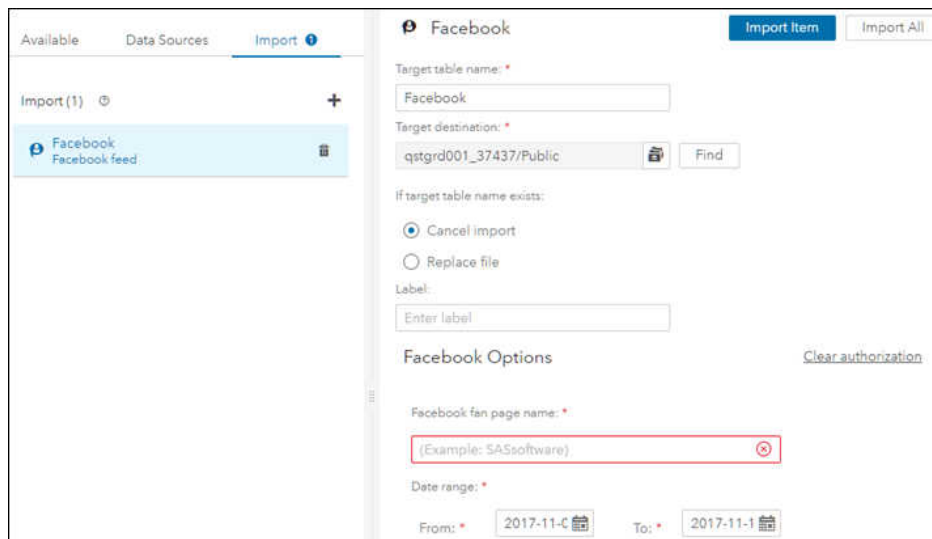
Importing Data from Social Media

Overview of Importing Data from Social Media

The **Social Media** folder on the **Import** tab enables you to import data from social media accounts and to load it to memory on a CAS server. The data can then be analyzed in SAS Visual Analytics and other SAS Viya applications.

For example, if you select the **Facebook feed** option in the **Social Media** folder, a **Facebook feed** item is added to the queue on the left side, as shown in the next figure.

Figure A.9 Facebook Feed Item in the Queue on the Import Tab



An import properties window appears on the right side. A default table name and a caslib are provided in the **Target table name** and **Target destination** fields. You can use the Facebook options in the window to select the data that you want from the Facebook feed.

When ready, you can right-click the **Facebook feed** item in the import queue and select **Import Item**. A copy of the specified Facebook data is loaded to memory on the CAS server that is specified in the caslib.

If the import fails, see [“General Usage Notes” on page 307](#).

Note: Any items in the queue on the **Import** tab will be removed from the queue when you log off. The target tables that you load to memory will persist beyond the current session. The tables are associated with a global caslib in the import properties window. For more information about global caslibs, see [“Caslibs on the Data Sources Tab and Import Tab” on page 307](#).

Administrators can change the default rules and deny access to the **Import** tab or to individual social media feeds on this tab. For more information, see [“Limit Access to the Import Tab” on page 305](#).

Import Data from Twitter

The **Twitter feed** option on the **Import** tab enables you to copy data from a Twitter account and load it to memory on a CAS server.

Before you can import information from Twitter, you must grant SAS the right to access your account. Perform these steps:

- 1 Display the window that contains the **Import** tab.

See “[SAS Data Explorer and the Choose Data Window](#)” on page 301.

2 Click the **Import** tab.

3 Expand the **Social Media** folder and select **Twitter feed**.

A message is displayed, saying that you must grant SAS Visual Analytics the right to access your account. A **Proxy Server** option is provided.

4 If your site uses an HTTP Proxy Server to access the internet, enter the proxy host and port. Otherwise, ignore the proxy settings.

5 When ready, click **OK**.

A Twitter authorization page is displayed in a new browser tab.

6 Enter the user name and password for the Twitter account and click **Authorize app**.

After you have granted SAS the right to access a Twitter account, perform the following steps to load data from that account to memory on a CAS server:

1 On the **Import** tab, expand the **Social Media** folder and select **Twitter feed**.

A **Twitter feed** item is added to the queue on the left side. An import properties window appears on the right.

2 The **Target table name** is **Twitter** by default. Accept or change this name as appropriate.

If you change the name, follow the conventions described in [Variable Names and Data Set Names in CAS Engine](#). The target table name can be a maximum of 247 characters long. However, the name might have to be shorter, due to the data source settings for the caslib that is specified in the **Target destination** field. Contact your data administrator for more information about the permitted length for target table names.

3 A default caslib is provided in the **Target destination** field. To store the data from Twitter elsewhere, you can select an existing global caslib or you can create a new one.

If you have the privilege required to create a new global caslib, follow the name conventions described in [Variable Names and Data Set Names in CAS Engine](#). For more information about global caslibs, see “[Caslibs on the Data Sources Tab and Import Tab](#)” on page 307.

4 Specify what action the import operation should take if the target filename exists in the caslib that is specified in the **Target destination** field. The options are to cancel the import operation or to replace the existing item that has the same name.

5 Specify a label that will help identify the output table of Twitter information, if desired.

6 Use the settings under **Twitter Options** to select data from the Twitter account.

Use the **Search term** field to select a list of tweets that include the specified term. The search operators that you can use are described on the [Twitter Search page](#).

Use other Twitter options as needed. If your site uses an HTTP Proxy Server to access the internet, enter the proxy host and port. Otherwise, ignore the proxy settings.

The **Clear authorization** option enables you to delete the SAS authorization for this Twitter account. If you delete this information, you must reauthorize the account before you can import Twitter data.

7 When ready, you can right-click the **Twitter feed** item in the queue and select **Import Item**.

If the import succeeds, the table of Twitter data is loaded to memory on the CAS server that is specified in the caslib. The table of Twitter data can be selected from the **Available** tab or the **Data Sources** tab.

If the import fails, see “[General Usage Notes](#)” on page 307.

Import Data from a Facebook Page

The **Facebook feed** option on the **Import** tab enables you to copy data from a Facebook account and load it to memory on a CAS server.

Before you can import information from Facebook, you must grant SAS the right to access your account. Perform these steps:

- 1 Display the window that contains the **Import** tab.

See [“SAS Data Explorer and the Choose Data Window” on page 301](#).

- 2 Click the **Import** tab.

- 3 Expand the **Social Media** folder and select **Facebook feed**.

A message is displayed, indicating that you must grant SAS Visual Analytics the right to access Facebook information.

- 4 Click **OK**.

To load data from a Facebook page to memory on a CAS server:

- 1 On the **Import** tab, expand the **Social Media** folder and select **Facebook feed**.

A **Facebook feed** item is added to the queue on the left side. Import properties appear on the right side.

- 2 The **Target table name** is **Facebook** by default. Accept or change this name as appropriate.

If you change the name, follow the conventions described in [Variable Names and Data Set Names in CAS Engine](#). The target table name can be a maximum of 247 characters long. However, the name might have to be shorter, due to the data source settings for the caslib that is specified in the **Target destination** field. Contact your data administrator for more information about the permitted length for target table names.

- 3 A default caslib is provided in the **Target destination** field. To store the data from Facebook elsewhere you can select an existing global caslib or you can create a new one.

If you have the access rights that are required to create a new global caslib, follow the name conventions described in [Variable Names and Data Set Names in CAS Engine](#). For more information about global caslibs, see [“Caslibs on the Data Sources Tab and Import Tab” on page 307](#).

- 4 Specify what action the import operation should take if the target filename exists in the caslib that is specified in the **Target destination** field. The options are to cancel the import operation or to replace the existing item that has the same name.

- 5 Specify a label that will help identify the output table of Facebook information, if desired.

- 6 Use the settings under **Facebook Options** to select data from a Facebook page.

Use the **Facebook fan page name** field to specify the name of the page from which you will import data. The name appears at the end of the Facebook URL, such as **MyFanPage** at the end of this URL: `https://www.facebook.com/MyFanPage`.

Use other Facebook options as needed. If your site uses an HTTP Proxy Server to access the internet, enter the proxy host and port.

The **Clear authorization** option enables you to delete the SAS authorization for this Facebook account. If you delete this information, you must reauthorize the account before you can import Facebook data.

- 7 When ready, you can right-click the **Facebook feed** item in the queue and select **Import Item**.

If the import succeeds, the table of Facebook data is loaded to memory on the CAS server that is specified in the caslib. The table of Facebook data can be selected from the **Available** tab or the **Data Sources** tab.


If the import fails, see [“General Usage Notes” on page 307](#).

Import Data from Google Analytics

Overview of Importing Data from Google Analytics

The **Google Analytics feed** option on the **Import** tab enables you to copy data from a Google Analytics account and load it to memory on a CAS server.

Before you can import information from Google Analytics, you must grant SAS the right to access your account. Perform these steps:

- 1 Display the window that contains the **Import** tab.
See [“SAS Data Explorer and the Choose Data Window” on page 301](#).
- 2 Click the **Import** tab.
- 3 Expand the **Social Media** folder and select **Google Analytics feed**.
A message is displayed, indicating that you must grant SAS Visual Analytics the right to access your Google Analytics account.
- 4 Click **Obtain Access Code**.
A new browser tab opens. Respond as prompted to sign in to your Google Analytics account, to provide authorization to SAS, and then to copy the access code.
- 5 Return to the tab where you clicked **Obtain Access Code**. Paste the access code into the text box on this tab.
- 6 If your site uses an HTTP Proxy Server to access the internet, enter the proxy host and port. Otherwise, ignore the proxy fields.
- 7 When ready, click **OK**.
A **Google Analytics feed** item is added to the queue on the **Import** tab. If you are ready to import information, see [“Import Data from Google Analytics” on page 319](#).. Otherwise, click  to remove the **Google Analytics feed** item from the queue.

Import Data from Google Analytics

After you have granted SAS the right to access your Google Analytics account, perform the following steps to load data from that account to memory on a CAS server:

- 1 On the **Import** tab, expand the **Social Media** folder and select **Google Analytics feed**.
A **Google Analytics feed** item is added to the queue on the left side. Import properties appear on the right.
- 2 The **Target table name** is **Google Analytics** by default. Accept or change this name as appropriate.
If you change the name, follow the conventions described in [Variable Names and Data Set Names in CAS Engine](#). The target table name can be a maximum of 247 characters long. However, the name might have to be shorter, due to the data source settings for the caslib that is specified in the **Target destination** field. Contact your data administrator for more information about the permitted length for target table names.
- 3 A default caslib is provided in the **Target destination** field. To store the data from Google Analytics elsewhere, you can select an existing global caslib or you can create a new one.

If you have the privilege that are required to create a new global caslib, follow the name conventions described in [Variable Names and Data Set Names in CAS Engine](#). For more information about global caslibs, see [“Caslibs on the Data Sources Tab and Import Tab” on page 307](#).

- 4 Specify what action the import operation should take if the target filename exists in the caslib that is specified in the **Target destination** field. The options are to cancel the import operation or to replace the existing item that has the same name.
- 5 Specify a label that will help identify the output table of Google Analytics information, if desired.
- 6 Use the settings under **Google Analytics Options** to select data from the account. To choose useful options, you must understand the information that is available in this account.

If your site uses an HTTP Proxy Server to access the internet, enter the proxy host and port. Otherwise, ignore the proxy settings.

The **Clear authorization** option enables you to delete the SAS authorization for this Google Analytics account. If you delete this information, you must reauthorize the account before you can import Google Analytics data.

- 7 When ready, you can right-click the **Google Analytics feed** item in the queue and select **Import Item**.
If the import is successful, the table of Google Analytics data is loaded to memory on the CAS server that is specified in the caslib. The table of Google Analytics data can be selected from the **Available** tab or the **Data Sources** tab.

If the import fails, see [“Usage Notes for Importing Local Data Files” on page 315](#).


Importing Data from YouTube

Overview of Importing Data from YouTube

The **YouTube feed** option on the **Import** tab enables you to copy data from a YouTube account and load it to memory on a CAS server.

Before you can import information from YouTube, you must grant SAS the right to access your account. Perform these steps:

- 1 Display the window that contains the **Import** tab.
See [“SAS Data Explorer and the Choose Data Window” on page 301](#).
- 2 Click the **Import** tab.
- 3 Expand the **Social Media** folder and select **YouTube feed**.
A message is displayed, indicating that you must grant SAS Visual Analytics the right to access your YouTube account.
- 4 Click **Obtain Access Code**.
A new browser tab opens. Respond as prompted to sign in to your YouTube account, to provide authorization to SAS, and then to copy the access code.
- 5 Return to the tab where you clicked **Obtain Access Code**. Paste the access code into the text box on this tab.
- 6 If your site uses an HTTP Proxy Server to access the internet, enter the proxy host and port. Otherwise, ignore the proxy fields.
- 7 When ready, click **OK**.

A **YouTube feed** item is added to the queue on the **Import** tab. If you are ready to import information, see [“Import Data from YouTube” on page 321](#).. Otherwise, click  to remove the **YouTube feed** item from the queue.

Import Data from YouTube

After you have granted SAS the right to access your YouTube account, perform the following steps to load data from that account to memory on a CAS server:

- 1 On the **Import** tab, expand the **Social Media** folder and select **YouTube feed**.
A **YouTube feed** item is added to the queue on the left side. Import properties appear on the right.
- 2 The **Target table name** is **YouTube** by default. Accept or change this name as appropriate.
If you change the name, follow the conventions described in [Variable Names and Data Set Names in CAS Engine](#). The target table name can be a maximum of 247 characters long. However, the name might have to be shorter, due to the data source settings for the caslib that is specified in the **Target destination** field. Contact your data administrator for more information about the permitted length for target table names.
- 3 A default caslib is provided in the **Target destination** field. To store the data from YouTube elsewhere, you can select an existing global caslib or you can create a new one.
If you have the access rights that are required to create a new global caslib, follow the name conventions described in [Variable Names and Data Set Names in CAS Engine](#). For more information about global caslibs, see [“Caslibs on the Data Sources Tab and Import Tab” on page 307](#).
- 4 Specify what action the import operation should take if the target filename exists in the caslib that is specified in the **Target destination** field. The options are to cancel the import operation or to replace the existing item that has the same name.
- 5 Specify a label that will help identify the output table of YouTube information, if desired.
- 6 Use the settings under **YouTube Options** to select data from the account. To choose useful options, you must understand the metrics, the dimensions, and other options that are available for this account.
If your site uses an HTTP Proxy Server to access the internet, enter the proxy host and port. Otherwise, ignore the proxy options.
The **Clear authorization** option enables you to delete the SAS authorization for this YouTube account. If you delete this information, you must reauthorize the account before you can import YouTube data.
- 7 When ready, you can right-click the **YouTube feed** item in the queue and select **Import Item**.
If the import succeeds, the table of YouTube data is loaded to memory on the CAS server that is specified in the caslib. The table of YouTube data can be selected from the **Available** tab or the **Data Sources** tab.
If the import fails, see [“General Usage Notes” on page 307](#).

Importing Files from Google Drive

Overview of Importing Files from Google Drive

The **Google Drive feed** option on the **Import** tab enables you to copy a file from Google Drive and load it to memory on a CAS server. Supported file formats are:

- Comma-delimited (CSV) text files, TXT files, and Google Sheets (gsheet) files.
- SAS data sets (SASHDAT or SAS7BDAT).
- Excel workbook (XLSX) files, Excel 97-2003 workbook (XLS) files, and macro-enabled spreadsheets (XLSM) files. You cannot import XLST, XLSB, or other Excel file types. You cannot import pivot tables.

Before you can import a file from Google Drive, you must grant SAS the right to access your account. Perform these steps:

- 1 Display the window that contains the **Import** tab.
See [“SAS Data Explorer and the Choose Data Window”](#) on page 301.
- 2 Click the **Import** tab.
- 3 Expand the **Social Media** folder and select **Google Drive feed**.
A message is displayed, indicating that you must grant SAS Visual Analytics the right to access your Google Drive account.
- 4 Click **Obtain Access Code**.
A new browser tab opens. Respond as prompted to sign in to your Google Drive account, to provide authorization to SAS, and then to copy the access code.
- 5 Return to the tab where you clicked **Obtain Access Code**. Paste the access code into the text box on this tab.
- 6 If your site uses an HTTP Proxy Server to access the internet, enter the proxy host and port. Otherwise, ignore the proxy fields.
- 7 When ready, click **OK**.
The **Import from Google Drive** window appears. If you are ready to import a file, see [“Import a File from Google Drive”](#) on page 322. Otherwise, click **Cancel** from this window.

Import a File from Google Drive

After you have granted SAS the right to access your Google Drive account, you can copy a file from Google Drive and load it to memory on a CAS server. Perform these steps,

- 1 On the **Import** tab, expand the **Social Media** folder and select **Google Drive feed**.
The Import from Google Drive window appears.
- 2 Navigate to the file that you want to import. Click the file, and then click **Select**.
A **Google Drive feed** item for the selected file is added to the queue on the left side. Import properties appear on the right.
- 3 The **Target table name** is the name of the selected table by default. Accept or change this name as appropriate.
If you change the name, follow the conventions described in [Variable Names and Data Set Names in CAS Engine](#). The target table name can be a maximum of 247 characters long. However, the name might have to be shorter, due to the data source settings for the caslib that is specified in the **Target destination** field. Contact your data administrator for more information about the permitted length for target table names.
- 4 A default caslib is provided in the **Target destination** field. If this is not where you want to store the file from Google Drive, you can select an existing global caslib or create a new one.
If you have the privilege required to create a new global caslib, follow the name conventions described in [Variable Names and Data Set Names in CAS Engine](#). For more information about global caslibs, see [“Caslibs on the Data Sources Tab and Import Tab”](#) on page 307.
- 5 Specify what action the import operation should take if the target filename exists in the caslib that is specified in the **Target destination** field. The options are to cancel the import or to replace the existing item that has the same name.

- 6 Specify a label that will help identify the output table of Google Drive information, if desired.
- 7 Use the settings under **File Specification Options** to specify options for the imported file.

Spreadsheet options for Microsoft Excel or Google Sheets:

The following options are available for these spreadsheets:

Specify sheet to import

For spreadsheets that contain multiple worksheets, indicate the name of the worksheet that you want to import. If you do not specify a worksheet name, then only the first worksheet in the spreadsheet is imported.

First row contains column names

Select this check box when the first row in the file contains column names.

Limit range of imported rows and columns

Select this check box to limit the import operation to a range of cells in the spreadsheet. To use this field, you must also indicate the name of the worksheet in the **Specify sheet to import** field, even if the spreadsheet contains only one worksheet.

Note: If you select both **Limit range of imported rows and columns** and **First row contains column names**, the first row will be the first row in the specified range.

Text file options:

The following options are available for text files:

Input file delimiter

Select the delimiter that is used in the file that you want to import. The default value is a comma. If you select **Custom**, you can specify a single character to use as a user-defined delimiter.

Scanned rows

Select the number of rows to scan in order to determine column data types and lengths. A smaller value causes the import to complete quickly, but you increase the possibility of obtaining a value that is too small to accommodate character columns. A larger value reduces the possibility of truncating character columns, but it increases processing time.

Locale

Enter the locale code for the file. An example is **fr-FR**. Specifying the locale code is important so that special characters (for example, commas) in the file are interpreted correctly based on locale. If you leave this field blank, it defaults to **en-US**.

For examples of locale code values, see [Table of Language Culture Names, Codes, and ISO Values](#).

Encoding

Enter the encoding of the file. When importing UTF-8 or UTF-16 data, make sure that SAS Web Application Server is a Unicode server or that the file contents can be transcoded to the encoding of SAS Web Application Server. Examples of valid values include **utf8**, **utf16le**, and **euc-cn**. The default value is **utf8**.

First row contains column names

Select this check box when the first row in the file contains column names.

Convert character columns to variable size

Select this check box if the file includes columns that contain varying-length character strings. When the file is imported, this option assigns the varying-length character string (VARCHAR) format to all columns that contain character data.

CAUTION! If you select this check box, the VARCHAR format is applied to all columns that contain character data, rather than to only those with varying-length character strings.

SAS data set, SAS7BDAT options:

The following options are available for SAS7BDAT files:

Password

If the file is password-protected, enter the password for the file. If the file is password-protected and you do not enter the password in this field, then the file will fail to import.

Encryption key

If the file is encrypted, enter the encryption key for the file. Otherwise, the file will fail to import.

Character multiplier

Modify this option to increase the number of characters that can fit in each cell so that character data truncation does not occur. The lengths for character variables can be increased by multiplying the current length by the value that you specify. You can specify a multiplier value from 1 to 5. The default value is 2.

SAS data set, SASHDAT options:

The following options are available for SASHDAT files:

Encryption key

If the file is encrypted, enter the encryption key for the file. Otherwise, the file will fail to import.

If your site uses an HTTP Proxy Server to access the internet, enter the proxy host and port. Otherwise, ignore the proxy fields.

The **Clear authorization** option enables you to delete the SAS authorization for this Google Drive account. If you delete this information, you must reauthorize the account before you can import Google Drive data.

- 8 When ready, you can right-click the **Google Drive feed** item in the queue and select **Import Item**.

If the import succeeds, the file is loaded to memory on the CAS server that is specified in the caslib. The file can be selected from the **Available** tab or the **Data Sources** tab.

If the import fails, see [“Usage Notes for Importing Local Data Files” on page 315](#).

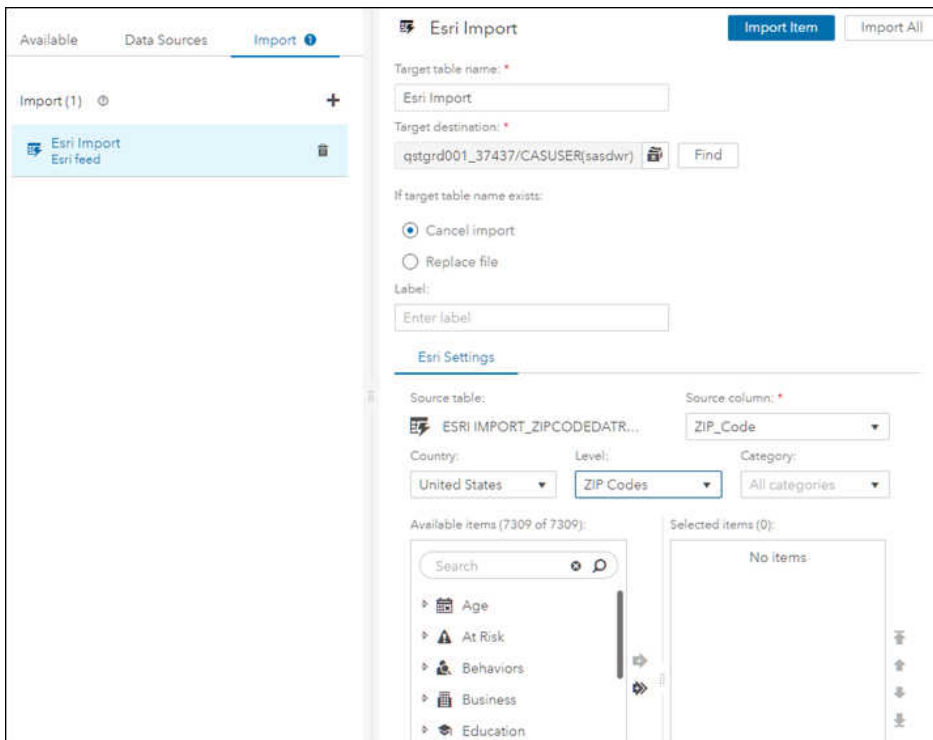
Importing Esri Data

Overview of Importing Esri Data

The Geo Enrichment Service from Environmental Systems Research Institute (Esri) provides a large collection of data sets, including population, income, housing, consumer behavior, and the natural environment. You can use this service to combine Esri enrichment data with data from a table that you select in your CAS environment. The resulting output table contains new columns of Esri data that are associated with geographic location codes in the source table. For example, you could combine Esri demographic information for the ZIP codes listed in a source table. The resulting output table could then be analyzed in SAS Visual Analytics or other SAS Viya applications.

If your login has appropriate privilege, you can select the **Esri** ⇒ **Geo Enrichment** option on the **Import** tab. You are prompted to select an in-memory source table in your CAS environment. After you select the source table, an **Esri Import** item is added to the queue on the left, as shown in the next figure.

Figure A.10 Esri Feed Item in the Queue on the Import Tab



Import properties for the Esri feed appear on the right. A default table name and caslib are provided in the **Target table name** and **Target destination** fields. This is the name and caslib of the output table that will combine Esri data with data from the selected source table. You can use the options under **Esri Settings** to select the type of enrichment data that you want to add to the information from the source table.

When ready, you can right-click the **Esri Import** item in the import queue and select **Import Item**. The specified Esri data is combined with data from the selected source table and is loaded to memory on the CAS server that is specified in the **Target destination** field.

Note: Any items in the queue on the **Import** tab will be dropped from the queue when you log off. The target tables that you load to memory will persist beyond the current session. The tables are associated with a global caslib in the import properties window. For more information about global caslibs, see [“Caslibs on the Data Sources Tab and Import Tab” on page 307](#).

Perform the following tasks before using the **Geo Enrichment** option:

- Ask an administrator to add your login to the custom group **Esri Users** in SAS Environment Manager. For information about custom groups, administrators can refer to [“Getting Started with Identity Management” on page 476](#).
- Obtain the user name and password for the Esri Geo Enrichment Service.
- Access the Settings window for your application. Select **Settings** at top right of the application banner. Select both Esri options in the **Geographic Mapping** section of the window. Specify the user name and password for the Geo Enrichment Service in the **Esri Premium Services Credentials** section.
- Identify the caslib and name of the source table to which you will add Esri data. The source table for an **Esri Import** item must be an in-memory table.
- Identify the caslib and name of the target table that will combine Esri data with data from the selected source table.
- Identify the geographic level that you want to analyze, such as by state or by ZIP code.

- Identify the Esri enrichment data that will support the analysis that you want to perform on the output table. For example, do you want to analyze demographic data, business data, or landscape data by geographic location? For more information about Esri enrichment data, see [The Esri GeoEnrichment Service](#).

Import Esri Data

After you have done the prerequisite tasks, perform the following steps to combine Esri enrichment data with data from an in-memory table on a CAS server:

- 1 On the **Import** tab, expand the **Esri** folder and select **Esri Import**.
You are prompted to select a source table in your CAS environment. Only in-memory tables are available for selection.
- 2 Select a source table.
After you select the source table, an **Esri Import** item is added to the queue on the left. Import properties appear on the right.
- 3 The **Target table name** is **Esri Import** by default. This is the name of the output table that will combine Esri enrichment data with data from the selected source table. Accept or change this name as appropriate.
If you change the name, follow the conventions described in [Variable Names and Data Set Names in CAS Engine](#). The target table name can be a maximum of 247 characters long. However, the name might have to be shorter, due to the data source settings for the caslib that is specified in the **Target destination** field. Contact your data administrator for more information about the permitted length for target table names.
- 4 A default caslib is provided in the **Target destination** field. If this is not where you want to store the output table, you can select an existing global caslib or create a new one.
If you have the privilege required to create a new global caslib, follow the name conventions described in [Variable Names and Data Set Names in CAS Engine](#). For more information about global caslibs, see [“Caslibs on the Data Sources Tab and Import Tab” on page 307](#).
- 5 Specify what action the import operation should take if the target filename exists in the caslib specified in the **Target destination** field. The options are to cancel the import or to replace the existing item that has the same name.
- 6 Specify a label that will help identify the output table, if desired.
- 7 Use the options under **Esri Settings** to select Esri enrichment data that will be combined with the data in the source table.

Source table

The in-memory table whose data will be combined with Esri enrichment data.

Source column

Select the column in the source table that specifies geographic location codes, such as a column for ZIP codes.

Country

Select the country where the geographic location codes in the **Source column** are valid.

Level

Select the Esri geographic level that corresponds to the geographic location codes in the **Source column** above. Levels include **States**, **Census Tracts**, and **ZIP Codes**.

For example, suppose that the source table has a column named ZIP_Code, and you want to add Esri demographic information for the ZIP codes listed in the source table. You could select **ZIP_Code** in the **Source column** field and select the Esri level that is named **ZIP Codes** in the **Level** field.

Category

Select a category of Esri enrichment data. Categories include **Business**, **Households**, and **Population**.

For example, suppose that you want to add Esri demographic information for ZIP codes, as specified in the **Level** field. You could select **Population** in the **Category** field.

Available items

Select items within the category selected in the **Category** field. For example, the **Population** category includes items for **Gender** and **Language**. Click the arrow to move an available item to the **Selected items** panel.

Selected items

Enrichment items that will be combined with information from the source table.

For example, if the **Gender** item appears in this panel, Esri gender information will be combined with information from the source table. Gender information would be provided by the geographic level that is specified in the **Level** field, such as **ZIP Codes**.

- 8 If you want to add more than one Esri category of information, go back to the **Category** field and add another category and items for that category.
- 9 When ready, you can right-click the **Esri Import** item in the queue and select **Import Item**.

If the import succeeds, Esri enrichment data will be combined with the data in the source table and written to the output table. The output table will be loaded to memory on the CAS server that is specified in the **Target destination** field. The output table can be selected from the **Available** tab or the **Data Sources** tab.

If the import fails, see [“Usage Notes for Importing Local Data Files”](#) on page 315.

Working with Data in CAS

Working with Caslibs and Tables

Working with Caslibs

When you right-click a caslib in the **Data Sources** tab, the following options are available.

Table A.27 Pop-up Menu Options for a Selected Caslib

Name	Action
Set as default target location	Makes the selected caslib the default location for various operations, including operations on the Import tab. This option is available only for global caslibs. See “Caslibs on the Data Sources Tab and Import Tab” on page 307.
Remove	Removes the selected caslib from memory on a CAS server.

Working with Tables

When you select a table in the **Available** tab or the **Data Sources** tab, three tabs of information about this table are displayed on the right side:

- **Details.** Displays basic information about the table, such as the number of columns and rows and the data type of columns.
- **Sample Data.** Displays sample data from the selected table.

- **Profile.** Displays a report that enables you to recognize data patterns, to identify scarcity in the data, and to review basic statistics for the selected table. Data profiles can be generated only for tables in a global caslib. For more information, see [“Profiling Data” on page 328](#).

If you right-click a table in the **Available** tab or the **Data Sources** tab, some of the options in the next table will be available, as appropriate for the context.

Table A.28 Pop-up Menu Options for a Selected Table on the Available Tab or Data Sources Tab

Name	Action
Actions ⇨ Prepare Data, etc.	Opens the selected table in another application such as SAS Data Studio or SAS Visual Analytics. For more information, see the documentation for that application.
Load	Loads the selected table to memory on a CAS server. The Load option is available only for tables in a global caslib. For more information about global caslibs, see “Caslibs on the Data Sources Tab and Import Tab” on page 307 .
Unload	Unloads the selected table from memory on a CAS server. The table must be loaded again if you want to work with it in CAS.
Delete	For in-memory tables: Delete unloads the table from memory and removes its associated physical table. For physical tables: Delete removes the physical table.
Add to import	Adds the selected table to the queue that is managed on the Import tab. You might want to do this to create a copy of a table on the Available tab or the Data Sources tab. For more information, see “Copy Data from the Data Sources Tab or Available Tab” on page 331 .
Run profile	Generates a report that enables you to recognize data patterns, to identify scarcity in the data, and to review basic statistics for the selected table. Data profiles can be generated only for tables in a global caslib. For more information, see “Profiling Data” on page 328 .

If you right-click a table or file in the queue on the **Import** tab, and the **Create job** option is available, you can use this option to create an import job. The job can be scheduled or executed in SAS Environment Manager. For more information, see [“Create Jobs” on page 333](#).

Profiling Data

Profile a Table

A data profile report enables you to recognize data patterns, identify scarcity in the data, and review basic statistics for the selected table. The table must be associated with a global caslib. For more information about global caslibs, see [“Caslibs on the Data Sources Tab and Import Tab” on page 307](#).

Perform the following steps to generate a data profile report for a table associated with a global caslib:

- 1 Select a table in the **Available** tab or the **Data Sources** tab.
Three tabs of information about this table are displayed: **Details**, **Sample Data**, and **Profile**.
- 2 Click the **Profile** tab.
If a profile was previously generated for the table, the most recent report is displayed. If a profile is not available, click **Run Profile**.

A profile report is generated for the table. Column statistics in the report include: Unique, Mean, Standard Deviation, Minimum, Maximum, Data Type, and Data Length.

If your site has licensed the SAS Data Preparation offering, additional statistics will be generated: Null, Frequency Distribution, Pattern Analysis, Minimum Length, Maximum Length, Ordinal Position, Primary Key Candidate, and Non-null Count. The next figure shows a data profile.

Figure A.11 Data Profile for a Selected Table

Column	Unique	Null	Blank	Pattern Count
Failure Date	47%	0	0	4
Institution Name	91% (3185)	0	0	1,524

- Note the buttons at the top right corner of the profile report: (Options) and (Version).
 - By default, columns for all statistics are included in the profile. To remove some statistics from the profile, click and select **Manage columns**. Use the arrows in the window to remove some statistics from the profile. Click **OK** to display the revised profile.
 - Each time you profile a table, a version of the report is saved. Click to view versions of the profile report. Click to remove outdated profile reports.

Usage Notes for Data Profiles

Profiling Tables with a Large Number of Columns

If you profile a table that has thousands of columns of data, the profile might require more memory than is available on your CAS server. In that case, you could try profiling a version of the table that had fewer columns. For information about the **Select Columns** tab, see [“Copying Data from the Available Tab or Data Sources Tab” on page 330](#).

Frequency Distribution and Pattern Metrics Might Be Truncated

When you profile a table, the length of the frequency distribution and pattern metrics will be truncated to match the profile internal process table. This table is currently set to 1000 characters.

Profiling Multiple Files with the Same Name but Different Extensions

If one directory contains multiple files with the same name but different extensions, the CAS Management Service will decide which one is preferred. The SASHDAT version might be preferred over the CSV version, for example. In this case, only the file with the preferred extension can be profiled. If you want to profile multiple files with the same name but different extensions, put the files with different extensions in different directories.

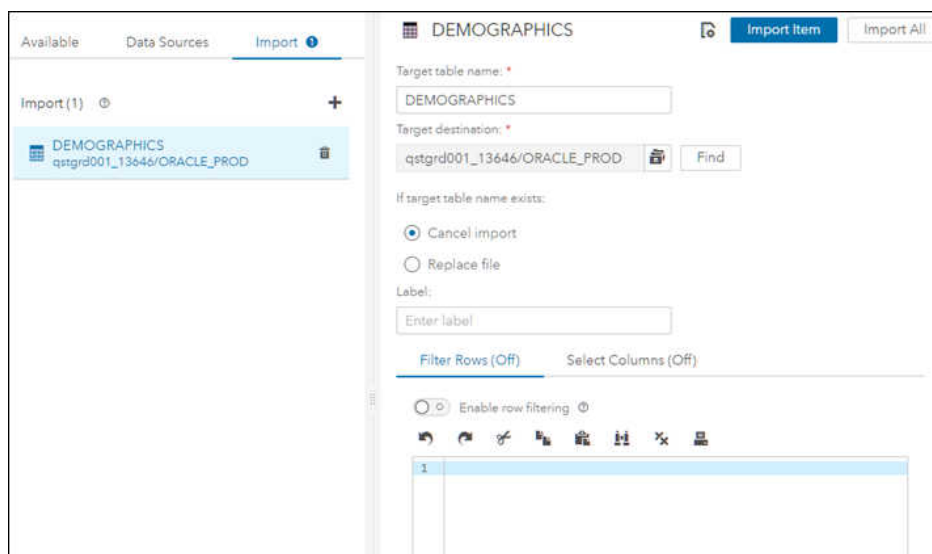
Copying Data from the Available Tab or Data Sources Tab

Overview of the Add to Import Option

The **Add to import** option enables you to copy a table or a file on the **Available** tab or the **Data Sources** tab and to load it to memory on a CAS server.

When you right-click a table or file that you want to copy, and select **Add to import**, the table or file is added to the queue on the **Import** tab. For example, in the next figure, a physical table has been added to the import queue on the left side.

Figure A.12 Physical Table in the Queue on the Import Tab



Note the **Target table name** field and the other import properties on the right. These properties apply to the copy of the table or file that will be loaded to memory on a CAS server. By default, the name in the **Target table name** field is the same as the name of the source table or file. The caslib in the **Target destination** field is the same as the caslib for the source table or file. You can change these values as needed.

Note the **Filter Rows** tab and **Select Columns** tab in the previous figure. These tabs are available for physical tables or files only, not for in-memory tables or files. You can use these tabs to create a copy that contains a subset of columns or rows from the original.

When ready, you can right-click the table or file in the import queue and select **Import Item**. A copy of the table or file is loaded to memory on the CAS server that is specified in the target caslib.

The **Import Item** option copies the table or file in the current session. For large items, you might want to schedule the import after business hours to minimize the impact on production systems. If you right-click the item in the import queue, and the **Create job** option is available, you can use this option to create an import job. The import job can be scheduled or executed in SAS Environment Manager.

Gather the following information to use the **Add to import** option:

- Identify the tables or files on the **Available** tab or the **Data Sources** tab that you want to copy. Note the size of the item that you want to import. Importing a large table or file can be time-consuming.
- For physical tables or files only, consider creating a copy that has a subset of columns or rows from the original. Plan which rows or columns would be appropriate for the copy. For more information, see [“Filter](#)

[Rows from an Imported Table or File](#)” on page 332 or [“Select Columns from an Imported Table or File”](#) on page 332.


Note: Any items in the queue on the **Import** tab will be dropped from the queue when you log off. The target tables or files that you load to memory will persist beyond the current session. The targets are associated with a global caslib in the import properties. For more information about global caslibs, see [“Caslibs on the Data Sources Tab and Import Tab”](#) on page 307.

Copy Data from the Data Sources Tab or Available Tab

Use information that you previously gathered to copy a table or a file on the **Available** tab or the **Data Sources** tab:

- 1 Display the window that contains the **Available** tab or the **Data Sources** tab. See [“SAS Data Explorer and the Choose Data Window”](#) on page 301.
- 2 Click the **Available** tab or the **Data Sources** tab.
- 3 Right-click a table or file that you want to copy and select **Add to import**. The selected table or file is added to the queue on the **Import** tab.
- 4 Click the **Import** tab. The table or file that you want to copy appears on the left side. Import properties for the copy appear on the right side.
- 5 The name of the source was copied into the **Target table name** field on the right. Accept or change this name as appropriate for the copy. If you change the name, follow the conventions described in [Variable Names and Data Set Names in CAS Engine](#). The target table name can be a maximum of 247 characters long. However, the name might have to be shorter, due to the data source settings for the caslib that is specified in the **Target destination** field. Contact your data administrator for more information about the permitted length for target table names.

Note: The name of the copy cannot be the same as the original unless you change the caslib or select the **Replace file** option.


- 6 The caslib for the source was copied into the **Target destination** field on the right. If this is not where you want to store the copy, you can select an existing global caslib or create a new one.
If you have the privilege that is required to create a new global caslib, follow the name conventions described in [Variable Names and Data Set Names in CAS Engine](#). For more information about global caslibs, see [“Caslibs on the Data Sources Tab and Import Tab”](#) on page 307.
- 7 Specify what action the import operation should take if the target filename exists in the caslib that is specified in the **Target destination** field. The options are to cancel the import or to replace the existing item that has the same name.
- 8 Specify a label that will help identify the copy, if desired.
- 9 If the item to be copied is a physical item (a table or file with  beside it), the **Filter Rows** tab and **Select Columns** tab are displayed on the right side. You can use these tabs to subset columns or rows from the original. For more information about these tabs, see [“Filter Rows from an Imported Table or File”](#) on page 332 and [“Select Columns from an Imported Table or File”](#) on page 332.
- 10 When ready, you can run the import operation. Do one of the following:
 - You can right-click the table or file to be copied and select **Import Item**. If the import succeeds, a copy of the table or file is loaded to memory on the CAS server that is specified in the caslib. The copy of the table or file can be selected from the **Available** tab or the **Data Sources** tab.
If the import fails, see [“General Usage Notes”](#) on page 307.

- If you right-click the table or file in the import queue, and the **Create job** option is available, you can use this option to create an import job. The import job can be scheduled or executed in SAS Environment Manager. For more information, see [“Create Jobs” on page 333](#). After the import job is run, a copy of the table or file is loaded to memory on the CAS server that is specified in the caslib. The copy can be selected from the **Available** tab or the **Data Sources** tab.

Filter Rows from an Imported Table or File

As described in [“Copy Data from the Data Sources Tab or Available Tab” on page 331](#), you can right-click a physical table or file and select **Add to import**. The table or file is added to the queue on the **Import** tab. The **Filter Rows** tab appears on the right side.

You can use the **Filter Rows** tab to enter the argument for a SAS WHERE expression that selects and copies rows from the selected table or file. For a syntax reference, see [Syntax of WHERE Expression](#). The SAS expression is translated into native database syntax and is pushed down to the database, as appropriate.

The data source must be a physical item (a table or file with  beside it). The table or file must support a WHERE clause. Supported databases include DB2, Hadoop (Hive), Impala, Oracle, PostgreSQL, Teradata, and databases accessed with ODBC. Other supported data sources include SAS data sets, SASHDAT files, SAS LASR files, Microsoft Excel files, and delimited text files. Social media files and Esri data are not supported.

To select and copy rows from a physical table or file in the queue on the **Import** tab:

- 1 If you have not done so already, perform steps 1–9 as described in [“Copy Data from the Data Sources Tab or Available Tab” on page 331](#).
- 2 Select **Enable row filtering** on the **Filter Rows** tab.
- 3 Enter the argument for the SAS WHERE expression that selects and copies rows from the table or file to be copied. Omit the WHERE= part of the expression. Use double quotation marks to surround column names with special characters, as shown in the following example:


```
("$Sal" between 1000 and 3000) and ("comm@" is NULL)
```

- 4 When ready to test your code, you can right-click the table or file to be copied and select **Import Item**. If the import succeeds, a copy of the table or file is loaded to memory on the CAS server that is specified in the **Target destination** field. The copy of the table or file can be selected from the **Available** tab or the **Data Sources** tab.

If the import fails, see [“Usage Notes for Importing Local Data Files” on page 315](#).

Select Columns from an Imported Table or File

As described in [“Copy Data from the Data Sources Tab or Available Tab” on page 331](#), you can right-click a physical table or file and select **Add to import**. The table or file is added to the queue on the **Import** tab. The **Select Columns** tab appears on the right.

By default, all columns from the table or file will be copied to the target table. You can use the **Select Columns** tab to select a subset of columns from the selected table or file. The data source must be a physical item (a table or file with  beside it).

To select a subset of columns from a physical table or file in the queue on the **Import** tab:

- 1 If you have not done so already, perform steps 1–9 as described in [“Copy Data from the Data Sources Tab or Available Tab” on page 331](#).
- 2 Select **Enable column selection** on the **Select Columns** tab. By default, all columns from the table or file are listed in the **Selected columns** panel.

3 To select a subset of columns, use the arrows to move columns from the **Selected columns** panel to the **Available columns** panel. You can use the up and down arrows to rearrange the columns in the **Selected columns** panel.

4 When you are ready to copy the table or file, select **Import Item**.

If the import is successful, a copy of the table or file is loaded to memory on the CAS server that is specified in the **Target destination** field. A message that indicates a successful operation is displayed above the import properties on the right. The copy of the table or file can be selected from the **Available** tab or the **Data Sources** tab.

Create Jobs

If you right-click a table or file in the queue on the **Import** tab, and the **Create job** option is available, you can use this option to create an import job. The job can be scheduled or executed in SAS Environment Manager. You might want to do this if the item to be imported is large, and you want to minimize the impact on production systems. The **Create job** option is not valid for local files, social media feeds, or Esri data. It is valid only for tables or files that have been copied from the **Available** tab or the **Data Sources** tab, using the **Add to import** option. These tables or files must be associated with a global caslib. For more information about global caslibs, see [“Caslibs on the Data Sources Tab and Import Tab” on page 307](#).

To select a subset of columns from a physical table or file in the queue on the **Import** tab:

1 If you have not done so already, perform steps 1–9 as described in [“Copy Data from the Data Sources Tab or Available Tab” on page 331](#).

2 Select the table or file in the queue on the **Import** tab to copy with the **Create job** option.


3 Verify that the import properties on the right are correct for the copy that you will create. For example, the name specified in the **Target table name** field cannot be the same as the original unless you have selected the **Replace file** option.

4 When ready, right-click the table or file and select the **Create job** option. The Create Job window is displayed.

5 Specify a name and description for the job and click **OK**. A status message is displayed, stating whether the job was created successfully.

If the job was successfully created, the next step is to locate the job in SAS Environment Manager.

6 From the application bar, click the applications menu button at the top left corner. Select **Manage Environment**. SAS Environment Manager is displayed.

7 Click  (Scheduling) in the navigation bar on the left. The Scheduling window is displayed. Any jobs that you have created will appear in the **Jobs** table.

8 Right-click a job and select **Run** or **Schedule**.

If you select **Run**, the job will run immediately. A status message is displayed, stating whether the job ran successfully. Skip to Step 11.

If you select **Schedule**, the Schedule Job window is displayed.

9 In the Schedule Job window, click **+** to add a trigger for the job. The New Trigger window is displayed.

10 In the New Trigger window, specify a date and time at which the job should be executed and click **Save**. For more information about scheduling, see [Scheduling on page 587](#).

11 After the import job runs, a copy of the table or file is loaded to memory on the CAS server that is specified in the caslib. The copy can be selected from the **Available** tab or the **Data Sources** tab.

12 (Optional) If you have access to SAS Job Monitor, you can check the status of the import job. From the application banner, click the applications menu button at the top left side. Select **Monitor Jobs**. SAS Job Monitor is displayed.

13 If an import job has run at least once, it appears in the Jobs window.

SAS Cloud Analytic Services: Fundamentals

Introduction

What Is SAS Cloud Analytic Services?

- SAS Cloud Analytic Services is a server that is suitable for both on-premises and cloud deployments. The server provides the run-time environment for data management and analytics. By run-time environment, we refer to the combination of hardware and software where data management and analytics take place.
- The server can run on a single machine or as a distributed server on multiple machines. The distributed server consists of one controller, an optional backup controller, and one or more workers. This architecture is often referred to as a massively parallel processing architecture. For both architectures, the server is multi-threaded for high-performance analytics.
- The distributed server has a communication layer that supports fault tolerance. A distributed server can continue processing requests even after losing connectivity to some nodes. The communication layer also enables you to remove or add worker nodes from a server while it is running.
- SAS Studio provides a programming environment for developing and submitting SAS programs to the server.
- The SAS Scripting Wrapper for Analytics Transfer (SWAT) enables open-source software such as Python, Lua, and R to run data analysis on the server. For Java, classes are provided to enable connections to the server and classes are provided to run data analysis.

About Memory Management

One of the design principles of the server is to handle large problems and to fully exploit the scalability of modern cluster computing hardware. In order to address this principle, data in the server is managed in blocks.

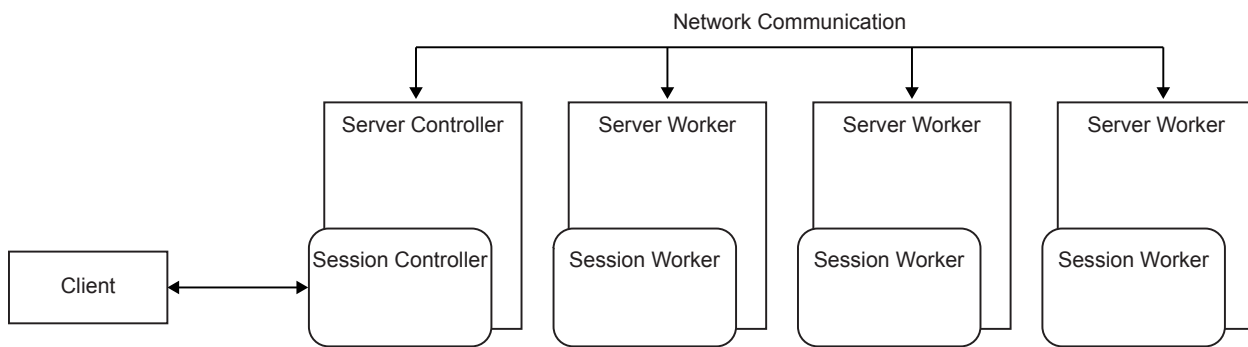
Whenever needed, the server caches the blocks on disk. It is this feature that enables the server to manage memory efficiently, handle large data volumes, and remain responsive to requests. More information is provided in the sections about data access and memory.

Architecture

Distributed Server

A distributed server uses multiple machines to perform massively parallel processing (MPP). The following figure depicts the server topology for a distributed server:

Figure A.1 Distributed Server Architecture



- One machine is designated as the controller. Client applications communicate with the controller and the controller coordinates the processing that is performed by the worker nodes.
- One or more machines are designated as worker nodes. Each worker node performs data analysis on the rows of data that are in-memory on the node.
- The server scales horizontally. If processing times are unacceptably long due to large data volumes, more machines can be added as workers to distribute the workload.
- Distributed servers are fault tolerant. If communication with a worker node is lost, a surviving worker node uses a redundant copy of the data to complete the data analysis.
- Whenever possible, distributed servers load data into memory in parallel. This provides the fastest load times.

Distributed Server: Controller Fault Tolerance

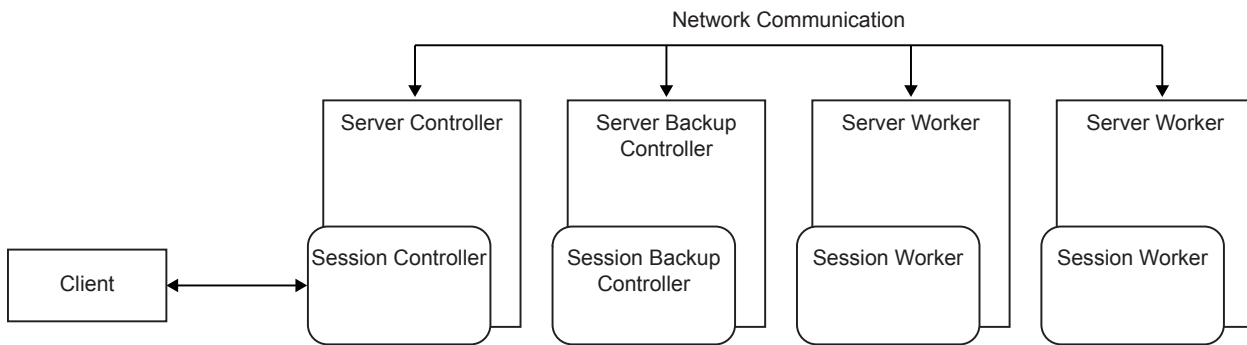
Overview

Support for a backup controller is added in the SAS Viya 3.3 release.

- Controller failover provides continuity of the service for the server and its global state when the primary controller fails. New connections to the server can be made through the backup controller. In some cases, clients can failover transparently—without any indication that connection has changed over to the backup controller.
- Global-scope in-memory tables remain loaded and are unaffected by failover. However, for data that is not in-memory at the time of failover, see [“Data Availability Considerations”](#).
- A backup controller is an optional feature. A backup controller applies to distributed servers only. You cannot add a backup controller to a running server.
- One backup controller is supported. When the primary controller fails, the site operates without fault tolerance for the controller until a planned outage.

The following figure depicts a distributed server with a backup controller:

Figure A.2 Distributed Server with a Backup Controller



How Failover is Detected

- The primary controller synchronizes with the backup controller continuously. The updates enable the backup controller to provide service rapidly.
- The primary controller sends *heartbeat* messages. A failover occurs when internode network communication does not receive a heartbeat message from the primary controller within the lost heartbeat time-out interval.

For more information, see `env.CAS_HEARTBEAT_LOST_TIMEOUT` in [SAS Cloud Analytic Services: Reference](#).

After Failover (Binary Protocol)

Most SAS Viya applications use the binary protocol. These applications look up the controller host name from Consul. Servers with a backup controller register themselves in Consul when they start and unregister when they stop. Most significantly, when the controller fails over, the backup controller updates the controller host address in Consul with its own host name. As a result, SAS Viya applications connect to the controller at the new address. For a few minutes during the failover, applications that are running CAS actions can become unresponsive. Connected users might need to discard their sessions (such as closing a browser) and connect again.

Programming clients such as SAS, Python, Java, and so on, can use binary protocol too. However, these clients do not use Consul to look up the connection information. These clients connect to the server by specifying the host name and network port of the CAS controller. These clients can accept both the primary controller and backup controller host names at connection time. Specifying both enables the client to switch automatically from the primary to the backup.

Be aware that the automatic connection to the backup controller for the programming clients requires that the network return an error. When the client attempts to connect to the failed primary controller, the network error is the trigger to connect to the backup controller. However, in some network topologies, when a host is lost, network communication can cause a long delay while a network time-out expires. The duration of the delay is controlled by the network software of the client operating system. In some cases, the delay can be up to 15 minutes. In these cases, the client is unresponsive until the time-out expires.

After Failover (REST and HTTP Protocol)

The REST interface provided by the server and CAS Server Monitor rely entirely on the Apache HTTP proxy to redirect clients from the failed controller to the backup controller. In a full deployment, the HTTP proxy is used and monitors the controller address that is registered in Consul. After failover, the backup controller updates the host name in Consul and HTTP requests are automatically directed to the backup controller.

During the failover, REST clients and CAS Server Monitor are likely to experience errors or unresponsiveness while the network communication software of the client operating system times out. After the failover is complete and the proxy has switched over, connections that use REST and HTTP can resume.

Programming-only Deployment Limitations

SAS offers a programming-only deployment of SAS Viya. This deployment is smaller and simpler. Consul is included in programming-only deployments, but the HTTP proxy is not dynamic. Using a backup controller is not recommended in programming-only deployments if the programming clients use the REST interface or if CAS Server Monitor is used for administration.

For information about a programming-only deployment, see [SAS Viya Administration: Orientation](#).

Data Availability Considerations

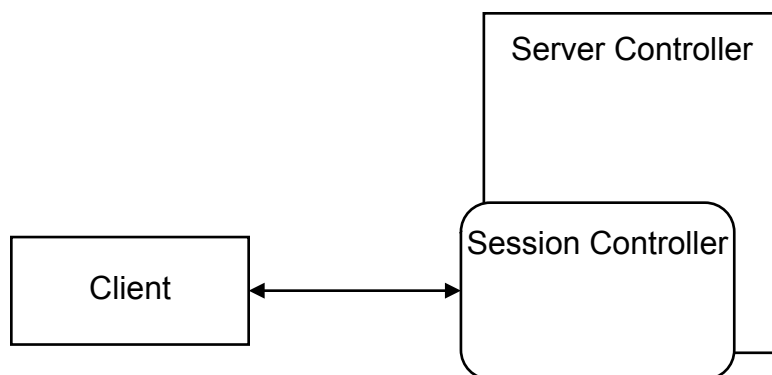
- For caslibs that use a path as the data source, the directory must be a network location that is available at the same path on both controller hosts.
- For server-based caslibs that use database drivers, the drivers must be available on both controller hosts.
- If user-defined format libraries are saved to path-based caslibs, then the directory must be a network location that is available at the same path on both controller hosts.

For more information, see [SAS Viya Administration: SAS Cloud Analytic Services](#).

Single-Machine Server

The following figure depicts the server topology for a single-machine server:

Figure A.3 Single-Machine Server Architecture



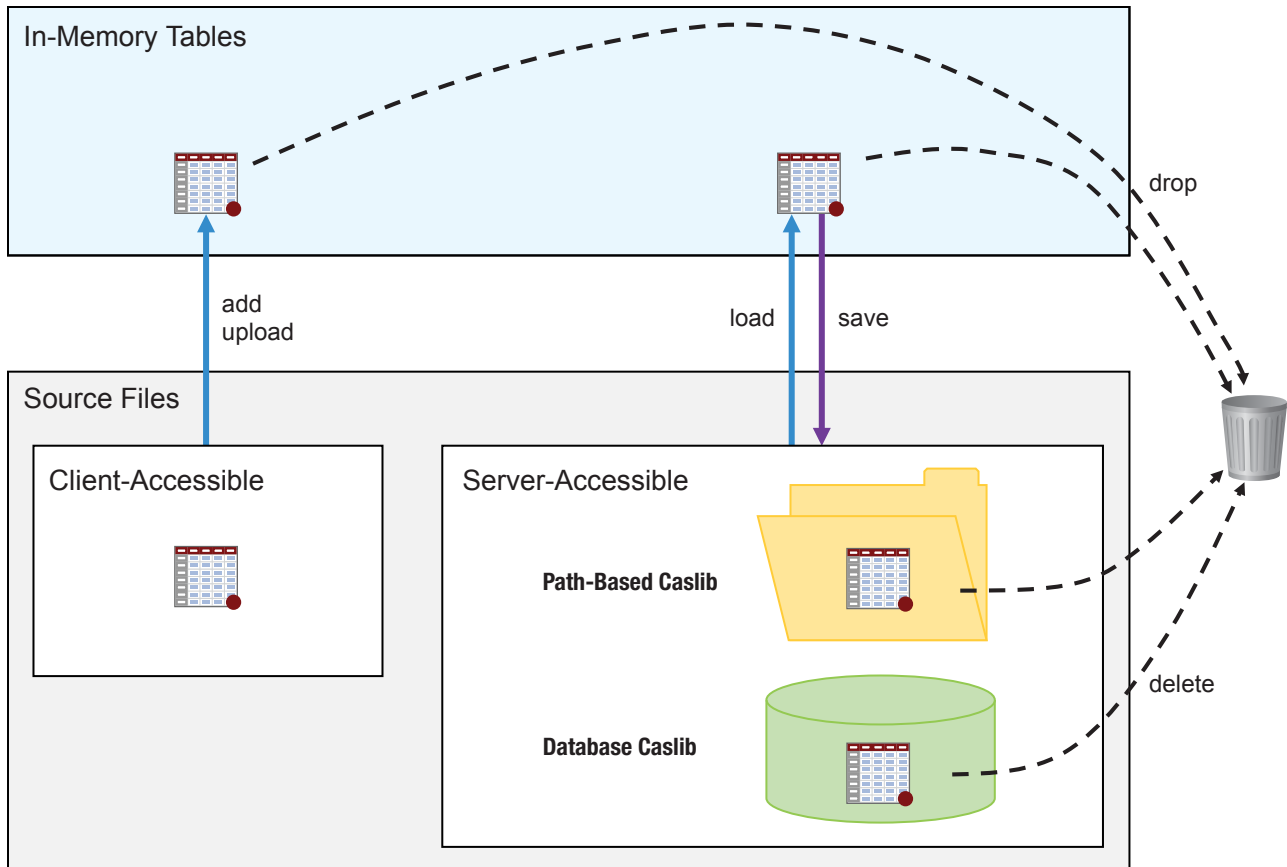
- The single machine is designated as the controller. Because there are no worker nodes, the controller node performs data analysis on the rows of data that are in-memory.
- The single machine uses multiple CPUs and threads to speed up data analysis. This architecture is often referred to as symmetric multi-processing (SMP).
- All the in-memory analytic features of a distributed server are available to the single-machine server.
- Single-machine servers cannot load data into memory in parallel from any data source.

Data

Data Lifecycle

The following graphic summarizes fundamental concepts of the data lifecycle for SAS Cloud Analytic Services:

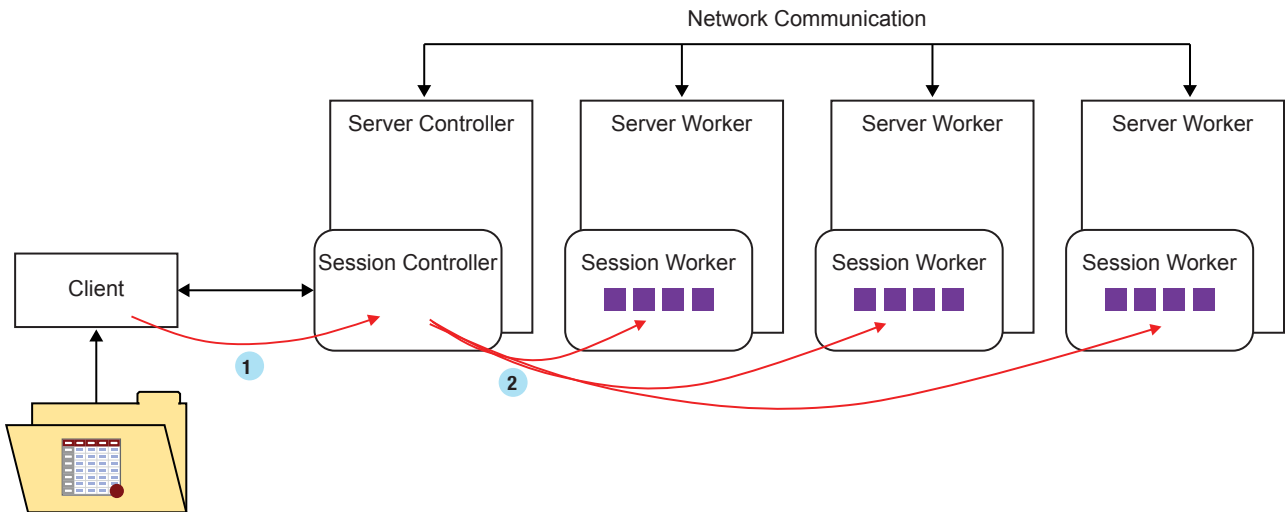
- SAS Cloud Analytic Services operates on in-memory tables. Clients can add or upload data. The server can load data from server-side data sources.
- The server can save and load data from server-side data sources.
- When an in-memory table is dropped, it does not affect a persisted file. Persisted files are removed from the data source by deleting them.



Client-Side Data Access

The following figure depicts a client application, such as SAS Studio, transferring data to a distributed server. The following SAS language elements operate this way:

- A DATA step with the CAS LIBNAME engine
- The LOAD DATA= form of the CASUTIL procedure



- 1 The client reads the file and transfers the data serially, in chunks, to the session controller.
- 2 The session controller receives the data from the client and distributes rows to the worker nodes. The rows are distributed in round-robin fashion to the workers.

Table A.29 Advantages and Disadvantages of Client-Side Data Loading

Advantages	Disadvantages
This is a very straightforward way to load data into the server.	<ul style="list-style-type: none"> ■ Large files can require a long time to transfer between the client and the server. ■ After the data is in memory, you should save it to a caslib's data source.

Server-Side Data Access

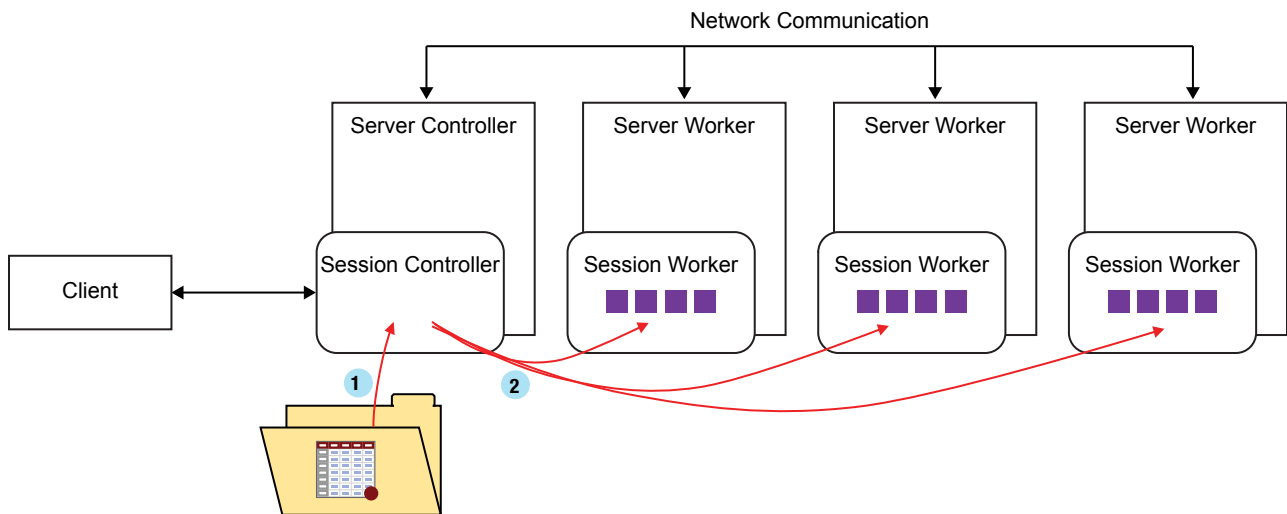
Serial Data I/O

A key feature of SAS Cloud Analytic Services and caslibs is that a caslib has an association with a data source. This is such a valuable feature to the server that caslibs associated with paths and caslibs associated with server-based data sources support server-side loading of data.

The `LOAD CASDATA=` form of the `CASUTIL` procedure is used to perform a server-side data load.

The following figure depicts how a server loads a SAS data set (or any file) that is accessible to the controller node only.

Figure A.4 Server-Side Load of a Data File



- 1 Because the file is accessible to the controller node only, the controller reads the file from the caslib's data source.
- 2 The controller node distributes rows to the worker nodes.

Parallel Data I/O

The following figure also depicts a server-side data load. This representation shows parallel data access between the data source and the worker nodes. This pattern applies to all caslib data sources that make the data available to the worker nodes. These data sources are as follows:

- Some path-based data sources:
 - Distributed NFS (DNFS)
 - Hadoop Distributed File System (HDFS)
- Data sources that support a SAS Data Connect Accelerator and SAS Embedded Process is installed.

Figure A.5 Server-Side Parallel Load

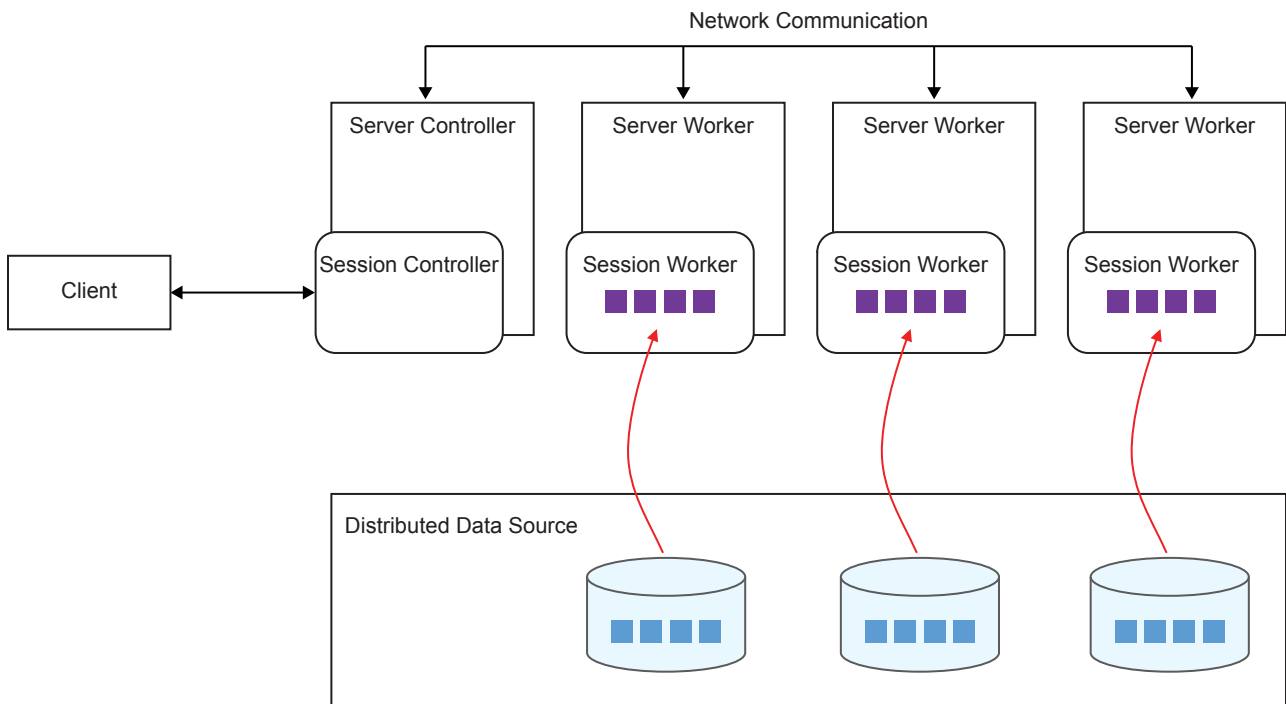


Table A.30 Advantages and Disadvantages of Server-Side Data Loading

Advantages	Disadvantages
<ul style="list-style-type: none"> ■ Large files and small files are loaded into memory as fast as possible. ■ Files can be loaded into memory after a server restart (or a table is dropped) very quickly. 	<ul style="list-style-type: none"> ■ You might not be sure which files are available in the caslib's data source. You can use the LIST FILES statement with the CASUTIL procedure to see which files can be loaded into memory.

Caslibs, Files, and Tables

All access to data with SAS Cloud Analytic Services is through a caslib. At its simplest, a caslib has the following properties:

- A caslib is associated with a data source and includes the connection information for the data source. For example, the data source can be a directory or the host, port, and other connection information for an Oracle database.
- The data in the associated data source is referred to as a file. For path-based caslibs, these files are SASHDAT files, CSV files, SAS data sets, and so on. For server-based caslibs, such as an Oracle database, the term file is still used to create a distinction between data from the caslib's data source, and an in-memory copy of the data.
- SAS Cloud Analytic Services performs analysis on in-memory tables only. In addition to providing an association with a data source, a caslib provides access to in-memory tables that have been loaded into memory.

Throughout product documentation, the following terms are used interchangeably:

- data table
- in-memory table

- table

Caslibs also provide access control to data. For more information, see *SAS Viya Administration: Cloud Analytic Services Authorization*.

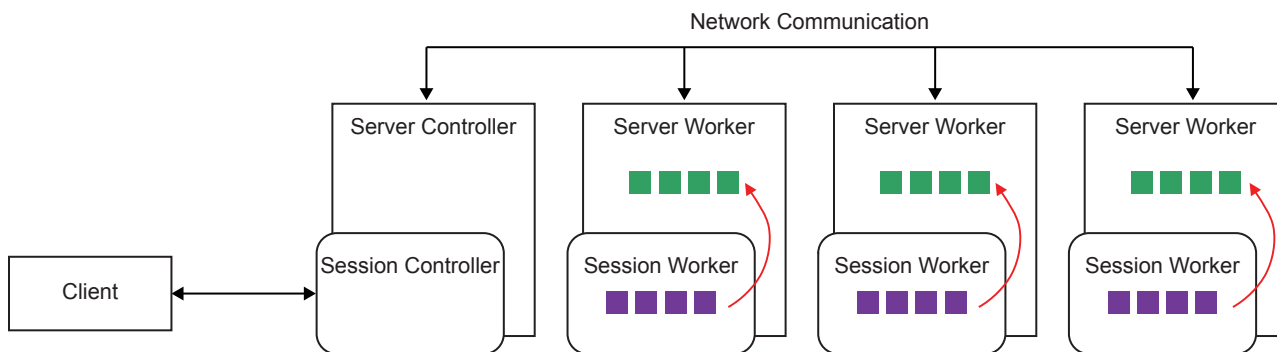
More About Tables

Session and Global Scope

By default, when you load a table into memory, the table has *session scope*. This means that the table is available to that session only. For ad hoc data access and analysis, session-scope tables are preferred because session-scope tables do not require access control checks or any form of locking for concurrent access.

The only disadvantage to a session-scope table is that no other sessions can access the same table. For example, if you want shared access to a single copy of an in-memory table, then a session-scope table does not work. In that case, a global-scope table can provide the shared access.

The following figure depicts how the rows of a session-scope table (in purple) are *promoted* to global scope (in green). After the table is promoted, any sessions that have access to the same caslib can access the single copy of the in-memory table.



Fault Tolerance: Data Redundancy

Fault tolerance applies to distributed servers only.

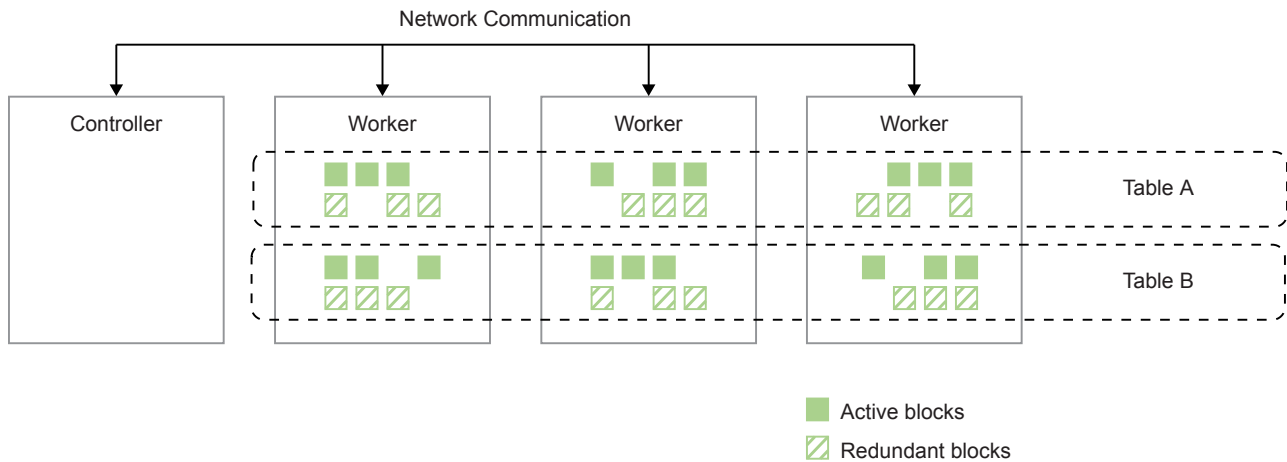
In order to offer fault tolerance, the server performs one of the following when data is loaded into memory:

- If the table was loaded from a SASHDAT file in HDFS, then fault tolerance is provided by the redundant blocks that were created when the file was saved to HDFS. In the event of node failure, a surviving node accesses the data from the redundant block.
- If the table was loaded from a SASHDAT file in a caslib with a source type of DNFS or PATH, then redundant blocks are not loaded preemptively into memory. In the event of a fault, the nodes that remain can access the source file and load copies of the blocks that were used by the failed nodes.
- For all other file and source data types, when you load the file into memory, you can specify the number of redundant blocks to create. In the event of node failure, a surviving node accesses the data from the redundant block.

Note: There is an exception for data that is loaded with a data connector or a data connect accelerator. The SAS Data Connector to Oracle and SAS Data Connector to Hadoop are examples of these products. There is no data redundancy for tables loaded with a data connector or data connect accelerators.

TIP The redundant copies of blocks are stored in the directories associated with the file system directories associated with the CAS_DISK_CACHE environment variable. Increasing the number of copies increases the amount of disk storage that is used. For more information, see [“About the Disk Cache” on page 347](#).

The following figure depicts how the server uses the system of active blocks and redundant blocks to provide fault tolerance. There are three active blocks for Table A on the first worker node. The redundant blocks for those three blocks are distributed between the second and third worker nodes.



Partitioning and Ordering

By default, the order of rows in a table is not predictable. This is true for both single-machine servers and distributed servers.

One way to introduce order is to partition a table by one or more columns and then specify one or more different columns to use for ordering the rows. When you partition a table, all the formatted values for the partitioning columns are kept in a single partition. For a distributed server, the partition is on a single machine. For a single-machine server, all the partitions are on the single machine.

If you use BY-group processing in a DATA step with the same variables, then it is a performance advantage to partition the table when you load the table into memory on the server. Otherwise, the BY groups are formed each time the DATA step is run.

Similarly, if you use a procedure that supports a GROUPBY statement and you specify the partitioning variables, then it is a performance advantage.

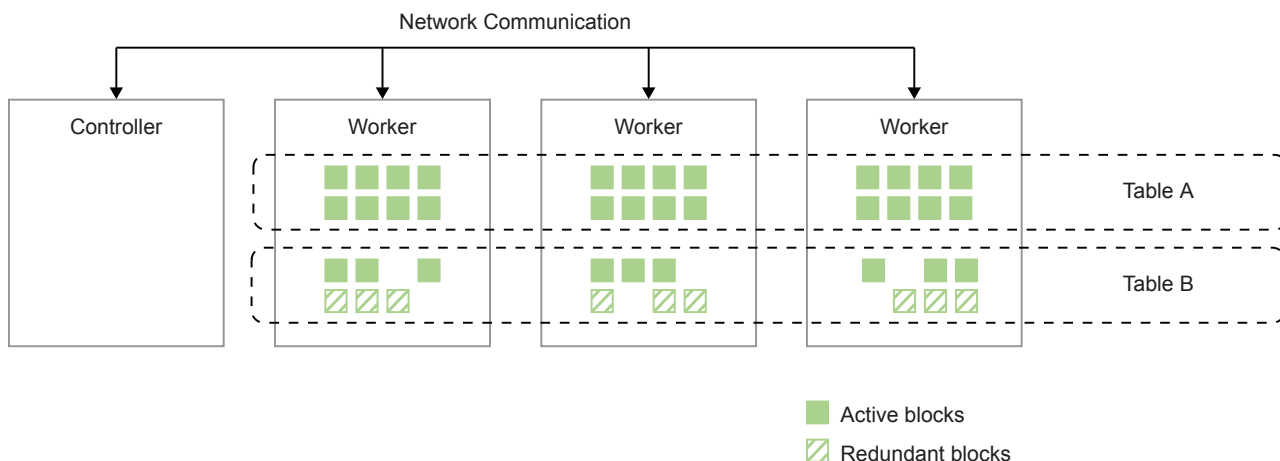
Indexing

Column indexes are introduced with the SAS Viya 3.3 release.

- The server supports indexes on numeric (DOUBLE), CHAR, and VARCHAR columns.
- Indexes are supported on single-machine and distributed servers.
- Indexes can be used with other table features such as compression, encryption, fault tolerance, and session-scope and global-scope tables.
- When you subset rows with a filter and specify at least one column that is indexed, the index is used to improve performance. The server identifies the indexed column with the highest cardinality and excludes data that can never satisfy the filter criteria.
- When you update values in rows or append rows, the indexes are updated. The index is part of the in-memory table and not a separate object or file. There is no need to re-index an in-memory table.
- When you save an indexed in-memory table to a caslib that uses PATH, DNFS, or HDFS, the index is included in the saved SASHDAT file. The next time you load the table from the file, the index is ready for use.

Repeated Tables

The following figure depicts a repeated table, Table A, and a distributed table, Table B. A repeated table has all of the rows in blocks that are identical on all worker nodes of a distributed server. Put another way, the table is loaded into memory on all the machines.



Repeated tables are useful in some operations, such as table joins, where the rows of a dimension table need to be matched against the keys in the fact table. If all the rows of the dimension table are repeated on each worker node, then the join can be completed without exchanging rows between the worker nodes. Repeated tables are not managed for fault tolerance because each node has all the rows.

Rules for Caslib, Table, and Column Names

Caslib Names

- Names can be 1 to 256 characters long. National and multi-byte characters are accepted.
- The following prefixes are reserved.
 - Names cannot begin with CASUSER (in uppercase, lowercase, or mixed case). This is a reserved prefix to prevent confusion with the CASUSER and CASUSERHDFS personal caslibs.
 - Names cannot begin with an underscore character (_). This is a reserved prefix for caslibs that are used for the server.
 - Names cannot begin with #LOCAL_. This is a reserved prefix that is used by the server for automatically named session-scoped caslibs that are used by the server.
 - Names cannot include the slash character (/).

Table Names

- Names can include national characters and have no limitation on length. However, most operating systems have limits on filename lengths if you need to save an in-memory table as a file. Similarly, most databases have limits on table name lengths and can deny saving an in-memory table with a long name as a database table.
- The server does not stipulate limits on lengths. However, the CAS LIBNAME engine, SAS, and most third-party clients have limits. If you intend to access the tables with the CAS LIBNAME engine or SAS, see [Names in the SAS Language](#) in *SAS Language Reference: Concepts*.

- In-memory table names are converted to uppercase. Searches for in-memory table names are case insensitive.
- If the table is loaded from a caslib that enables access to subdirectories, slashes are replaced with periods (.). However, searches for table names do not convert slashes to the period character.
- When you load data from files, the file extensions are stripped by default.
- When saving a table to a path-based caslib that enables access to subdirectories, you can specify a slash in the filename to save the file in a subdirectory.
- UNIX filenames use the character encoding that is based on the server's locale when it is started. A mix of encodings can be an issue when referencing SAS data sets in a shared file system. This is common when data sets are created by SAS in one encoding and referenced by a CAS server that is running a different encoding. For example, there is no way to reference a pathname that was created with the EUC-CN encoding unless all paths from that CAS server are also EUC-CN. The server sets the encoding according to the LANG environment variable at server start-up.
- Temporary in-memory tables that are created by the server use a `_T_` prefix for the name. It is not a reserved prefix, but avoid naming tables with the prefix to prevent confusion between temporary tables and named tables.

Column Names

- Names can include national characters and have no limitation on length. However, most databases have limits on column name lengths and can deny saving an in-memory table with a long name as a database table.
- The server does not stipulate limits on lengths. However, the CAS LIBNAME engine, SAS, and most third-party clients have limits. If you intend to access the tables with the CAS LIBNAME engine or SAS, see [Names in the SAS Language](#) in *SAS Language Reference: Concepts*.
- Searches for column names are case insensitive. A table cannot have one column that is named `colA` and attempt to add a column that is named `ColA`.

Memory

How SAS Cloud Analytic Services Uses Memory

SAS Cloud Analytic Services (CAS) is an in-memory server and it analyzes in-memory tables. The primary concern for memory use is for in-memory tables. The goal of the server is to use memory efficiently and provide the best performance for the amount of physical memory available and the data volume to analyze.

To meet the goal, the server uses file-based memory mapping. For SASHDAT files, because the file is already on disk, the server memory maps the file. For other file types and data sources, the server uses the directories associated with the `CAS_DISK_CACHE` environment variable to store blocks temporarily in files.

Some of the benefits of memory mapping are as follows:

avoid paging to system swap space

System swap space is small compared to the overall disk space for the host. The swap space is used when memory demand is high. The cost to page out data is high because performance is limited to the write speed of disk drives.

With memory mapping, the host can write in-memory blocks to the cache during idle time and avoid poor performance when free memory is extremely low. The read speed for disk drives is high, so the cost to read memory mapped blocks is low.

data that exceeds physical memory capacity

By memory-mapping blocks, the server is able to analyze data that is larger than physical memory capacity. This applies to both single large tables and when the combined size of many tables exceeds memory capacity. Blocks are read from the memory until the physical memory limit is met. Then, because the blocks are memory mapped, some physical memory can be freed (without the performance penalty associated with paging out) and blocks are paged in from the next series of memory-mapped files.

memory efficiency

For global-scope tables and all tables loaded from SASHDAT files, the use of memory mapping enables multiple sessions to share the same physical memory. If many sessions access the same global-scope table or SASHDAT-backed table, only one instance of the data is in physical memory.

Managing and Monitoring Overview

About the Disk Cache

SAS Cloud Analytic Services organizes data from tables in blocks. With the exception of SASHDAT files and specialized cases, a copy of in-memory blocks are temporarily stored in file system directories. When the server is installed, one or more of these directories are specified for the `CAS_DISK_CACHE` environment variable.

The disk cache affects performance in two ways. First, the disk space must be sufficient to store the blocks. For distributed servers, keep in mind that additional copies of blocks are stored to provide data redundancy. Secondly, the number of disks is a factor for speed. The server uses the specified directories in a round-robin fashion. If the directories correspond to different physical disks, then contention is reduced and performance is better.

The best practice is to set the directories during deployment. For tuning advice and sample scenarios, see [Set the CAS Cache Directory](#) in *SAS Viya for Linux: Deployment Guide*.

Tools for Monitoring Memory Use

The stand-alone grid monitor shows memory use for the server process and for each session. The metrics include the virtual memory size, the size blocks in the `CAS_DISK_CACHE`, and the size of blocks that are read from SASHDAT files. You can drill down to view the metrics by host (listed as "rank").

The CAS Server Monitor shows the memory use for the server process, session processes, and the host. You can view per-session resident memory size (physical memory).

The `table.tableDetails` action can list the number of mapped blocks, unmapped blocks (backup blocks), and blocks that are allocated from physical memory only (not backed by a file). This information is available at the table level. The action is used by the `CASUTIL` procedure, so some of the functionality is available with the procedure.

How to Limit Memory Use

Limiting the Address Space Is Not the Best Choice

To enforce memory use limits, an administrator can set an address space `ulimit` for a user or group. However, this is not universally recommended because `ulimit` is a single-process control. Memory is used by the server process and each of the many session process that users start. Because each user can start more than one session, the single-process accounting is unlikely to provide the control that is needed.

Finally, the address space `ulimit` does not distinguish between virtual memory and the memory efficiencies of sharing physical memory. The shared memory size is included in the address space for each session and that does not help manage physical memory.

Linux Cgroup

A Linux cgroup can enforce a physical memory limit for all the session processes that are started on that server. To enable the cgroup memory limit, an administrator starts the server with the `cas.memorysize` configuration file option. The minimum limit is 3 GB. For distributed servers, a cgroup with the specified limit is created on each host.

The cgroup is used to limit the physical memory use and not the address space use. As a result, users can still operate on tables that exceed the cgroup limit because only a portion of the table is paged in to physical memory at any time. The cgroup enables administrators to manage physical memory utilization instead of virtual memory. This is important because physical memory is the scarce resource and the subject of management and tuning.

Because most tables are memory mapped from SASHDAT files or `CAS_DISK_CACHE` (the strategy for short-lived output tables is an exception), the server does not use any system swap space. CAS configures the cgroup to prevent use of swap space to avoid poor performance.

Be aware of the following:

- In addition to memory use for tables, some memory is allocated by an action to operate on a table. The memory limit that you set must accommodate this transient memory use.
- Global-scope tables are always memory mapped from a SASHDAT file or `CAS_DISK_CACHE`. The server process does not maintain the mapped memory that contains the data. Instead, the server passes references to data so that the session processes can map the data. As a result, the operating system often caches the data in physical memory and multiple sessions benefit from accessing the cached memory.
- If the cgroup memory limit and the specialized settings for short-lived tables are used at the same time, the risk of having a session killed is high. The cgroup restricts the amount of physical memory that the server process can use. When all memory has no backing store and the limit is reached, the server terminates one or more sessions. The server reviews the out-of-memory (OOM) score that the operating system assigns for each session. The server terminates the session with the highest OOM score.
- If a session is killed, the server log includes the message "Session 'xxxxx (ID)' terminated due to low memory."

The use of a Linux cgroup to limit memory use can be combined with Hadoop YARN for distributed servers that must share computing resources with other software that is installed on the same hardware. Administrators enable integration with YARN by setting the `cas.useyarn` configuration file option. When this option is enabled, as the server starts, it makes sure that YARN reserves the memory limit that is specified in the `cas.memorysize` option on each host. The server does not accept client connections until YARN can reserve the specified memory size. If YARN cannot reserve the memory, the error message "xxxx greater than max container memory" is generated.

Special Cases

SASHDAT Files

When a server has local data access to a SASHDAT file, the server does not copy blocks to the `CAS_DISK_CACHE` because the server can memory map the file itself.

For a single-machine server, SASHDAT files from a `caslib` with a data source type of `PATH` does not use the `CAS_DISK_CACHE`. For a distributed server, SASHDAT files from a `caslib` with a data source type of `HDFS` or `DNFS` also do not use `CAS_DISK_CACHE`.

In those use cases, the server is already using the memory efficiency and performance strategies that are associated with memory mapping.

Short-lived Output Tables

In some cases, an output table from an action is large and only needed for a very short period of time—until it is used by another action or reporting is complete. In these cases, programmers and SAS applications can set the number of backup copies to zero and set the `maxTableMem` session option to a large size. (The `maxTableMem` session option specifies the amount of physical memory to allocate before the `CAS_DISK_CACHE` is used to store data in files.)

Use this strategy with care:

- An in-memory table with no redundant blocks cannot survive a node failure. There is no fault tolerance in this case.
- Because these settings insist on using physical memory only:
 - Unlike tables that memory map from `CAS_DISK_CACHE`, data for the table might become written to the system swap space when all memory is in use. Use of the system swap space degrades performance severely.
 - The session is at greater risk of being killed by the out-of-memory (OOM) killer that is part of the Linux operating system if Linux cgroups are used.

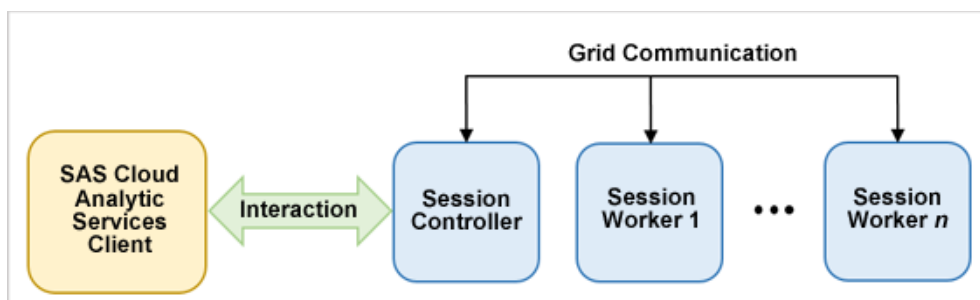
Sessions

About SAS Cloud Analytic Services Sessions

In SAS Cloud Analytic Services, sessions are used to enable clients to communicate with the server to request actions. When you log on to SAS Studio, you must use the `CAS` statement to create a session in order to connect to SAS Cloud Analytic Services. The code snippet **New Session** in SAS Studio provides the SAS code that is needed to create a session.

When you create your session, the server first authenticates your identity. If your identity is successfully authenticated, the server then starts the session controller process. In a distributed system, the server also sends a message to each machine in the cluster to start a session process for your client. The session processes for a distributed system are shown in the following figure.

Figure A.6 Session Processes in a Distributed System



The session controller process does the following:

- manages client connections to the session
- monitors the state of the session
- waits for an interface to force a shutdown of the session

The session worker process in each worker machine waits for requests from the session controller process.

After the session processes are started, your client connects to the session. Each client can establish one connection to the session. Multiple clients can connect to the same session. The session is identified by a session name and a UUID. While connected to your session, you can create resources such as caslibs, SAS library references, tables, and user-defined formats. By default, these resources are visible only to your session. You can make these resources visible to all sessions, if needed. You can use the session identity to manage your sessions and session resources. A session executes requests serially. While a request is processing, subsequent requests are queued until the current request is completed. For each request, the results are returned to the client that made the request.

A client typically terminates the session when it is no longer needed. You can disconnect from your session without terminating it, or you might be disconnected from it unexpectedly due to a loss of network connectivity. In either case, if there are no other client connections to your session, it becomes an orphan session. To conserve system resources, an orphan session is automatically terminated if a client connection is not established within a time-out period (60 seconds by default). The time-out is controlled by the `TIMEOUT=` session option.

By default, SAS Cloud Analytic Services enforces a limit of 5000 concurrent CAS sessions. For more information, see *SAS Viya Administration: SAS Cloud Analytic Services*.

Why Sessions Are Used

Sessions are used to provide the following:

- identification
- fault isolation
- efficiency
- resource tracking

Identification is provided by the use of user credentials. When you connect to SAS Cloud Analytic Services, the server authenticates your user credentials before it creates the session processes. The processes that are created for your session retain your identity and use it to access resources on the server.

Fault isolation is provided for each session through the isolation of its processes from those other client sessions and those of the server itself. If a problem occurs in your session, it does not impact other clients or the server.

Efficiency is achieved through the use of session-scope resources and through concurrent processing, when needed. By default, the resources that you create in your session have session scope. That is, they are visible within your session but not to other client sessions. Other users cannot access and modify your session-scope resources. Concurrent processing provides greater efficiency when processing large amounts of data, especially in distributed systems.

Finally, resource tracking is enabled through the use of resource metrics. Session option `METRICS=` enables the display of information about the system resources that are consumed by a session. By default, metrics information is disabled. When `METRICS=` is set to `True` for your session, system resource metrics are displayed after each request is executed. Metrics include real time, CPU time, memory usage, and so on. Using the metrics information, you can see the system resources that your session is using and make adjustments, if necessary.

Session Properties

Each session has configurable properties that control various session characteristics. A default value is provided for those properties that require a value. You can use session options to manage the properties for your session.

Security

Authentication

Authentication is the process of verifying the identity of a user that is attempting to log on to or access software.

The server is configured to use an authentication provider during the deployment process. There are three interfaces that are authentication points:

SAS Studio	When programmers access SAS Studio from a URL that is similar to <code>https://webserver-host-name/SASStudio/</code> , the programmers are prompted for a user ID and a password. Those credentials are used to authenticate to SAS Cloud Analytic Services.
CAS Server Monitor	The CAS Server Monitor provides a web-based interface for administration. It is accessible within SAS Studio from the More application options menu as well as a URL that is similar to <code>http://webserver-host-name/cas-shared-default-http/</code> . A user ID and a password are required, there is no single-sign on between SAS Studio and CAS Server Monitor.
Batch processing programs	When you need to run SAS programs in batch (as opposed to interactive processing with SAS Studio), credentials for the user ID that runs the program are supplied from an authinfo file.
SAS Home	If available, it enables you to access the visual interfaces. The visual interfaces can include SAS Visual Analytics and SAS Environment Manager. The URL is similar to <code>https://webserver-host-name/SASHome/</code> . A user ID and password are required to authenticate to SAS Logon Manager. Any connection to CAS from a visual interface is authenticated using an OAuth token that is generated by SAS Logon Manager.

See [Authentication Options](#) in *SAS Viya Administration: Authentication*.

Authorization

Authorization is the aspect of security that determines which resources are available to each identity. The primary task is to provide appropriate access to any global caslib that you add. You can also manage access at the table, column, and row levels.

See *SAS Viya Administration: Cloud Analytic Services Authorization*.

Encryption for Data at Rest

The SASHDAT file format is used when SAS Cloud Analytic Services saves tables to disk. This file format supports encryption for data at rest in two ways:

- file-by-file encryption with passwords that are supplied by programmers for encryption and decryption.
- all SASHDAT files in a caslib can be encrypted and decrypted seamlessly by specifying a password when the caslib is added to the server.

For more information, see *Encryption in SAS Viya: Data at Rest*.

Encryption for Data in Motion

When you deploy SAS Viya into your environment, the basic framework for TLS encryption is included by default. In particular, the SAS deployment includes the following to help configure TLS for data in motion:

- On the Apache HTTP Server, the module called `mod_ssl` provides TLS support. This module relies on OpenSSL to provide the cryptography engine. In addition, SAS Viya includes customizations to support SAS internal standards for developing software that protects data-in-motion.
- The Apache HTTP Server (web server) has a localhost certificate and key that allow HTTPS access to SAS Studio, CAS Server Monitor and Visual Analytics (depending on your order).
- Each machine in the deployment has a Mozilla bundle of trustedcerts CA certificates in the `SASSecurityCertificateFramework` that is used by SAS and Java processes if TLS between the CAS Client and controller is turned on.
- The `SASSecurityCertificateFramework` also generates encrypted self-signed certificates for a CAS controller machine in the deployment that can be used to turn on TLS between the CAS Client and controller. These self-signed certificates are part of the CA bundle of trusted certificates (`trustedcerts`).

For more information, see [Encryption in SAS Viya: Data in Motion](#).

Caslibs

What is a caslib?

A caslib is an in-memory space to hold tables, access control lists, and data source information. All data is available to CAS through caslibs and all operations in CAS that use data are performed with a caslib in place. Authorized users can add or manage caslibs with the `CASLIB` statement in SAS Studio.

Caslibs perform the following functions:

- provide access to data from the data source and access to in-memory tables that are copied from the data source. For example, a caslib named `HPS` might be defined as the HDFS path `/hps` and its subdirectories. All files with that path are potential data sources that the server can access.
- provide a space to hold temporary, in-memory tables that can have operations performed on them.
- provide a space to hold connection information to the data source. For example, a caslib that accesses an Oracle database contains connection information such as password, schema information, and data source type.
- provide a common interface into data providers.
- create an association with access controls that define what groups and individual users are authorized to do with the contents of the caslib.

You can also hide caslibs. A hidden caslib is omitted from most lists of caslibs. Tables in a hidden caslib are omitted from most lists of tables. For more information, see [Reduced Visibility: Hidden Caslibs](#).

Personal, Pre-defined, and Manually Added Caslibs

Caslibs can be personal, pre-defined, or manually added. Your level of authorization determines your interaction with each type of caslib. For complete information about caslib authorization, see *SAS Viya Administration: Cloud Analytic Services Authorization*.

personal caslibs

Personal caslibs are an optional feature that must be selected when the server is configured. When you start a session, personal caslibs are always available and have global scope. This enables you to access files and in-memory tables from any session that you start. However, they are personal and only your user ID can access the data. These are an optional feature that must be selected when the server is configured.

Predefined caslibs

Predefined caslibs are automatically created during deployment, managed by an administrator, and have global scope. They are created with the appropriate data access controls and are available to multiple users as defined by the administrator.

manually added caslibs

Only administrators and authorized users can add caslibs with the CASLIB statement. Manually added caslibs are typically added in a program for ad hoc data access that might not be generally available to all programmers that use the server.

Caslibs Scope

One property of a caslib is scope. A caslib can have one of two scopes: session scope or global scope. A session-scope caslib is accessible only from the session that adds it. This enables straightforward server-side data access to a programmer and does not interfere with other sessions. When you add a session caslib, your permission to do so is checked. If the permission is granted, the caslib exists only in the session where the caslib was added. By default, when a session is added with the CASLIB statement or the table.addCaslib action, it is added with session scope. Session-scope caslibs are useful if you do not need to share tables, but you only need to access them.

TIP Caslib names must be unique. If you add a session-scope caslib with the same name as a global-scope caslib, the global-scope caslib is effectively hidden because session-scope is searched first.

Global-scope caslibs can be accessible to any session on the server. Depending on access controls, users can share access to in-memory tables. Personal caslibs and pre-defined caslibs from the permissions file are global caslibs. Names of global caslibs must be unique across all sessions within a server. Global-scope caslibs are useful if you want other users to have access to the table, subject to access controls. Global caslibs are also useful if you want to access the table from a second client session.

You can promote tables into global caslibs. You can promote from a session caslib to a global caslib, but you cannot promote into a session caslib. For information about promoting tables with the CASUTIL procedure, see the documentation PROC CASUTIL. You can also promote a table with the DATA step. For information, see [SAS Cloud Analytic Services: DATA Step Programming](#).

The Active Caslib

The active caslib is used to access data in CAS. If your administrator has enabled personal caslibs, then, when you start a session, the initially active caslib is your personal caslib. When you add a caslib, the newly added caslib becomes the active caslib. If you drop the active caslib, the initially active caslib is set as the active caslib again.

Authorized users can change the active caslib programmatically by adding a caslib with the CASLIB statement or by specifying the CASLIB= session option. To see what caslib is active, you can use the CASLIB _ALL_ LIST statement. For documentation about the CASLIB statement and CASLIB= option, see *SAS Cloud Analytic Services: User's Guide*.

QKB Management

Overview

If your site has licensed the SAS Data Quality offering for SAS Viya, you can use SAS Environment Manager to manage SAS Quality Knowledge Bases (QKBs) on a Cloud Analytic Services (CAS) server. A QKB is a collection of rules and reference data that define data quality operations such as parsing, standardization, and matching. SAS software products can use the QKB to perform data quality operations on a CAS server.

Depending on your privileges in SAS Environment Manager, you can perform the following actions:



- list QKBs that are available on your CAS server
- import a QKB onto your CAS server
- query the contents of a QKB on your CAS server in order to see the supported locales
- remove a QKB from your CAS server

Outside of SAS Environment Manager, you can perform the following actions:

- set a default QKB for your CAS server
- customize your QKB before importing it
- create a QKB archive file

How to Access Your QKBs

To access your QKBs from the SAS Viya home page:

- 1 Click  beside the **SAS® Home** banner.
- 2 Select **Manage Environment** to access SAS Environment Manager.
- 3 Click  to open the Quality Knowledge Bases window.






List QKBs

When you open the **Quality Knowledge Bases** window, a list of QKBs that were previously imported to your CAS servers appears in the table view. If no QKBs were imported, then the table is empty and this message is displayed: `No QKBs were found.` A QKB must be imported to a CAS server before the QKB can be accessed on that server. See [“Import a QKB” on page 356](#) for more information.

Here are the prerequisites for viewing QKBs:

- The CAS server to which the QKB was imported must be running.
- The QKB must be previously imported or in the process of being imported to a CAS server.
- You must have permission to access the CAS server to which the QKB was imported.

Table A.31 Actions Available in the Table View

Task	Action
Filter	Filter the list of QKBs by selecting the column name in the Filter by field. Enter single or multiple keywords in the Filter field to locate or subset QKBs in the list.
View Properties	Select a QKB from the table and click  to view the Properties window. You can also right-click the QKB to access the Properties. See “View QKB Properties” on page 357 .
Import a QKB	Click  to import a QKB. See “Import a QKB” on page 356 .
Delete a QKB	Select a QKB from the table and click  to delete a QKB. You can also right-click the QKB to delete it. See “Delete a QKB” on page 358 .
Refresh	Click  to refresh the list of QKBs.
Display or hide columns	Click  to display or hide columns.

For information about customizing the list of QKBs, see [“Work with Information Displayed in Tables” on page 473](#).

Import a QKB

Import a QKB to a CAS server so that SAS software can perform QKB-supported data quality operations on that server.

The SAS Data Quality offering for SAS Viya includes a QKB in a format that is ready to import to a CAS server. The QKB archive (QARC) file is included in the reference data directory in your deployment on the CAS controller node. For example, on Linux, it is located at `/opt/sas/viya/home/share/refdata/qkb/QKB-product-type/QKB-version`. The filename has the `.qarc` extension.


Here are the prerequisites for importing a QKB:

- Your user ID must have the ability to assume the Superuser role for the CAS server to which the QKB is being imported. You do not need to explicitly assume the Superuser role before invoking the import action. To have the ability to assume the Superuser role, you must be listed in the Superuser role membership for the CAS server. For more information, see [“Manage Properties for a Server” on page 282](#). To add the Superuser

role membership to a user who is using SAS Environment Manager, see [“Add or Remove CAS Role Members \(in SAS Environment Manager\)”](#) on page 479.

- The QKB must be in a QARC format. If the QKB that you want to import is not in a QARC format, you must convert it to a QARC file first. For more information, see [“Create a QKB Archive \(QARC\) File”](#) on page 359.
- The QARC file must be accessible to your local machine. If the QARC file is remote, you can mount the file system where the QARC file resides, or you can copy the QARC file from the remote location to your local machine.
- If your user ID is a member of the CASHostAccountRequired custom group, see [“Configure Access for CASHostAccountRequired Custom Group”](#) on page 360. To check whether your user ID is a member of the CASHostAccountRequired group, see [“Add or Remove Custom Group Members”](#) on page 477.


To import a QKB:

- 1 Click . The Import QKB dialog box appears.
- 2 Click **Browse** to locate the QARC file, and then click **Open**.
- 3 Enter a descriptive name to identify the QKB on the target server in the **Target name** field. This QKB name will be supplied as a parameter in SAS data quality software interfaces. Therefore, it is recommended that you use a name that is meaningful and easy to remember. By default, the target name is the same as the imported file.
Note: A QKB name cannot contain any of the following characters: \ / ; %
- 4 In the **Target server** field, select the CAS server to which you want to import the QKB.
- 5 Click **Import**.
- 6 The Import QKB dialog box closes, and the imported QKB is represented as a row in the table. The status of the import is displayed in the **Install Time** column. If the import is successful, the **Install Time** column displays the date and time of the import. If the import is not successful, the **Install Time** column displays a failed status and you can click the **see details** link to view the error.

After you have imported the QKB, you might want to set that QKB as your default QKB on the CAS server. For those instructions, see [“Set the Default QKB and the Default Locale”](#) on page 358.

View QKB Properties

View the properties of a QKB in SAS Environment Manager by right-clicking the table row for the QKB and

selecting **Properties** from the list of actions. You can also select the table row for the QKB and click  (the Properties icon) in the toolbar. A Properties window appears with the following metadata information:

Name	The name of the QKB.
Server	The name of the server where the QKB resides.
Type	The type of server. The only available value in this release is CAS .
Default	The default QKB for data quality operations that run on the server.
Product	The domain supported by the QKB (for example, Contact Information (CI) or Product Data (PD)).
Version	The version of the QKB.
Install time	The date and time that the QKB was imported to the server.
Locales	The list of locales supported by the QKB. If the default locale has been configured, it is identified with a check mark.


Note: For more information about configuring the default locale, see [“Set the Default QKB and the Default Locale” on page 358](#).

Delete a QKB

You can remove the QKB from the list and from the CAS server by using the delete function.

Here are the prerequisites for deleting a QKB:

- Your user ID must be able to assume the Superuser role for the CAS server where the QKB is located. You do not need to explicitly assume the Superuser role before performing the delete action, but your user ID must be listed in the Superuser role membership for that CAS server. For more information, see [“Manage Properties for a Server” on page 282](#). To add the Superuser role membership to a user, see [“Add or Remove CAS Role Members \(in SAS Environment Manager\)” on page 479](#).
- If your user ID is a member of the CASHostAccountRequired custom group, see [“Configure Access for CASHostAccountRequired Custom Group” on page 360](#). To check whether your user ID is a member of the CASHostAccountRequired group, see [“Add or Remove Custom Group Members” on page 477](#).

To delete a QKB, right-click the name of the QKB and select **Delete** from the list of actions. You can also select the QKB, and click . A dialog box appears for you to confirm the action.

Set the Default QKB and the Default Locale

After you import a QKB to use as your default QKB for data quality operations on a CAS server, you must update the `casconfig_usermods.lua` file and specify the default QKB and the default QKB locale. SAS data quality software uses this configuration to identify which QKB to use when performing data quality operations. In some SAS data quality software interfaces, you can choose to use the default QKB or you can explicitly set a different QKB. In other cases, you must use the default QKB.

To set the default QKB and the default locale on a CAS server:

- 1 Update the CAS configuration on the CAS controller machine by using the `casconfig_usermods.lua` file. In Ansible-based deployments, the `casconfig_usermods.lua` file is located in the `/opt/sas/viya/config/etc/cas/default` directory. However, for single-machine, programming-only environments that have been deployed using yum directly, this file might not exist, and you must create it. In either case, you must edit `casconfig_usermods.lua`.
- 2 Add or locate the `cas.dqSetupLoc` and `cas.dqLocale` variables in the `casconfig_usermods.lua` file. Specify the QKB to use as the default QKB, and the ISO code name for the QKB locale to use as the default QKB locale.
 - For the value of `cas.dqSetupLoc`, specify the name of the QKB exactly as specified when the QKB was imported to the CAS server. Use SAS Environment Manager to view a list of previously imported QKBs. See [“List QKBs” on page 355](#).
 - For the value of `cas.dqLocale`, specify the ISO code name of the locale to use as your default.

To browse the list of locales supported by your QKB in SAS Environment Manager, see the **Locales** property in [“View QKB Properties” on page 357](#). Once you choose a default locale, find the ISO code name of that locale and enter it as the `cas.dqLocale` value. For information about QKB ISO code names, see [QKB Locale ISO Codes](#) on the SAS support website.

Here is an example:

```
cas.dqSetupLoc="QKBCI28"
```

```
cas.dqLocale="ENUSA"
```

- 3 Save and close the `casconfig_usermods.lua` file.
- 4 Restart the CAS controller. For more information, see [“Operate” on page 608](#).

Customize a QKB

You can customize your QKB by adding or modifying its definitions and rules using DataFlux Data Management Studio in SAS 9. After customizing your QKB, you can import it to a CAS server using SAS Environment Manager. See [“Import a QKB” on page 356](#).

Note: If you want to customize your QKB, but do not have DataFlux Data Management Studio, contact SAS Technical Support.

Follow these tasks to customize a QKB:

- 1 It is recommended that you download the Windows version of the QKB because the QKB customization tools are available only on Windows. The QKB installer for Windows creates a QKB with a directory tree, which is a ready-to-use format for the customization tools.
Download a copy of the latest Windows version of the QKB from the [SAS download site](#).
- 2 Use the customization tools to enhance the QKB, as described in the “Managing and Customizing QKBs” section of the [DataFlux Data Management Studio: User’s Guide](#). Make sure that you test the changes thoroughly using the customization tools before using the customized QKB for data quality operations.
- 3 To use your customized QKB for data quality operations in CAS, you must convert the QKB into a QKB Archive (QARC) and import the QARC to a CAS server in SAS Viya. For more information about converting your QKB to a QARC file, see [“Create a QKB Archive \(QARC\) File” on page 359](#). For more information about importing the QARC to a CAS server, see [“Import a QKB” on page 356](#).

If you already have a customized QKB, then you can migrate your changes from your previously customized QKB to the latest QKB. To migrate existing QKB customizations to the latest QKB, refer to the [“QKB Merge Tool”](#) topic in the [DataFlux Data Management Studio: User’s Guide](#).

Create a QKB Archive (QARC) File

To import a QKB to a CAS server in SAS Viya, your QKB must be in a format that is suitable for import. This format is a QKB Archive (QARC). If your QKB is stored in the file system as a directory tree instead of a QARC file, you must first convert it into a QARC file.

The SAS Data Quality offering for SAS Viya provides a command-line tool (QARC tool) that you can use to convert a QKB with a directory tree to a QARC. The QARC tool is an executable file named `qarc` (with no file extension) that is located on the CAS controller at `/opt/sas/viya/home/SASFoundation/utilities/bin/`.

Here are the prerequisites for creating a QKB archive:

- The QKB to be converted must be accessible from the location at which you run the QARC tool.
- The QARC tool must be run on a Linux system.
- If your QKB and the CAS controller are on different Linux systems, then you can do either of the following:

- ❑ Copy the QARC tool from the CAS controller to a temporary location on the file system where your QKB resides.
- ❑ Copy the QKB root directory tree to the CAS controller so that the QARC tool can access it.
- When you copy the QARC tool to another location, ensure that it is not copied to the QKB directory tree.

To convert a QKB with a directory tree to a QARC file:

- 1 Change your directory to the directory where the QARC tool is located. The following example assumes that this command is executed from the CAS controller where the QARC tool is deployed.

```
cd /opt/sas/viya/home/SASFoundation/utilities/bin/
```

- 2 Run this command to create a QARC file from your QKB, supplying the path to the QKB root directory as the first parameter and the desired QARC file output name as the second parameter. The QKB root directory is the directory that contains the file `dfx.meta`.

```
./qarc create path-to-qkb-root-dir-input path-to-filename-with-qarc-extension-output
```

The supplied path to the QARC file output should not be under the QKB directory tree. For example, if your QKB root directory is `/qkb/ci/27`, and your desired path to the QARC file output (with a `.qarc` file extension) is `/qkb_qarc/qkbci27.qarc`, the command would look like this:

```
./qarc create /qkb/ci/27 /qkb_qarc/qkbci27.qarc
```

- 3 This QARC file can be imported to your CAS server. To import a QKB using SAS Environment Manager, see [“Import a QKB” on page 356](#).

For more information about using the QARC tool, use the command:

```
./qarc --help
```

Configure Access for CASHostAccountRequired Custom Group

If your user ID is a member of the `CASHostAccountRequired` custom group, and you want to import or delete QKBs, then your user ID must have Write access to the Reference Data `caslib` file system. Perform the following steps:

- 1 Add your user ID as a member of the group that owns the `referenceData` file system directory: `/opt/sas/viya/config/data/cas/default/referenceData`
- 2 Modify the file system permissions of the `referenceData` directory to allow Write access to the group owner by granting `775` permission access.

For more information about the `CASHostAccountRequired` custom group, see [“The CASHostAccountRequired Custom Group” on page 486](#). To check whether your user ID is a member of the `CASHostAccountRequired` group, see [“Add or Remove Custom Group Members” on page 477](#).

Encryption in SAS Viya

Overview	361
Encryption Coverage	361
Encryption in a SAS Viya Deployment	362
Terminology	363
How To	363
Configure TLS and HTTPS	363
Configure SAS 9.4 Clients to Work with SAS Viya	389
Manage Certificates	390
Secure Credentials in the CAS Server (cas.servicesbaseurl)	400
Manage Tokens and Create JWT Signing Keys	401
Concepts	407
SAS/SECURE	407
Transport Layer Security (TLS)	408
Certificates	410
SSH (Secure Shell)	414
Encrypting ODS Generated PDF Files	415
Encryption Algorithms	416
Reference	418
SAS System Options for Encryption	418
SAS Environment Variables for Encryption	434
CAS TLS Environment Variables	435
Configuration File Options for Data Transfer	443
Examples	447
Create Site-Signed or Third-Party-Signed Certificates in PEM Format	447

Overview

Encryption Coverage

SAS Viya provides encryption in two contexts:

- Data in motion is data that is being transmitted to another location. Data is most vulnerable while in transit. Sensitive data in transit should be encrypted. You can protect all traffic in transit between servers and clients. This document covers encrypting data in motion.

Note: This document does not cover encryption for data in motion for Cloud Foundry. See [SAS Viya for Cloud Foundry: Operations](#) for encryption information relevant to Cloud Foundry.

- Data at rest is data stored in databases, file servers, endpoint devices, and various storage networks. This data can be on-premises, virtual, or in the cloud. This data is usually protected in conventional ways by

firewalls. Numerous layers of defense are needed, and encrypting sensitive data is another layer. See [Encryption in SAS Viya: Data at Rest](#).

Encryption in a SAS Viya Deployment

The SAS Viya deployment process provides a default level of encryption for data in motion. SAS Viya is deployed with Transport Layer Security (TLS) to secure network connections and is fully compliant with SAS security standards.

Data at rest can be secured with additional encryption options; by default, data at rest is presumed to be behind the firewall and is not encrypted.

In a full deployment of SAS Viya, all external communication paths are secured by default. In particular, the SAS Viya deployment provides the following default security and additional ways to increase the level of security.

- The default configuration of a full SAS Viya deployment assumes an Apache HTTP server that can be configured as a reverse proxy server. On the reverse proxy server, the module called `mod_ssl` provides a default level of TLS support. This module relies on OpenSSL to provide the cryptography engine. These settings are reasonably secure, but they are not compliant with SAS security standards. SAS recommends that you replace the default self-signed certificates with your own custom certificates and strengthen the default cryptography.
- SAS Viya uses HashiCorp Vault to generate and sign root and intermediate certificates used to secure communication between various SAS Viya processes. Vault provides a point of contact for SAS Viya services requiring certificates needed to maintain secured communication.
- SAS Viya supports TLS encryption between the data provider (Hadoop, Teradata) and the CAS server, and you can take steps to enable that encryption. If you are using a SAS Data Connect Accelerator, the data that is transferred between the data provider and the CAS server is not encrypted by default.

Security certificates are required to use TLS. SAS Viya provides default certificates at deployment that can be replaced with custom certificates that comply with the security policies at your enterprise.

- The Apache HTTP Server is installed with a localhost certificate and key that allow HTTPS access to SAS Studio, CAS Server Monitor, Visual Analytics, SAS Environment Manager, and SAS Theme Designer (depending on your order). However, SAS recommends that you replace these certificates with your own custom certificates pre-deployment to increase the level of security.
- Certificates used for connecting to CAS are signed by a root CA and intermediate certificate generated by Vault. These certificates are part of the CA bundle of trusted certificates (`trustedcerts`) provided by default. These certificates can be replaced with your own custom certificates post deployment.
- Ansible utilities are provided to easily update certificates and distribute them to the truststores in the deployment.

In a SAS Viya programming-only deployment, the basic framework for security is included by default, but is not enabled by default. In particular, the SAS Viya deployment provides the following default framework to secure data-in-motion.

- On the Apache HTTP Server (reverse proxy server), the module called `mod_ssl` provides TLS support. This module relies on OpenSSL to provide the cryptography engine. In addition, SAS Viya includes customizations to support SAS internal standards for developing software that protects data-in-motion.
- The Apache HTTP Server (reverse proxy server), has a localhost certificate and key that allows HTTPS access to SAS Studio and CAS Server Monitor.
- Each machine in the deployment has a Mozilla bundle of `trustedcerts` CA certificates in the `SASSecurityCertificateFramework` that is used by SAS and Java processes if CAS Client TLS is turned on.
- The `SASSecurityCertificateFramework` also generates encrypted self-signed certificates for a CAS controller machine in the deployment that can be used to turn on CAS Client TLS. These self-signed certificates are part of the CA bundle of trusted certificates (`trustedcerts`).

- Customers can encrypt communication between the CAS controller machine in the deployment and implement TLS communications with CAS Clients. This requires generating and registering additional security certificates for the CAS controller machine.

Note: All discussion of TLS is also applicable to the predecessor protocol, Secure Sockets Layer (SSL).

Terminology

Various security strategies are used to maintain data usability and data confidentiality, as well as to validate the integrity of content. Various encryption, hashing, and encoding algorithms are used by SAS to protect your data in motion and data at rest. SAS highly recommends using TLS for protecting data and credentials (user IDs and passwords) that are exchanged in a networked environment.

encoding

Encoding transforms data into another format using a scheme that is publicly available so that it can easily be reversed. It does not require a key. The only thing required to decode it is the algorithm that was used to encode it. PROC PWENCODE, for example, encodes passwords.

encryption

Encryption is a process of protecting data and credentials. Encryption transforms data into another format in such a way that only specific individuals can reverse the transformation. It uses a key that is kept secret, in conjunction with the plaintext and the algorithm, in order to perform the encryption operation. As such, the ciphertext, algorithm, and key are all required to return to the plaintext. Example encryption algorithms are AES and RSA. TLS is an encryption technology.

hashing

Hashes are commonly used to store passwords to prevent them from being viewed. Hash algorithms are one-way functions. They turn any amount of data into a fixed-length "fingerprint" that cannot be reversed. If the input changes by even a tiny bit, the resulting hash is completely different. When passwords are hashed, only the hash is kept. To verify a password, you hash the password and check to see whether the password matches the stored hash. SHA-256 is a hashing algorithm.

salting

Salt is data used as an additional input to the algorithm that encrypts data. The salt is randomly generated and is used to increase the difficulty of brute-force decryption attacks on the data.

How To

Configure TLS and HTTPS

Secure Apache HTTP Server

Overview

SAS Viya uses an Apache HTTP server to act as a reverse proxy server to secure your environment. At deployment, the Ansible playbook can install Apache httpd with mod_ssl automatically. This option uses default Apache security settings and self-signed certificates. These settings are reasonably secure, but they are not compliant with SAS security standards. SAS recommends that you replace the default self-signed certificates with your own custom certificates and strengthen the default cryptography.

By default, HTTPS access to SASHome is enabled in a SAS Viya full deployment. The URL to access SASHome after installing Apache httpd and installing SAS Viya is <https://reverse-proxy-server/SASHome/>.

Note: In a programming-only deployment, there is no SASHome. SAS Viya end users must use HTTP to connect to SAS Studio or CAS Server Monitor because the Apache HTTP Server does not support HTTPS.

SAS recommends strengthening the default cryptography using the `sas-ssl.conf` file. The `mod_ssl` module relies on OpenSSL to provide strong cryptography for the Apache server using TLS cryptographic protocols. You can read more about `mod_ssl` at [User Manual for mod_ssl](#).

SAS also recommends that you replace the default certificates with your own custom certificates that comply with the security policies at your enterprise. These can be replaced pre-deployment or post-deployment of SAS Viya. SAS recommends replacing the Apache `httpd` certificates before deploying SAS Viya. Whether you replace the certificates pre-deployment or post-deployment, SAS recommends replacing the certificates before giving end users access to SAS Viya.

You can strengthen security on the Apache HTTP Server by performing the tasks in this section. These tasks can be performed at any time after your initial deployment. The task for replacing your certificates pre-deployment is the exception.

Block Port 80

When you block Port 80, the port is blocked internally and externally. Port 443 is now used for external communications. Refer to the [Red Hat Enterprise Linux Reference and Security Guides](#) for information about best practices for securing ports.

See “[Enable Required Ports](#)” in [SAS Viya for Linux: Deployment Guide](#) for more information.

To direct all access to use HTTPS and not HTTP, you can redirect port 80 to port 443 on the Apache HTTP server as follows.

- 1 Edit the `ssl.conf` file located in `/etc/httpd/conf.d/` directory.
- 2 Locate the `<VirtualHost>` code block for TLS. Before that block of code, add the following line of code:

Note: There needs to be a space after the `/` and before `https://`
`<machine_name_where_HTTP_Server_installed>/` in the following code fragment.

```
<VirtualHost *:80>
  ServerName <machine_name_where_HTTP_Server_installed>
  Redirect / https://<machine_name_where_HTTP_Server_installed>/
</VirtualHost>
```

- 3 Restart the Apache HTTP Server by entering the following command:

```
sudo service httpd restart
```

Disable Consul Port (full deployment)

To disable Consul port 8500, set the following in `vars.yml` file:

```
DISABLE_CONSUL_HTTP_PORT: true
```

Note: See “[Modify the vars.yml File](#)” in [SAS Viya for Linux: Deployment Guide](#) for more details.

Replace Self-Signed Certificates with Custom Certificates (Pre-Deployment)

The SAS Viya deployment can install Apache `httpd` with `mod_ssl` and self-signed certificates. These settings are reasonably secure, but they are not compliant with SAS security standards. SAS recommends replacing these self-signed certificates with custom certificates that comply with the security policies at your enterprise.

Note: SAS recommends that you install Apache `httpd` and replace the self-signed certificates before you start the deployment process. When you perform this task before installing SAS Viya, the Ansible playbook used to deploy SAS Viya distributes your custom certificates and adds them to the truststore. This process avoids the brief outage necessary to replace the certificates after SAS Viya has been deployed.

For information about deploying Apache httpd see [“Security Requirements” in SAS Viya for Linux: Deployment Guide](#)

During the deployment, the playbook inspects any existing certificates and the CA chain to determine whether they comply with SAS security requirements.

- If compliant certificates are found (the custom certificates), the certificates are not changed.
- If certificates that do not meet SAS security standards are found, the playbook generates a SAS provided self-signed certificate and configures `mod_ssl` to use it.

If you do not add compliant certificates and instead keep the default security settings and certificates, end-users will see a standard web browser warning message. SAS recommends replacing the default certificates before giving end-users access to SAS Viya. Adding your own certificates post-deployment requires a brief outage. See [“Replace Self-Signed Certificates with Custom Certificates \(Post-Deployment\)” on page 366](#).

The certificate filenames and locations of the certificates are set in the `ssl.conf` file at `/etc/httpd/conf.d/ssl.conf`.

- This is the server identity certificate. `SSLCertificateFile /etc/pki/tls/certs/localhost.crt`

Note: `localhost.crt` is the name of the default Apache certificate.

- RSA private key associated with certificate file `server.crt` `SSLCertificateKeyFile /etc/pki/tls/private/localhost.key`

Note: `localhost.key` is the name of the default Apache key file.

- The file that contains the chain of trust. SAS recommends that this file contain the Root CA and all intermediate certificates. `SSLCertificateChainFile /etc/pki/tls/certs/custom-chain.crt`

1 Install httpd and `mod_ssl` on the desired machines.

```
sudo yum install -y httpd mod_ssl
```

2 Download your server identity certificate files.

- #### 3 Copy your new certificate file to `/etc/pki/tls/certs`. If you have both a root certificate file and a chain file that includes the root certificate and intermediate certificates, you only need to copy the chain file to this location.

Note: The certificate file needs to be a Base-64 PEM encoded file.

4 Copy your new key file to `/etc/pki/tls/private`.

Note: The key file needs to be a Base-64 PEM encoded file.

- #### 5 Change the permissions on your certificate file and your chain file to 644. Change permissions on the key file to 600. Use `chmod` or `sudo` commands to change the permissions.

```
chmod 600 custom.key
chmod 644 custom.crt
chmod 644 custom-chain.crt
```

When you list the files, you see the permissions are Read/Write only for the root account: `-rw-r--r--` for the certificate files and `-rw-----` for the key file.

- #### 6 Update the certificate and key file directives in file `/etc/httpd/conf.d/ssl.conf` to point to your new certificates and key.

```
SSLCertificateFile /etc/pki/tls/certs/custom.crt
SSLCertificateKeyFile /etc/pki/tls/private/custom.key
SSLCertificateChainFile /etc/pki/tls/certs/custom-chain.crt
```

- #### 7 Update the value of `HTTPD_CERT_PATH` in `vars.yml` file to point to the CA root certificate file. If there are also intermediate CA certificates, point to the chain certificate file.

```
HTTPD_CERT_PATH: '/install/sas/sas_viya_playbook/certs/custom-chain.crt'
```

Note: See “[Modify the vars.yml File](#)” in [SAS Viya for Linux: Deployment Guide](#) for more details.

See “[Modify the vars.yml File](#)” in [SAS Viya for Linux: Deployment Guide](#) for details about setting the HTTPD_CERT_PATH variable.

- 8 Run the Ansible playbook to install the SAS Viya deployment. See “[Installation](#)” in [SAS Viya for Linux: Deployment Guide](#) for details.

Replace Self-Signed Certificates with Custom Certificates (Post-Deployment)

The SAS Viya deployment can install Apache httpd with mod_ssl and self-signed certificates. These settings are reasonably secure, but they are not compliant with SAS security standards. SAS recommends replacing these self-signed certificates with custom certificates that comply with the security policies at your enterprise.

Note: SAS recommends that you install Apache httpd and replace the self-signed certificates before you start the deployment process. When you perform this task before installing SAS Viya, the Ansible playbook used to deploy SAS Viya distributes your custom certificates and adds them to the truststore. This process avoids the brief outage necessary to replace the certificates after SAS Viya has been deployed. See “[Replace Self-Signed Certificates with Custom Certificates \(Pre-Deployment\)](#)” on page 364.

During the deployment, the playbook inspects any existing certificates and the CA chain to determine whether they comply with SAS security requirements.

- If compliant certificates are found (custom certificates), the certificates are not changed.
- If no certificates are found or if certificates that do not meet SAS security standards are found, the playbook generates a SAS provided self-signed certificate and configures mod_ssl to use it. These are server identity certificates and can be found at `/etc/pki/tls/certs`.

If you do not add compliant certificates and instead keep the default security settings and self-signed certificates, end-users will see a standard web browser warning message. SAS recommends replacing the self-signed certificates before giving end-users access to SAS Viya. Adding your own certificates post-deployment requires a brief outage.

The certificates and key files that Apache specifies by default are set in the directives in the `ssl.conf` file at `/etc/httpd/conf.d/ssl.conf`. The `SSLCertificateFile` and `SSLCertificateKeyFile` are set by default. If you are replacing the apache default certificates with your own custom site-signed certificates, you will modify the three directives shown below.

- This is your server identity certificate. `SSLCertificateFile /etc/pki/tls/certs/localhost.crt`.
Note: `localhost.crt` is the name of the certificates provided by Apache.
- RSA private key associated with certificate file `SSLCertificateKeyFile /etc/pki/tls/private/localhost.key`
Note: `localhost.key` is the name of the certificates provided by Apache.
- The file that contains the chain of trust. SAS recommends that this file contain the Root CA and all intermediate certificates. `SSLCertificateChainFile /etc/pki/tls/certs/custom-chain.crt`

To replace the default certificate files, add your custom certificates (site-signed or third-party-signed certificates) to the `/etc/pki/tls` directory structure and point to the new server certificate and key files.

- 1 Download your server identity certificate files.
- 2 Copy your new certificate file to `/etc/pki/tls/certs`. If you have both a root certificate file and a chain file that includes the root certificate and intermediate certificates, you only need to copy the chain file to this location.

Note: The certificate file needs to be a Base-64 PEM encoded file.

3 Copy your new key file to `/etc/pki/tls/private`.

Note: The key file needs to be a Base-64 PEM encoded file.

4 Change the permissions on the certificate files to 644. Change the permissions on the key file to 600. Use `chmod` or `sudo` commands to change the permissions.

```
chmod 600 custom.key
chmod 644 custom.crt
chmod 644 custom-chain.crt
```

When you list the files, you see the permissions are Read/Write only for the root account: `-rw-r--r--` for the certificate files and `-rw-----` for the key file.

5 Update the certificate and key file directives in file `/etc/httpd/conf.d/ssl.conf` to point to your new certificates and key.

```
SSLCertificateFile /etc/pki/tls/certs/custom.crt
SSLCertificateKeyFile /etc/pki/tls/private/custom.key
SSLCertificateChainFile /etc/pki/tls/certs/custom-chain.crt
```

6 Restart the `httpd` service.

```
sudo sas-viya-httpd-proxy restart
```

7 Restart `httpd`.

```
sudo service httpd restart
```

8 Update the value of `HTTPD_CERT_PATH` in `vars.yml` file to point to the CA root certificate file. If there are also intermediate CA certificates, point to the chain certificate file.

```
HTTPD_CERT_PATH: '/install/sas/sas_viya_playbook/certs/custom_chain.cer'
```

See [“Modify the vars.yml File” in SAS Viya for Linux: Deployment Guide](#) for details about setting the `HTTPD_CERT_PATH` variable. Also see [“Modify the vars.yml File” in SAS Viya for Linux: Deployment Guide](#) for more details.

9 Distribute the certificate chain file to the CA Certificate directory and rebuild the truststores. You can run the following Ansible play to perform this function. On the Ansible controller machine, locate the utility file in the `sas_viya_playbook/utility` directory.

```
ansible-playbook -i inventory.ini utility/distribute-httpd-certs.yml
```

This play adds your new custom certificate to `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts`. The play distributes copies of the certificate chain file to all machines with a name of `httpd-proxy-inventory_name-ca.crt`. The play then rebuilds the `trustedcerts.pem` and `trustedcerts.jks` files and includes the CA certificates from `custom-chain.cer` in the `trustedcerts.pem` and `trustedcerts.jks` file on every machine in the deployment.

10 Restart all services on all machines.

```
sas-viya-all-services stop
sas-viya-all-services start
```

Improve TLS Security for the Apache HTTP Server

In the SAS Viya deployment, the Apache HTTP Server is configured with the `mod_ssl` security module enabled. The `mod_ssl` module provides strong cryptography for the Apache server using SSL and TLS cryptographic protocols. You can read more about what `mod_ssl` does at [User Manual for mod_ssl](#).

SAS recommends that you update your Apache HTTP Server to not only use your own custom certificates, but to also upgrade the security protocol and ciphers being used by default with the ones contained in the `sas-ssl.conf` file. The TLS protocols and ciphers are recommended by SAS to meet the highest data-in-motion standard for cryptography.

The `mod_ssl` configuration file is called `sas-ssl.conf` and is typically located at `/etc/httpd/conf.d/`. If you do not find the `sas-ssl.conf` file at that location, create your own `sas-ssl.conf`.

Here is an example of the contents of the `sas-ssl.conf` for SAS Viya that shows the ciphers being used.

Note: The following code is shown on more than one line for display purposes only. The `SSLCipherSuite` variable plus the ciphers must be on one line and must not contain line breaks.

```
Header set Strict-Transport-Security "max-age=31536000"
SSLProtocol TLSv1.2
SSLHonorCipherOrder On
# The line containing variable SSLCipherSuite and values
must not include line breaks
SSLCipherSuite
ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:
ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:
ECDHE-ECDSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:
ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA256:
AES256-GCM-SHA384:AES128-GCM-SHA256:AES256-SHA256:
AES128-SHA256
```

To include the `sas-ssl.conf` file, edit the `ssl.conf` file as follows.

- 1 Locate the file that contains the `<VirtualHost>` code for TLS. By default, this is in the `ssl.conf` file. This file is located in the `/etc/httpd/conf.d/` directory.
- 2 Edit the `ssl.conf` file to include file `sas-ssl.conf`. In the `ssl.conf` file, locate the `<VirtualHost_default:443>` block of code. Just above the line that contains `</VirtualHost>`, add the following line of code:

```
Include /etc/httpd/conf.d/sas-ssl.conf
```

- 3 Restart the Apache HTTP Server by entering the following command:

```
sudo service httpd restart
```

Configure CAS TLS to Use Custom Certificates (full deployment)

Note: The following instructions are for adding custom certificates to a SAS Viya full deployment.

By default, in a full deployment of SAS Viya, Hashicorp Vault issues certificates and keys that are used to secure the deployment. These certificates issued by Vault are provided for each CAS machine and are added to the Mozilla bundle of trusted CA certificates by default.

Table A.32 Security Certificates and Keys Provided for CAS in a SAS Viya Full Deployment

Security Artifact	Default Certificate and Key Files	Location	Description
trusted CA certificates	trustedcerts.pem trustedcerts.jks	/opt/sas/viya/config/etc/ SASSecurityCertificateFram ework/cacerts	CA certificates issued by Vault. The trusted list of CA certificates includes the Mozilla Bundle of trusted CA certificates, the Root CA certificates issued by Vault, the Apache httpd certificates, and the chain of trust certificates.

Security Artifact	Default Certificate and Key Files	Location	Description
Certificate File	<code>sas_encrypted.crt</code>	<code>/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs/cas/shared/default/sas_encrypted.crt</code> (full deployment)	Vault issued certificates. This file contains the SAS Viya root CA and the Intermediate CA chain.
Private Key File	<code>sas_encrypted.key</code>	<code>/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/cas/shared/default</code>	Vault issued key file that is encrypted.
Certificate Private Key Passphrase File	<code>customerName.key</code> (optional)	<code>/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/cas/shared/default</code>	It is highly recommended that you protect the <code>sas_encrypted.key</code> file with a passphrase (key). When you create your passphrase and create a file to hold the passphrase, you will need to encrypt this new file also.

You can use your own custom certificates instead of the certificates provided by SAS. Best practices for managing certificates and securing your private keys should be followed.

The following instructions are provided to configure TLS for the CAS Client with your own custom certificates. In a full deployment, the SAS Configuration Server (Consul) handles most configuration tasks.

- 1 On the main Consul machine (the machine listed within the `[consul]` host group in `inventory.ini` file), edit the `sitedefault.yml` file located at `/opt/sas/viya/config/etc/consul.d/default/` and place the certificate strings for your custom CA certificates (in PEM format) in the file.

```
sudo vi /opt/sas/viya/config/etc/consul.d/default/sitedefault.yml
```

Here is an example of what the `sitedefault.yml` file might look like after you add your custom CA bundle to this file. First add the identifier (node) named `cacerts`. Beneath that node, add nodes that identify each of the CA root certificates that are being used. In this example, certificate identifiers were added named `sascaroot`, `sasha2rootca`, and `digicertrootca` certificates. Another node was also added to include the custom CA root chain of certificates (`custom_ca_chain`).

Note: How this file is indented is important.

```

cacerts:
  sascaroot: |
    -----BEGIN CERTIFICATE-----
    certificate string
    -----END CERTIFICATE-----
  sasha2rootca: |
    -----BEGIN CERTIFICATE-----
    certificate string
    -----END CERTIFICATE-----
  digicertrootca: |
    -----BEGIN CERTIFICATE-----
    certificate string
    -----END CERTIFICATE-----
  custom_ca_chain: |
    -----BEGIN CERTIFICATE-----
    certificate string
    -----END CERTIFICATE-----
    -----BEGIN CERTIFICATE-----

```

```
certificate string
-----END CERTIFICATE-----
```

- 2 On the main Consul machine, restart Consul. This act copies the customer CA bundle from the `sitedefault.yml` file to Consul's key-value (KV) store.

```
sudo service sas-viya-consul-default restart
```

- 3 Rebuild the truststores. On an Ansible controller machine, run the `./utility/rebuild-trust-stores.yml` Ansible play. This act incorporates the customer CA bundle of trusted certificates (from Consul configuration) into the various truststore files (`trustedcerts.pem` and `trustedcerts.jks`) on each machine in the Viya deployment.

```
sudo -u qstauto ansible-playbook -i inventory.ini
./utility/rebuild-trust-stores.yml
```

Note: Use the same admin user that you used during the initial SAS Viya deployment. In our example, our user is `qstauto`.

For more information, see [“Deploy the Software” in SAS Viya for Linux: Deployment Guide](#).

To configure TLS between the CAS client and server:

- 1 Log on to the CAS controller machine as a user with root or sudo privileges.
- 2 If you have a CAS session running, cancel it.
- 3 Place your custom certificate in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs/cas/shared/default`. The certificate file provided by SAS Viya is named `sas_encrypted.crt`.

Note: Intermediate certificates need to be added to the server identity certificate in a certificate chain. The file needs to include the server identity certificate first, and then the signing intermediate CA certificates in the order in which they were signed. The root CA does not need to be included in this chain file.

Note: Ensure that your files have file system permissions 644: `-rw-r--r--`. Also, ensure that the file has appropriate file system ownership and permissions for CAS ADMIN user. See [“Set Up the CAS Administrator” in SAS Viya for Linux: Deployment Guide](#).

- 4 Place your private key in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/cas/shared/default`. The certificate file provided by SAS Viya is named `sas_encrypted.key`.

Note: Ensure that your files have file system permissions 644: `-rw-r--r--`. Also, ensure that the file has appropriate file system ownership and permissions for CAS ADMIN user.

- 5 Protect your certificate private key file with a passphrase. In this example, the key file is named `custom.key` and the encrypted key file is `custom_encrypted.key`.

- a Use the following OpenSSL command to password protect the file.

```
openssl rsa -aes128 -in /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/cas/shared/default/custom.key
-out /opt/sas/viya/config/etc/SASSecurityCertificateFramework/
private/cas/shared/default/custom_encrypted.key -passout pass:password
```

- b Ensure that the customer's encrypted private key file has permissions set to 644. Use `chmod` to change the permissions:

```
chmod 644 custom_encrypted.key
```

- 6 Create a customer-supplied certificate private key passphrase file. Use the `echo` command to create the private key passphrase file. In this example, the private key passphrase filename is `customer_encrypted.encrypted.key`.

```
sudo bash -c "echo -n 'password' > custom_encrypted.encrypted.key"
sudo chown qstauto:casadmins custom_encrypted.encrypted.key
sudo chmod 0600 custom_encrypted.encrypted.key
sudo cat custom_encrypted.encrypted.key ;
echo password
```

- 7 You can remove the original custom.key file as you have now secured the custom_encrypted.key (encrypted key file) and custom_encrypted.encrypted.key (passphrase protected encrypted key file) files.
- 8 Configure CAS to use the customer-supplied certificates and key. Edit casconfig_usermods.lua file located at `/opt/sas/viya/config/etc/cas/default` on the CAS Controller. Change the required CAS_CLIENT_SSL environment variables. Specify the names of your custom certificate and the custom encrypted certificate private key file (custom_encrypted.key), and the customer-supplied certificate private key passphrase file (custom_encrypted.encrypted.key).

Note: This file has 0600 permissions: -rw-r--r--

```
env.CAS_CLIENT_SSL_REQUIRED=true
env.CAS_CLIENT_SSL_CERT="/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tls/certs/cas/shared/default/custom.crt"
env.CAS_CLIENT_SSL_KEY="/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/cas/shared/default/custom_encrypted.key"
env.CAS_CLIENT_SSL_KEYPWLOC = '/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/cas/shared/default/custom_encrypted.encrypted.key'
env.CAS_CLIENT_SSL_KEYPW = nil
```

When setting the CAS Client environment variables, consider the following information.

Note: See [“Modify the vars.yml File” in SAS Viya for Linux: Deployment Guide](#) for more details.

- If you are using an intermediate CA certificate, then a certificate chain file needs to be specified for the CAS_CLIENT_SSL_CERT= environment variable. The file needs to include the server identity certificate first, and then the signing intermediate CA certificates in the order in which they were signed. The root CA does not need to be included in this chain file.
 - If you are using your own custom certificate and key, you should copy the changes made to CAS_CLIENT_SSL_CERT= and CAS_CLIENT_SSL_KEY= environment variables to the vars.yml file. This change ensures that your settings are not changed when upgrades are made to the deployment.
 - If you are setting the CAS_CLIENT_SSL_REQUIRED= environment variable to true, you should copy the change made to this environment variable to the vars.yml file. This change ensures that your settings are not changed when upgrades are made to the deployment.
- 9 On the CAS controller, restart the cascontroller service.

```
sudo service sas-viya-cascontroller-default restart
```

- 10 For other client-side connections (Lua, Python) you need the root CA certificate on the client specified on Linux.

- The workspace server exports the trustedcerts.pem file by default.
- On Linux, if the root CA is already in the OpenSSL trusted certificate store, Lua or Python clients should work without having to set the CAS_CLIENT_SSL_CA_LIST= environment variable.
- Otherwise, set the CAS_CLIENT_SSL_CA_LIST= environment variable to point to the location of your certificate chain. Root CA certificates at a minimum are needed in the certificate chain.

```
export CAS_CLIENT_SSL_CA_LIST="/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/cacerts/trustedcerts.pem"
```

- For SAS client-side connections, SAS should automatically find the trustedcerts.pem file that is located in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts` either through the workspace server export statement shown previously or the SSLCALISTLOC= system option that is set during installation.

Configure CAS TLS to Use Custom Certificates (programming-only deployment)

Note: The following instructions are for adding custom certificates to a SAS Viya programming-only deployment.

CAS supports encrypted connections between the server and the clients. Use TLS to secure communications between the server and clients. The certificate used for client server communication needs to be signed by a certificate authority (CA) that is trusted by all potential clients.

The Viya deployment provides certificates and keys at installation that secures the deployment. SAS also adds self-signed certificates created for each CAS machine in the deployment to the Mozilla bundle of trusted certificates.

Table A.33 Security Certificates and Keys Provided for CAS in a SAS Viya Programming-only Deployment

Security Artifact	Deployment File name	Location	Description
trusted CA certificates	trustedcerts.pem trustedcerts.jks	/opt/sas/viya/config/etc/ SASSecurityCertificateFramework/cacerts	Contains the trusted list of CA certificates. The trusted list of CA certificates includes the Mozilla Bundle of trusted CA certificates, the SAS-issued Root CA certificates, the Apache httpd certificates, the chain of trust certificates
Certificate File	sas_encrypted.crt	/opt/sas/viya/config/etc/ SASSecurityCertificateFramework/tls/certs/ sas_encrypted.crt	SAS issued certificates. This file contains the SAS self-signed certificates.
Certificate Private Key File	sas_encrypted.key	/opt/sas/viya/config/etc/ SASSecurityCertificateFramework/private	SAS-issued key file. Contains the private key generated by SAS.
Certificate Private Key Passphrase File	customerName.key (optional)	/opt/sas/viya/config/etc/ SASSecurityCertificateFramework/private	Contains the encrypted passphrase. It is highly recommended that you protect the sas_encrypted.key file with a passphrase (key). When you create your passphrase and create a file to hold the passphrase, you will need to encrypt this new file also.

You can use your own custom certificates instead of the certificates provided by default by SAS. See [“Use Best Practices to Manage Certificates”](#) on page 390.

The following instructions are provided to configure TLS for CAS using your own custom certificates.

Note: If you plan to use the SAS provided self-signed certificates to configure TLS for CAS, see [“Configure CAS TLS to Use SAS Default Certificates \(programming-only deployment\)”](#) on page 375.

You can use your own custom certificates or generate your own custom certificates.

- Generate your own site-signed or third-party-signed certificates. To generate site-signed or third-party-signed certificates in PEM format using OpenSSL, see [“Generate Site-Signed or Third-Party-Signed Certificates in PEM Format”](#) on page 394.

- Generate your own self-signed or root CA certificates. To generate self-signed or root CA certificates, see [“Generate Self-Signed Certificates” on page 397](#).
- If your custom root CA is not already included in the trusted CA bundle of certificates, you can add those certificates to the trustedcerts files. For information, see [“Add Your Certificates to the Trust List or to a Certificate Chain” on page 392](#).

To configure TLS between the CAS client and server:

- 1 Log on to the CAS controller machine as a user with root or sudo privileges.
- 2 If you have a CAS session running, cancel it.
- 3 Place your custom certificates and key in the following locations:

- The server identity certificate chained to any intermediate certificates should be placed in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs`.

Note: Intermediate certificates need to be added to the server identity certificate in a certificate chain. The file needs to include the server identity certificate first, and then the signing intermediate CA certificates in the order in which they were signed. The root CA does not need to be included in this chain file.

The private server key is placed in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private`.

- If your CA certificate is not already included in the Mozilla bundle of CA certificates, append the root certificate to the trustedcerts files in the `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts/` directory.
 - To add the root certificate to the trustedcerts.pem file, just include the root certificate at the end of the trustedcerts.pem file.
 - To add the root certificate to the trustedcerts.jks file, you need to import the file using a keytool command.

See [“Add Certificates to the Trustedcerts File” on page 392](#) for information about adding your certificates to the truststore.

Note: Do not delete the trustedcerts files.

Note: Add your root certificate to the trustedcerts.pem and trustedcerts.jks files on every machine in the deployment.

- 4 Protect your certificate private key file with a passphrase. In this example, the key file is named `custom.key` and the encrypted key file is `custom_encrypted.key`.

- a Use the following OpenSSL command to password protect the file.

```
openssl rsa -aes128 -in /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/cas/shared/default/custom.key
-out /opt/sas/viya/config/etc/SASSecurityCertificateFramework/
private/cas/shared/default/custom_encrypted.key -passout pass:password
```

- b Ensure that the customers encrypted private key file has permissions set to 644. Use `chmod` to change the permissions:

```
chmod 644 custom_encrypted.key
```

- 5 Create a customer-supplied certificate private key passphrase file. Use the `echo` command to create the private key passphrase file. In this example, the private key passphrase filename is `customer_encrypted.encrypted.key`.

```
sudo bash -c "echo -n 'password' > custom_encrypted.encrypted.key"
```

```
sudo chown qstauto:casadmins custom_encrypted.encrypted.key
sudo chmod 0600 custom_encrypted.encrypted.key
sudo cat custom_encrypted.encrypted.key ;
echo password
```

- 6 You can remove the original custom.key file as you have now secured the custom_encrypted.key (encrypted key file) and custom_encrypted.encrypted.key (passphrase protected encrypted key file) files.
- 7 Edit the casconfig_usermods.lua file located at `/opt/sas/viya/config/etc/cas/default` on the CAS Controller. Change the required CAS_CLIENT_SSL environment variables. Specify the names of your custom certificate and the custom encrypted certificate private key file (custom_encrypted.key), and the customer-supplied certificate private key passphrase file (custom_encrypted.encrypted.key).

Note: This file has 0600 permissions: -rw-r--r--

```
env.CAS_CLIENT_SSL_REQUIRED=true
env.CAS_CLIENT_SSL_CERT="/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tls/certs/custom.crt"
env.CAS_CLIENT_SSL_KEY="/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/custom_encrypted.key"
env.CAS_CLIENT_SSL_KEYPWLOC = '/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/cas/shared/default/
custom_encrypted.encrypted.key'
env.CAS_CLIENT_SSL_KEYPW = nil
```

When setting the CAS Client environment variables, consider the following information.

- If you are using an intermediate CA certificate, then a certificate chain file needs to be specified for the CAS_CLIENT_SSL_CERT= environment variable. The file needs to include the server identity certificate first, and then the signing intermediate CA certificates in the order in which they were signed. The root CA does not need to be included in this chain file.
 - If you are using your own custom certificate and key, you should copy the changes made to CAS_CLIENT_SSL_CERT= and CAS_CLIENT_SSL_KEY= environment variables to the vars.yml file. This change ensures that your settings are not changed when upgrades are made to the deployment.
- 8 On the CAS controller, restart the cascontroller service.

```
sudo service sas-viya-cascontroller-default restart
```

- 9 For other client-side connections (Lua, Python) you need the root CA certificate on the client specified on Linux.

- The workspace server exports the trustedcerts.pem file by default.
- On Linux, if the root CA is already in the OpenSSL trusted certificate store, Lua or Python clients should work without having to set the CAS_CLIENT_SSL_CA_LIST= environment variable.
- Otherwise, set the CAS_CLIENT_SSL_CA_LIST= environment variable to point to the location of your certificate chain. Root CA certificates at a minimum are needed in the certificate chain.

```
export CAS_CLIENT_SSL_CA_LIST="/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/cacerts/trustedcerts.pem"
```

- For SAS client-side connections, SAS should automatically find the trustedcerts.pem file that is located in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts` either through the workspace server export statement shown previously or the SSLCALISTLOC= system option that is set during installation.

Configure CAS TLS to Use SAS Default Certificates (programming-only deployment)

Use TLS to secure communications between the CAS server and clients. The certificate used for client and server communication needs to be signed by a certificate authority (CA) that is trusted by all potential clients.

At installation, SAS provides certificates that can be used to secure the deployment. Here are the certificates that are added to the CAS machines in a SAS Viya programming-only deployment..

Table A.34 Security Artifacts Provided at Installation for Programming-only Viya deployment

Security Artifact	Deployment File Name	Location	Description
Certificate truststore	trustedcerts.pem trustedcerts.jks	/opt/sas/viya/config/etc/ SASSecurityCertificateFramework/cacerts/	Contains the trusted list of CA certificates. These include the Mozilla bundle of CA certificates, the SAS self-signed certificate, and the Apache Server certificates?
Certificate File	sas_encrypted.crt	/opt/sas/viya/config/etc/ SASSecurityCertificateFramework/tls/certs/	Contains the certificate generated by SAS. These are self-signed certificates.
Certificate Private Key File	sas_encrypted.key	/opt/sas/viya/config/etc/ SASSecurityCertificateFramework/private	Contains the private key generated by SAS.
Certificate Private Key Passphrase File	your_EncryptedPassphrase.key	/opt/sas/viya/config/etc/ SASSecurityCertificateFramework/private/	Contains the encrypted passphrase. You generate your own passphrase and then encrypt the file that contains it.

To use the certificates shown above, configure TLS between CAS servers and the CAS Client. If you plan to use the SAS certificates that are provided at installation, perform the following tasks:

Note: If you plan to use your own custom certificates to configure CAS Client TLS, see [“Configure CAS TLS to Use Custom Certificates \(programming-only deployment\)”](#) on page 372.

- 1 Log on to the CAS controller machine as a user with root or sudo privileges.
- 2 If you have a CAS session running, cancel it.
- 3 Add the following environment variables to `casconfig_usermods.lua` file located at `/opt/sas/viya/config/etc/default` for the CAS Client on port 5570. Set the `CAS_CLIENT_SSL_REQUIRED` environment variable to `true`. The other environment variables are already set to point to the self-signed certificates and keys that were provided at installation.

```
env.CAS_CLIENT_SSL_REQUIRED = 'true'
env.CAS_CLIENT_SSL_CA_LIST = /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/cacerts/trustedcerts.pem'
env.CAS_CLIENT_SSL_CERT = 'opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tls/certs/sas_encrypted.crt'
env.CAS_CLIENT_SSL_KEY = '/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/sas_encryption.key'
env.CAS_CLIENT_SSL_KEYPWLOC = '/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/encryption.key'
```


Note: By default, SAS self-signed certificates are generated using the fully qualified domain name for the Common Name. Make sure that the CAS hostname in programs submitted to CAS matches the Common Name used in the SAS self-signed certificates.

Note: If you are setting the `CAS_CLIENT_SSL_REQUIRED=` environment variable to `true`, you should copy the change made to this environment variable to the `vars.yml` file. This change ensures that your settings are not changed when upgrades are made to the deployment. See [“Modify the vars.yml File” in SAS Viya for Linux: Deployment Guide](#) for more details.

- Restart the `cascontroller` service.

```
sudo service sas-viya-cascontroller-default restart
```

- For other client-side connections (Lua, Python), you need the root CA certificate on the client specified on Linux.

In SAS Viya, the workspace server exports the `trustedcerts.pem` file by default.

On LAX, if the root CA is already in the OpenSSL trusted certificate store, Lua or Python clients should work without having to set the `CAS_CLIENT_SSL_CA_LIST=` environment variable.

Otherwise, set the `CAS_CLIENT_SSL_CA_LIST=` environment variable to point to the location of your certificate chain. Root CA certificates at a minimum are needed in the certificate chain.

```
export CAS_CLIENT_SSL_CA_LIST="/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts/trustedcerts.pem"
```

Note: For SAS client-side connections, SAS should automatically find the `trustedcerts.pem` file that is located in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts` either through the workspace server export statement shown previously or the `SSLCALISTLOC=` system option that is set during installation.

CAS Server Monitor HTTP and HTTPS Access

Overview

CAS Server Monitor is set up for HTTP and HTTPS during initial deployment. Open a web browser and enter one of the URLs in the address field in the following format:

- In a full deployment where CAS is secured by default, enter `https://host1.sas.com/cas-shared-default-http/` (Encrypted Communications)
- In a programming-only deployment, enter `http://reverse-proxy-server/cas-shared-default-http/` (Insecure Communication)

Note: To access CAS Server Monitor, the password must be set for the CAS user ID or other administrative account. See [“Access CAS Server Monitor” in SAS Viya for Linux: Deployment Guide](#) for additional information. For information about password access set up during deployment.

If you did not add compliant certificates and instead kept the default security settings and certificates, you will see the `Your connection is not private` message. SAS recommends replacing the certificates before giving end-users access to SAS Viya. In a full deployment, dual authentication occurs for logon to CAS Server Monitor and access to CAS from SAS Studio.

Block External Connections to Port 8777

You can also access CAS Server Monitor directly using `http://controller-machine:8777/`.

However, to secure web access to your SAS Viya software, you should block external communications to port 8777. Refer to the Red Hat Enterprise Linux Reference and Security Guides at <https://access.redhat.com/documentation/en/red-hat-enterprise-linux/> for information about best practices for securing ports.

See [“Enable Required Ports” in SAS Viya for Linux: Deployment Guide](#) for more information.

Access to CAS Server Monitor from SAS Studio

If you access CAS Server Monitor from SAS Studio, CAS Server Monitor is accessed using the HTTPS protocol by default. If you receive a “Connection Not Secure” message because of HTTPS access, you need to take one of the following actions:

- Import the Certificate Authority certificates used by the Apache HTTP Server into your browser.
- Change the HTTPS protocol to HTTP by changing the variables in the `casconfig_usermods.lua` file that control the protocol and port used in the CAS Server Monitor link accessed from within SAS Studio. Change the following variables:

```
env.CAS_VIRTUAL_HOST = 'external.mycompany.com'
env.CAS_VIRTUAL_PORT = 443
env.CAS_VIRTUAL_PROTO = 'https'
```

- Make sure the host name for the `CAS_VIRTUAL_HOST` variable is the same as the Common Name used in the server identity certificate that the Apache HTTP Server is using.

Enable TLS Ports at Run time

Port Families


Based on values set in Consul, TLS needs to be able to be enabled or disabled at run time without using Ansible. Using SAS Environment Manager, you can enable or disable network security traffic for TLS using categories (families) of ports. The families of ports are shown in the following table. For more information, see [“Enable Required Ports” in SAS Viya for Linux: Deployment Guide](#).


Family Name	Description	Ports that can be controlled
web	Port family for any network associated with machines running web applications	Apache httpd, all web apps and microservices, SAS Cache Locator (Apache Geode), SAS Message Broker (RabbitMQ), CAS Rest, ESP App, SAS Studio, CAS Server Monitor
databaseTraffic	Port family that needs to control traffic to database servers that might be located on different network segments.	SAS Infrastructure Data Server (PostgreSQL), EP Data Connectors
sasData	Port family that controls traffic transporting data to SAS servers. The SAS Workspace Server and the SAS Object Spawner use this port family to enable and disable AES encryption at on startup.	CAS Client, SAS Compute Server, SAS/CONNECT Server, SAS/CONNECT Spawner, SAS Event Stream Processing (ESP) Server SAS Workspace Server, SAS Object Spawner
serverControl	Port family that controls traffic sent between clustered servers to maintain the cluster.	SAS Launcher Server, SAS Configuration Server (Consul)

Note: Ports for Vault and CAS Internodes are exempt from the family of ports that can have TLS security enabled or disabled.

Enable TLS Using SAS Environment Manager

The SAS Configuration Server (consul) settings are controlled by the configuration service and SAS Environment Manager. To alter the settings in SAS Environment manager, change the sas.security network settings.

1 From the side menu , select **SAS Environment Manager**.


2 In the navigation bar, click .

The Configuration page is an advanced interface. It is available to only SAS Administrators.

3 The default view is **Basic Services**. Select **Definitions** from the drop-down box.

4 In the **Definitions** list, filter on **sas.security**. Select sas.security.

5 If the definition has no properties configured, complete the following:

a In the top right corner of the window, click .

b In the New sas.security Configuration dialog box, select **network.web.enabled**. This is the property for the port family that you want to update to turn off web-enabled TLS.

false Disables TLS for this property.

true Enables TLS.

For a description of the properties, see [“Configuration Properties: Reference \(Applications\)”](#) on page 242.

c Click **Save**.

Note: The system will take a few minutes to recognize the new key before starting to use the new key.

6 Restart all services on all machines

```

sas-viya-all-services stop
sas-viya-all-services start

```

Programmatically Enable/Disable TLS on Port Families

Using the SAS Environment Manager to disable or enable TLS on port families is the preferred method. However, you can enable or disable the TLS ports programmatically by using the following commands. To alter the settings in SAS Environment manager, change the sas.security network settings.

Log on to the SAS Configuration Server as a user with root or sudo privileges.

```

opt/sas/viya/home/bin/sas-bootstrap-config
--token-file /opt/sas/viya/config/etc/SASSecurityCertificateFramework/
tokens/consul/default/client.token kv write --force --site-default
config/application/sas.security/network.databaseTraffic.enabled false
/opt/sas/viya/home/bin/sas-bootstrap-config
--token-file /opt/sas/viya/config/etc/SASSecurityCertificateFramework/
tokens/consul/default/client.token kv write --force --site-default
config/application/sas.security/network.sasData.enabled false
/opt/sas/viya/home/bin/sas-bootstrap-config --token-file
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/
tokens/consul/default/client.token kv write --force --site-default
config/application/sas.security/network.serverControl.enabled false
/opt/sas/viya/home/bin/sas-bootstrap-config --token-file

```

```
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/
tokens/consul/default/client.token kv write --force --site-default
config/application/sas.security/network.web.enabled false
```

Enable/Disable TLS on Port Families Using Ansible

Prior to deployment, add the following to the playbook `sitedefault.yml` file:

```
config:
  application:
    sas.security:
      network.web.enabled: false
      network.sasData.enabled: false
      network.databaseTraffic.enabled: false
      network.serverControl.enabled: false
```

Individual servers and spawners can be disabled by changing their instance key. For example, a connect spawner can be changed using the following code:

```
config/connect-spawner-${SASDEPLOYID}/sas.security/network.sasData.enabled
```

Each application must check configuration service entries that record the default settings for TLS. Customers can use SAS Environment Manager to override the default behavior by altering the settings shown above. The new settings are picked up the next time the application starts. Each application/service (including third-party applications) should use best practices for secure configuration.

Configure SAS Viya to Connect to LDAPS Provider

Lightweight Directory Access Protocol (LDAP) connections can be established in a TLS session so that all data that is sent between the LDAP client and LDAP server is encrypted. LDAP over SSL/TLS is known as LDAPS.

To connect to an LDAPS provider, SAS Viya needs access to the CA certificate used by the LDAPS provider. To configure TLS between SAS Viya and the LDAPS provider, use the following instructions to add the CA certificates to the `trustedcerts` files on every machine in the deployment (as a best practice). See [“Use Best Practices to Manage Certificates” on page 390](#).

Note: Only LDAP-based identity providers are supported. These instructions assume that you have basic familiarity with LDAP administration.



- 1 Log on to your machine as a user with root, SAS Admin, or sudo privileges.
- 2 If your LDAPS provider’s CA certificate is not already included in the Mozilla bundle of trusted CA certificates, append the root certificate to the `trustedcerts` files in the `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts/` directory.
 - To add the root certificate to the `trustedcerts.pem` file, just include the root certificate at the end of the `trustedcerts.pem` file.
 - To add the root certificate to the `trustedcerts.jks` file, import the file using the `keytool` command.

See [“Add Your Certificates to the Trust List or to a Certificate Chain” on page 392](#) for information about adding your certificates to the `trustedcerts` files.

Note: Do not delete the `trustedcerts` files.

Note: Add your root certificate to the `trustedcerts.pem` and `trustedcerts.jks` files on every machine in the deployment.

- 3 Use the SAS Environment Manager to set the configuration property `sas.identities.providers.ldap.connection`. Specify an LDAPS port number (by default LDAPS is 636) and specify `LDAPS` in the `url` field. You can also use the port value 3269 (Global Catalog) for LDAPS.

- a If the Configuration page of SAS Environment Manager is not already displayed, select **Resources** ⇨ **Configuration** from the side menu .
- b Select **Basic Services** from the list, and then select the **Identities service** from the list of services.
- c In the **sas.identities.providers.ldap.connection** section, click . In the Edit `sas.identities.providers.ldap.connection` Configuration window, do the following:
 - i Update values for the **port** field, adding an LDAPS port value. Update the **url** field to specify *LDAPS*. For the remaining fields, review the default values and make changes as necessary. The default values are appropriate for most sites.
 - ii Click **Save**.

For additional configuration instructions, see [“Configure Security” in SAS Viya for Linux: Deployment Guide](#). For details about the `sas.identities.providers.ldap.connection` property, see [“Configuration Properties: Reference \(Services\)” on page 223](#).

- 4 If needed, restart the SAS Logon Manager service by running the following command:

```
sudo service sas-viya-saslogon-default restart
```

Note: It might take several minutes to restart SAS Logon Manager.

If needed, restart the Identities service.

```
sudo service sas-viya-identities-default restart
```

Replace Certificates for LDAPS

By default, the Apache HTTP Server has been configured to serve as a reverse proxy to connect to the SAS Viya Web Application. SAS Viya installation provides certificates and configures TLS options in the Apache HTTP server configuration. However, SAS recommends replacing the default certificates with custom certificates. After the certificates have been replaced in the Apache HTTP server, the truststore can be rebuilt using the following instructions.

Note: See [“Secure Apache HTTP Server” on page 363](#).

- 1 Locate the certificates that you would like to remove from the `sitedefault.yml` file, located at `/opt/sas/viya/config/etc/consul.d/default/sitedefault.yml`. For our example, we are removing `sascaroot` and `sassha2rootca` certificates from `sitedefault.yml`.
- 2 Remove certificates from Consul using `sas-bootstrap-config` commands as follows:

Note: The following code is shown on more than one line for display purposes only. This command may need to be on one line and should not contain line breaks.

```
/opt/sas/viya/home/bin/sas-bootstrap-config --token-file
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/
tokens/consul/default/client.token kv delete cacerts/sascaroot
```

```
/opt/sas/viya/home/bin/sas-bootstrap-config --token-file
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens
/consul/default/client.token kv delete cacerts/sassha2rootca
```

- 3 Ensure that the certificates have been removed.

```
/opt/sas/viya/home/bin/sas-bootstrap-config --consul
https://your-configuration-server:8501
--token 75bef370-cc93-44bb-a290-82833f6c4ddf kv read
--recurse cacerts
```

4 Stop the microservices.

```
sas-viya-all-services stop
```

5 Using Ansible, run the utility play rebuild-trust-stores.yml to rebuild the truststores.

```
ansible-playbook -i inventory.ini utility/rebuild-trust-stores.yml
```

6 Restart all the services so that they now reference the new truststores.

```
sudo sas-viya-all-services start
```

Restart the identities service.

```
sudo service sas-viya-identities-default restart
```

7 Sign in to LDAPS.**SAS Studio HTTP and HTTPS Access**

SAS Studio is set up for HTTP and HTTPS during initial deployment. In SAS Viya, SAS Studio is configured to work with the Apache HTTP server.

To access SAS Studio in a full deployment, open a web browser and enter the URL in the address field: `https://reverse-proxy-server/SASStudio/`. By default, SAS Studio is secured.

In a programming-only deployment, SAS Viya end users must use HTTP to connect to SAS Studio or CAS Server Monitor because the Apache HTTP Server does not support HTTPS. To access SAS Studio in a programming-only deployment, open a web browser and enter the URL in the address field: `http://reverse-proxy-server/SASStudio/` (Insecure Communication)

Log on using the credentials for your operating system account.

HTTPS Access to SAS Message Broker

Note: This section is applicable only if you have a full deployment. If you have a programming-only deployment, skip this section.

By default, SAS provides self-signed certificates and keys to secure the deployment. The URLs available to access SAS Message Broker for HTTP and HTTPS post installation are as follows:

- `https://RabbitMQ-IP-address:15672/#/` (Encrypted Communications)
- `http://RabbitMQ-IP-address:15672/#/` (Insecure Communication)

If after clicking on the HTTPS link shown above to access the Message Broker, you receive a certificate error, you might need to import the SAS Viya trusted CA certificate into the browser trust store. One way to perform this task follows.

- 1 From the Internet Explorer browser, enter `https://RabbitMQ-IP-address:15672/#/`
- 2 Import your SAS Viya intermediate CA.
 - a In the RabbitMQ login window, click "Certificate error".
 - b In the **Untrusted Certificate** dialog, select **View Certificates**.
 - c In the Certificate window, click the **Install Certificate** button.
 - d From the Certificate Import Wizard dialog, select Store Location **Current User**. Select **Next**.
 - e For Certificate Store, I select **Place all certificates in the following store**. Browse and select **Trusted Root Certification Authorities**. Click **OK**.
 - f Click **Next**. Thenf select **Finish**. You should receive a message that your "Import was successful".

Sign On to a SAS/CONNECT Spawner Using TLS

Use SAS/CONNECT as a bridge to access data across environments. You can use TLS to secure that bridge when you sign on to the SAS/CONNECT spawner from the SAS/CONNECT client. The sign-on command starts a SAS/CONNECT server.

In a full deployment of SAS Viya, TLS is on by default for SAS/CONNECT. No additional configuration is required.

In a SAS Viya programming-only deployment, you need to configure SAS/CONNECT by performing the following steps.

You can use the certificates provided by SAS, add custom certificates, or generate your own certificates to use TLS to secure SAS/CONNECT.

- If you are using a certificate whose root CA is not already in the Mozilla Trusted CA Certificate bundle, you need to add the root CA certificate to the Mozilla bundle by editing the trustedcerts.pem file. See [“Add Your Certificates to the Trust List or to a Certificate Chain” on page 392](#).
- To create site-signed or third-party-signed certificates, see [“Generate Site-Signed or Third-Party-Signed Certificates in PEM Format” on page 394](#).
- To create self-signed certificates, see [“Generate Self-Signed Certificates” on page 397](#).

Note: If you are using custom certificates or generating your own certificates, use [Best Practices on page 390](#) for securing your certificates and keys.

To configure SAS/CONNECT, perform the following steps:

- 1 Sign in with administrator privileges to the machine containing the SAS/CONNECT spawner.
- 2 In the connect_usermods.sh file located in `/opt/sas/viya/config/etc/connect/default`, set up TLS by adding the SSL encryption options. Edit the connect_usermods.sh file, and add the following encryption options to the USERMODS= line to encrypt the connection for the SAS/CONNECT spawner. Note that this file needs to have global Read permissions: `-rw-r--r--`

In the following code example, the names of the certificate file and the private key file are just example names. These would be the names of the files that you placed in the `/opt/sas/viya/config` directories.

Note: The options are enclosed in double quotation marks.

```
USERMODS="-netencrypt          /* a */
-netencryptalgorithm ssl      /* b */
-sslcertloc /opt/sas/viya/config/etc/SASSecurityCertificateFramework/
tls/certs/server.crt         /* c */
-sslpvtkeyloc /config/etc/SASSecurityCertificateFramework/private/
private.key                  /* d */
-sslpvtkeypass 'password'"   /* e */"
```

- a The NETENCRYPT option specifies that encryption is required.
- b The NETENCRYPTALGORITHM= option specifies that the spawner is started using TLS.
- c The SSLCERTLOC= option specifies the location of a file that contains a digital certificate for the machine's public key. This is used by the server to send to clients for authentication.

Note: If the certificate is not self-signed, the file specified by the SSLCERTLOC= option needs to be a certificate chain file that starts with the server identity certificate and includes the signing intermediate CA certificates. The root CA certificate does not need to be included in the certificate chain.

- d The SSLPVTKEYLOC= option specifies the location of the file that contains the private key that corresponds to the digital certificate that was specified by the SSLCERTLOC= option.

- e The SSLPVTKEYPASS= option specifies the password that TLS requires to decrypt the private key. The private key is stored in the file that was specified by the SSLPVTKEYLOC= option.

Note: SAS first looks for CA certificates in a file named `trustedcerts.pem`, located in the `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts` directory. Therefore, the `SSLCALISTLOC=` system option is not required if you are storing your trusted certificates in the default location. However, if you choose not to use the default location to store certificates, you need to specify the `SSLCALISTLOC=` option with a location for the certificates for the SAS/CONNECT client and spawner. For each of the preceding examples, the default location is used.

- 3 Start the SAS/CONNECT spawner.

```
sudo service sas-viya-connect-default start
```

- 4 The SAS/CONNECT spawner runs the `connectserver.sh` script, which runs the `connectserver_usermods.sh` script. The `connectserver_usermods.sh` script is located in `/opt/sas/viya/config/etc/connectserver/default`. Edit the `connectserver_usermods.sh` file, and add the following encryption options to the `USERMODS_OPTIONS=` line. Note that this file needs to have global Read permissions: `-rw-r--r--`

Note: The options are enclosed in double quotation marks.

```
USERMODS_OPTIONS="-sslcertloc /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tls/certs/Server.crt
-sslpvtkeyloc /config/etc/SASSecurityCertificateFramework/
private/private.key
-sslpvtkeypass 'password'"
```

Note: The certificates specified above are your server certificates.

- 5 After a spawner is started on a SAS/CONNECT server, a SAS/CONNECT client can connect to it. The following example shows how to connect a client to a spawner that is running on a SAS/CONNECT server:

```
options netencryptalgorithm=SSL;
%let myserver=<myHost.myDomain.com> <port>;
SIGNON myserver user=sasdemo passwd="password";
```

If the spawner requires TLS encryption (`NETENCRYPTALGORITHM=SSL`), the SAS/CONNECT client needs to locate the root CA certificate to validate the spawner's certificate. For a Linux client, SAS first looks for the root CA certificate in the `trustedcerts.pem` file in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/default`. Otherwise, you need to specify the location of the root CA certificate by using system options `SSLCALISTLOC=` or `SSLCACERTDIR=`.

For a Windows SAS/CONNECT client, import the trusted root CA certificate into the Windows trusted root certificate store.

- 6 If you need a SAS 9.4 client to work with SAS Viya, see [“Configure SAS 9.4 Clients to Work with SAS Viya” on page 389](#).

See Also

- [“SAS System Options for Encryption” on page 418](#)
- [SAS Viya Overview on page 13](#)

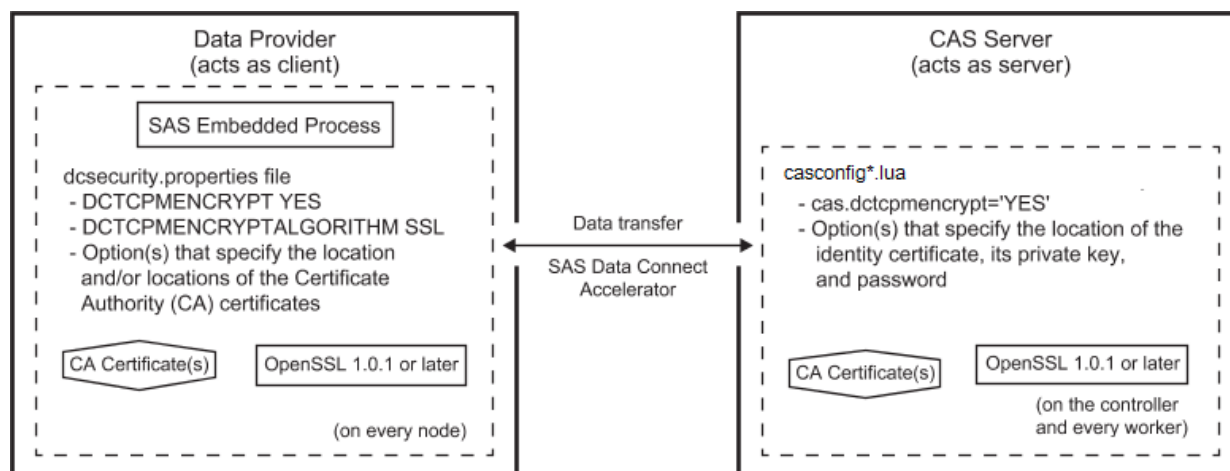
Encrypt Data Transfer When Using the SAS Data Connect Accelerator

If you are using a SAS Data Connect Accelerator, the data that is transferred between the data provider and the CAS server is not encrypted by default. However, SAS Viya does support TLS encryption between the data provider and the CAS server, and you can take steps to enable that encryption. It should be noted that performance can be affected when TLS encryption is enabled and large amounts of data are being transferred.

Overview of SAS Data Connect Accelerator Encryption

When data is transferred between a data provider and CAS, the data provider acts as the client and the CAS server acts as a server.

Figure A.1 Data Connect Accelerator Encryption and the CAS Server



When the SAS Embedded Process is deployed on the data provider, a `dcsecurity.properties` file and a `certs` directory are created in the `SAS-Embedded-Process-home/security` directory. The `certs` directory will hold the TLS security certificates. The `dcsecurity.properties` file must be updated to enable data connector encryption.

When Viya 3.3 is deployed, TLS is enabled and configured on the CAS server (server side). The deployment process provides a default level of encryption for data in motion. Options are set in the `vars.yml` file and defined in the `casconfig_deployment.lua` file to enable data connector encryption and to provide the location of the TLS private key and password.

The prerequisites and process for enabling TLS encryption on the data provider is different for each data provider.

Note: A TLS private key and certificate are required for each CAS host.

Prerequisites When Enabling Encryption for the SAS Data Connect Accelerator for Teradata (on SAS Viya)

Here are the prerequisites for enabling encryption for the SAS Data Connect Accelerator for Teradata (on SAS Viya).

- Upgrade the OpenSSL package on all Teradata nodes to 1.0.1g or later to support TLS.

The 64-bit OpenSSL library package that is most likely being used at your site is `libopenssl10_9_8-0.9.8j-0.50.1`. The required version is `libopenssl11_0_0-1.0.1g-0.37.1` or later. This package update is available on the Teradata patch server. Contact Teradata Customer Services to get this package updated. If you plan to use TLS now or in the future, it is best to upgrade the OpenSSL package before you install the SAS In-Database Technologies for Teradata (on SAS Viya).

Note: The old version, `openssl10_`, and new version, `openssl11_0_0` (or later), can coexist.

- Install SAS/ACCESS Interface to Teradata (on SAS Viya) and SAS In-Database Technologies for Teradata (on SAS Viya).

These offerings include the SAS Embedded Process, the SAS Data Connect Accelerator for Teradata (on SAS Viya), and the SAS Embedded Process support functions. For more information, see [SAS Viya for Linux: Deployment Guide](#).

- Obtain TLS identity certificates (site-signed, third-party-signed, or self-signed) from the CAS controller machine. These certificates are located in the `trustedcerts.pem` file. Corresponding certificate authority (CA) certificates must be installed on the Teradata nodes. If you use externally signed identity certificates in the CAS server, the Mozilla bundle of CA certificates that are provided by SAS can be deployed on the Teradata nodes.

For more information about the location of the `trustedcerts.pem` file, see [“\(Optional\) Deploy TLS Certificates” in SAS Viya for Linux: Deployment Guide](#).

For more information about configuring CAS, see [“Configure CAS TLS to Use Custom Certificates \(programming-only deployment\)” on page 372](#) and [“Configure CAS TLS to Use SAS Default Certificates \(programming-only deployment\)” on page 375](#).

Certificates, keys, and passwords produced for authenticating to the SAS Embedded Process for Teradata might coincide with those produced for other clients of the CAS server. However, they do not need to coincide. For information about generating certificates, see the appropriate topic in [“Manage Certificates” on page 390](#).

- When you installed the SAS Embedded Process, the following file and directory were created:

```
/opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security/dcsecurity.properties
/opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security/certs
```

- All directories and files should have the `owner:group = tdatuser:tdatudf` setting.
- The `/opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security/` directory should have `drwxr-xr-x` permissions.
- The `/opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security/certs` directory should have `drwxr-xr-x` permissions.
- The `dcsecurity.properties` file should have `-rwxr-xr-x` permissions.

Enable Encryption for the SAS Data Connect Accelerator for Teradata (on SAS Viya)

Follow these steps to encrypt data transfer between Teradata and the CAS server using the SAS Data Connect Accelerator for Teradata (on SAS Viya).

- 1 On Teradata, modify the `dcsecurity.properties` file to enable SAS Data Connect Accelerator encryption.
 - a Navigate to the `/opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security/` directory.
 - b Change the `DCTCPMENCRYPT` option in the `dcsecurity.properties` file as follows.

```
-DCTCPMENCRYPT YES
```

CAUTION! The `DCTCPMENCRYPT` option is set on both the CAS server and on the data provider. How the option is set on both sides determines whether the data being transferred is encrypted or not. For more information, see [“DCTCPMENCRYPT Option Setting Interaction” on page 388](#).

- c Add either the `DCSSLACERTDIR` or `DCSSLCALISTLOC` option to the `dcsecurity.properties` file to specify either the location of the trusted certificate authorities or the public certificate(s) for trusted certificate authorities.

Here is an example.

```
-DCSSLCALISTLOC /opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security/certs/certs-filename.pem
```

For more information about the options, see [“dcsecurity.properties File Options for Data Transfer with the SAS Data Connect Accelerator” on page 445](#).

- 2 Copy the necessary TLS CA certificates to the `/opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security/certs` directory.

- If your CA certificates already exists on the Teradata server, copy the CA certificates to this directory.
- If your CA certificates exist on the CAS server, using a method of your choice, copy the CA certificates to this directory on the Teradata server. Here is an example.

```
scp CASCA1.pem tdatuser@teramach1:/opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security/certs
```

For more information about the location of the CA certificates on the CAS server, see [“\(Optional\) Deploy TLS Certificates” in SAS Viya for Linux: Deployment Guide](#).

Note:

- The CA certificates on the Teradata server must authorize the identity certificates that are specified on the CAS server.
 - All Teradata files and directories should have the `owner:group = tdatuser:tdatudf` setting.
 - The `/opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security/` directory should have `drwxr-xr-x` permissions.
 - The `/opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security/certs` directory should have `drwxr-xr-x` permissions.
 - The `dcsecurity.properties` file should have `-rwxr-xr-x` permissions.
- 3** Copy the contents of the `/opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security/` directory to all nodes on the Teradata cluster.

- a** Navigate to the `/opt/SAS` directory.

```
cd /opt/SAS/
```

- b** Create a compressed archive file.

```
tar cvof /tmp/sasep_security.tar/opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security
```

- c** Do a parallel file transfer to push the files to all nodes.

```
pcl -send /tmp/sasep_security.tar /tmp
```

- d** Use the parallel shell command to extract the TAR file.

```
psh tar xvof /tmp/sasep_security.tar
```

- e** Create a backup copy of the TAR file. Keep a backup copy.

```
cp /tmp/sasep_security.tar /root/sasep_security.tar
```

- f** Remove the TAR file from the nodes.

```
psh rm /tmp/sasep_security.tar
```

- 4** Restart the SAS Embedded Process.

- a** Disable the SAS Embedded Process to stop new queries from being started.

```
CALL SQLJ.SERVERCONTROL ('SAS', 'disable', :A);
```

- b** Query the status of the SAS Embedded Process until the status returns this message: Hybrid Server is disabled with no UDFs running.

```
CALL SQLJ.SERVERCONTROL ('SAS', 'status', :A);
```

- c** Shutdown the SAS Embedded Process.

```
CALL SQLJ.SERVERCONTROL ('SAS', 'shutdown', :A);
```

- d** Enable the SAS Embedded Process.

```
CALL SQLJ.SERVERCONTROL ('SAS', 'enable', :A);
```

- Test the SAS Embedded Process. The SAS Embedded Process will start when the next SAS query that uses the SAS Embedded Process is sent to the database.

For more information about stopping and starting the SAS Embedded Process for Teradata, see [Controlling the SAS Embedded Process](#).

Prerequisites When Enabling Encryption for the SAS Data Connect Accelerator for Hadoop (on SAS Viya)

Here are the prerequisites for enabling encryption for the SAS Data Connect Accelerator for Hadoop (on SAS Viya).

- Upgrade the OpenSSL package on all Hadoop nodes to 1.0.1g or later to support TLS.
- Install SAS/ACCESS Interface to Hadoop (on Viya) and SAS In-Database Technologies for Hadoop (on SAS Viya).

These offerings include the SAS Embedded Process and the SAS Data Connect Accelerator for Hadoop (on SAS Viya). For more information, see *SAS Viya for Linux: Deployment Guide*.

- Obtain TLS identity certificates (site-signed, third-party-signed, or self-signed) from the CAS controller machine. These certificates are located in the `trustedcerts.pem` file. Corresponding certificate authority (CA) certificates must be installed on the Hadoop nodes. If you use externally signed identity certificates in the CAS server, the Mozilla bundle of CA certificates that are provided by SAS can be deployed on the Hadoop nodes.

For more information about the location of the `trustedcerts.pem` file, see “(Optional) Deploy TLS Certificates” in *SAS Viya for Linux: Deployment Guide*.

For more information about configuring CAS, see “Configure CAS TLS to Use Custom Certificates (programming-only deployment)” on page 372 and “Configure CAS TLS to Use SAS Default Certificates (programming-only deployment)” on page 375. Certificates, keys, and passwords produced for authenticating to the SAS Embedded Process for Hadoop might, but do not need to coincide with such produced for other clients of the CAS server.

- When you installed the SAS Embedded Process, the following file and directory were created:

```
EPInstallDir/sasexe/SASEPHome/security/dcsecurity.properties
EPInstallDir/sasexe/SASEPHome/security/certs
```

- The `EPInstallDir/sasexe/SASEPHome/security/` directory should have `drwxr-xr-x` permissions.
- The `EPInstallDir/sasexe/SASEPHome/security/certs` directory should have `drwxr-xr-x` permissions.
- The `dcsecurity.properties` file should have `-rwxr-xr-x` permissions.

Enable Encryption for the SAS Data Connect Accelerator for Hadoop (on SAS Viya)

Follow these steps to encrypt data transfer between Hadoop and the CAS server using the SAS Data Connect Accelerator for Hadoop (on SAS Viya).

- 1 On Hadoop, modify the `dcsecurity.properties` file to enable SAS Data Connect Accelerator encryption.
 - a Navigate to the `EPInstallDir/sasexe/SASEPHome/security/` directory.
 - b Change the `DCTCPMENCRIPT` option in the `dcsecurity.properties` file as follows.

```
-DCTCPMENCRIPT YES
```

CAUTION! The **DCTCPMENCRIPT** option is set on both the CAS server and on the data provider. How the option is set on both sides determines whether the data being transferred is encrypted or not. For more information, see “[DCTCPMENCRIPT Option Setting Interaction](#)” on page 388.

- c Add either the `DCSSLACERTDIR` or `DCSSLALISTLOC` option to the `dcsecurity.properties` file to specify either the location of the trusted certificate authorities or the public certificate(s) for trusted certificate authorities.

Here is an example.

```
-DCSSLALISTLOC EPInstallDir/sasexe/SASEPHome/security/certs/certs-filename.pem
```

For more information about the options, see “[dcsecurity.properties File Options for Data Transfer with the SAS Data Connect Accelerator](#)” on page 445.

- 2 Copy the necessary TLS CA certificates to the `EPInstallDir/sasexe/SASEPHome/security/certs` directory.
 - If your CA certificates already exist on the Hadoop cluster, copy the TLS CA certificates to this directory.
 - If your CA certificates exist on the CAS server, using a method of your choice, copy the CA certificates to this directory on the Hadoop cluster. In the following example, `hdplus1` is the name of the Hadoop cluster.

```
scp CASCA1.pem username@hdplus1: EPInstallDir/sasexe/SASEPHome/security/certs
```

For more information about the location of the `trustedcerts.pem` file, see “[\(Optional\) Deploy TLS Certificates](#)” in *SAS Viya for Linux: Deployment Guide*.

Note:

- The CA certificates on the Hadoop cluster must authorize the identity certificates that are specified on the CAS server.
- The `EPInstallDir/sasexe/SASEPHome/security/` directory should have `drwxr-xr-x` permissions.
- The `EPInstallDir/sasexe/SASEPHome/security/certs` directory should have `drwxr-xr-x` permissions.
- The `dcsecurity.properties` file should have `-rwxr-xr-x` permissions.

- 3 Use the `sasep-admin.sh` script to copy the contents of the `EPInstallDir/sasexe/SASEPHome/security/` directory to all nodes on the Hadoop cluster.

- a Navigate to the `EPInstallDir/sasexe/SASEPHome/bin` directory.

```
cd EPInstallDir/sasexe/SASEPHome/bin
```

- b Run the `sasep-admin.sh` script with the `-security deploy` argument.

```
./sasep-admin.sh -security deploy
```

This script deploys the SAS Data Connect Accelerator security settings to all nodes on the cluster. For more information, see “[SASEP-ADMIN.SH Script](#)” in *SAS Viya in Linux: Deployment Guide*.

Note: You can use `sasep-admin.sh -security deploy -force` to overwrite the current settings.

DCTCPMENCRIPT Option Setting Interaction

The **DCTCPMENCRIPT** option must be set for both the CAS server and the data provider. How the option is set on both sides determines whether the data being transferred is encrypted, the data is sent in plaintext, or the data transfer fails. The following table describes the interaction.

DCTCPMENCRIPT	CAS setting - YES	CAS setting - NO	CAS setting - OPT
Data provider setting - YES	Data transfer - encrypted	Data transfer - fails	Data transfer - encrypted
Data provider setting - NO	Data transfer - fails	Data transfer - plaintext	Data transfer - plaintext
Data provider setting - OPT	Data transfer - encrypted	Data transfer - plaintext	Data transfer - encrypted

You might want to use the OPT setting if you have more than one cluster set up as a client. If you want one cluster to use encrypted data transfer and one cluster to use plaintext, you would set the DCTCPMENCRIPT option of the first cluster to YES and the DCTCPMENCRIPT option of the second cluster to NO. You would then set the DCTCPMENCRIPT option of the CAS server to OPT.

Note: During deployment of Viya 3.3, the DCTCPMENCRIPT option is set to OPT on the CAS server. You can change CAS server settings in the `casconfig_usermods.lua` file.

Updating the CAS Configuration File Options for Data Transfer

You can check the current run-time data transfer encryption settings of the CAS server by using the CAS Server Monitor. The settings are on the Runtime Environment panel of the System State page. For more information about the CAS Server Monitor, see [“Using CAS Server Monitor” on page 625](#).

CAS server options are stored in a configuration file. During deployment, the `casconfig_deployment.lua` is created in the `/opt/sas/viya/config/etc/cas/default` directory from content provided in the `vars.yml` file. When the `sas-viya-cascontroller-default` service is started, the options in the lua file are processed.

Changes to data transfer encryption options such as the DCTCPMENCRIPT option should be made in the `casconfig_usermods.lua` file.

For a complete list of options, see [“CAS Configuration File Options for Data Transfer with the SAS Data Connect Accelerator” on page 443](#).

Updating the dcsecurity.properties File Options for Data Transfer

On Hadoop or Teradata, the file options for data transfer encryption are located in the `dcsecurity.properties` file. The `dcsecurity.properties` file is located in the following directory on your cluster.

- For Teradata, `/opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security`
- For Hadoop, `EPInstallDir/sasexe/SASEPHOME/security`

After you update the `dcsecurity.properties` file, copy the file to all nodes of the cluster.

- For Teradata, do a parallel file transfer to push the `dcsecurity.properties` file to all nodes.
- For Hadoop, Use the `sasep-admin.sh` script to copy the contents of the `EPInstallDir/sasexe/SASEPHOME/security/` directory to all nodes on the Hadoop cluster. Run this command from the `EPInstallDir/sasexe/SASEPHOME/bin` directory.

```
./sasep-admin.sh -security deploy
```

For a complete list of options, see [“dcsecurity.properties File Options for Data Transfer with the SAS Data Connect Accelerator” on page 445](#).

Configure SAS 9.4 Clients to Work with SAS Viya

Configure a SAS 9.4 client to work with SAS Viya.

Note: You must have SAS administrator privileges to import certificates from SAS Viya.

- 1 Obtain the CA certificate that was used to sign the certificate that the CAS Server is using. On most SAS Viya deployments, the files can be found as follows:
 - In a SAS Viya full deployment, the file needed is the vault-ca.crt file, located at `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts`. The vault-ca.crt file contains two certificates. The first certificate is the SAS Viya root CA certificate issued by Vault and the second certificate is the SAS Viya intermediate CA certificate issued by Vault.
 - In a programming-only SAS Viya deployment, the trustedcerts.pem and trustedcerts.jks files are located at `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts/trustedcerts`.
- 2 On a Windows client, you need to import the SAS Viya root CA certificate and the intermediate certificate into the Windows certificate stores. These files have to be imported into the Windows truststore one at a time. Because the vault-ca.crt file contains two files, the SAS Viya root CA certificate and the SAS Viya intermediate CA certificate, you need to create two unique files, one containing the root CA and the other containing the intermediate CA certificates. Use a text editor and cut-and-paste as appropriate.

Each certificate in the file is denoted with a -----BEGIN CERTIFICATE----- and an -----END CERTIFICATE----- . Include the -----BEGIN CERTIFICATE----- and -----END CERTIFICATE----- header and footer in each of the two new files.

For our example, we named our new certificate files example_root.cer and example-intermediate-ca.cer. Save these two files on your Windows machine and then [add your certificates to the Windows CA stores](#).

- 3 If you have a Linux 9.4m5 client connecting to a CAS Server that is TLS enabled, perform the following steps:
 - a Copy the SAS Viya CA certificates (vault-ca.crt or the trustedcerts files) to a location on your SAS 9.4 deployment where you can access the certificates. The directory structure where the SAS 9.4 trusted CA certificates (trustedcerts.pem or trustedcerts.jks) are found is at `<SASHome>/SASSecurityCertificateFramework/1.1/cacerts/trustedcerts.pem`.
 - b Append the contents of the SAS Viya vault-ca.crt file (or the SAS Viya trustedcerts file) to the end of the SAS 9.4 trustedcerts file. There are various ways to add your certificates to the trustedcerts files:
 - Use the SAS Deployment Manager to [add your certificates to the trusted CA bundle](#).
 - Use a text editor to [add your certificates to the trustedcerts file](#).
 - c On the Linux server, set environment variable CAS_CLIENT_SSL_CA_LIST= to the trust list that the client uses to connect to the server.

```
export CAS_CLIENT_SSL_CA_LIST=
'<SASHome>/SASSecurityCertificateFramework/1.1/cacerts/trustedcerts.pem'
```

If your SAS 9.4 client is SAS Studio, you can add the export statement to the sasenv_local file that is located at `/SASHome/SASFoundation/bin`.

Note:

In the December 2017 release of SAS 9.4M5, the CAS_CLIENT_SSL_CA_LIST= environment variable does not need to be set.

Manage Certificates

Use Best Practices to Manage Certificates

SAS recommends the following best practices for managing certificates and securing your private keys.

- Place your server identity certificate chained to any intermediate certificates in the `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs` directory.

Intermediate certificates need to be added to the server identity certificate in a certificate chain. The server identity certificate must be the first certificate in the chain. The intermediate certificate must be second. This order is important to allow validation with the private key to be successful.

- If your custom root certificate is site-signed or is not already included in the Mozilla bundle of trusted CA certificates, then you need to manually add the root certificate to the `trustedcerts` files under the `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts` directory.

You should also place a copy of the root certificate that you are adding to the `trustedcerts` files in the same directory. The root certificate should have a `.crt` file extension. This ensures that if the playbook needs to be rerun to update the installation, then this root certificate is automatically included in the regeneration of the `trustedcerts` files.

For information, see [“Add Your Certificates to the Trust List or to a Certificate Chain” on page 392](#).

Note: Do not delete the `trustedcerts.jks` and the `trustedcerts.pem` files.

Note: Add your root certificate to the `trustedcerts.pem` and `trustedcerts.jks` files on every machine in the deployment.

- Place your private server key in the `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private` directory.
- Encrypt your private key when possible.
- Password-protect your private key file.
- Place your password in the `encryption.key` file as the first line of the file. The encryption key file is located at `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/`. This action ensures that if the playbook needs to be rerun to update the installation, the password in the `encryption.key` file will be used in the `keys.lua` file.

Managing Certificates Using Ansible Play Utilities

When using the Ansible playbook, the following utilities can be used to manage certificates on a SAS Viya full deployment. These utilities are run from the `sas_viya_playbook` directory.

rebuild-trust-stores.yml

On an Ansible controller machine, from the `sas_viya_playbook` directory, run the `rebuild-trust-stores.yml` play to incorporate the customer CA bundle of trusted certificates (from Consul configuration) into the various truststore files (`trustedcerts.pem` and `trustedcerts.jks`) on each machine in the Viya deployment.

```
ansible-playbook -i inventory.ini ./utility/rebuild-trust-stores.yml
```

distribute-httpd-certs.yml

This Ansible play adds your new custom certificate to `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts`. The play distributes copies of the certificate chain file to all machines with a name of `httpproxy-inventory name-ca.crt`. The play then rebuilds the `trustedcerts.pem` and `trustedcerts.jks` files and distributes them to every machine in the deployment.

On an Ansible controller machine, from the `sas_viya_playbook` directory, you can run the `distribute-httpd-certs.yml` play to distribute new certificates. On the Ansible controller machine, locate the utility file in the `/sas_viya_playbook/utility` directory.

```
ansible-playbook -i inventory.ini utility/distribute-httpd-certs.yml
```

Note: This utility works on a full and programming-only deployment.

renew-security-artifacts.yml

On the Ansible controller machine, you can run the `renew-security-artifacts.yml` play to refresh the Vault CA certificates, tokens, keys, and server certificates.


```
ansible-playbook -i inventory.ini sas_viya_playbook/renew-security-artifacts.yml
```

Add Your Certificates to the Trust List or to a Certificate Chain

Add Certificates to the Trustedcerts File

SAS provides a trusted list of root CA certificates at installation. This trusted list includes the Mozilla bundle of CA certificates, the default Apache httpd certificates, and the CA certificates issued by Vault (only in a SAS Viya full deployment). There are two files named trustedcerts that contain the trusted list of certificates, trustedcerts.pem and trustedcerts.jks.

- In a SAS Viya deployment, the trusted certificates are found at `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts`.
- In a SAS 9.4 deployment, the trusted certificates are found at `<SASHome>/SASSecurityCertificateFramework/1.1/cacerts`.

Note: To ensure that the additional trusted certificates are not overwritten when you update your deployment, place your trusted root CA certificates in the directories shown above.

You can add additional root CA certificates to the trust list of certificates for the file format needed, JKS or PEM.

The following steps show how to add your CA root certificates, self-signed certificates, or your chained certificates to the trustedcerts.pem file.

- 1 You can use a text editor to add your certificates in any order to the trustedcerts.pem file. Here is an example template of certificates that a trustedcerts.pem file might contain.

```
<PEM encoded sascaroot>
-----BEGIN CERTIFICATE-----
certificate string
-----END CERTIFICATE-----
<PEM encoded sassha2rootca>
-----BEGIN CERTIFICATE-----
certificate string
-----END CERTIFICATE-----
<PEM encoded digicertrrootca>
-----BEGIN CERTIFICATE-----
certificate string
-----END CERTIFICATE-----
<PEM encoded custom_ca_chain>
-----BEGIN CERTIFICATE-----
certificate string
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
certificate string
-----END CERTIFICATE-----
```

The content of the digital certificate in this example is represented as `<PEM encoded certificate>`. The content of each digital certificate is delimited with a `- - - - -BEGIN CERTIFICATE- - - - -` and `- - - - -END CERTIFICATE- - - - -` pair. All text outside the delimiters is ignored. Therefore, you might not want to use delimited lines for descriptive comments.

You can also concatenate the certificate authority files. For example, you can concatenate a root authority certificate file and a primary certificate file into a single PEM file. Here are two examples of concatenating certificates:

```
cat custom_ca_chain.pem >> trustedcerts.pem
cat vault-ca.crt >> trustedcerts.pem
```


Note: You can place these files in any order.

- 2 Because the digital certificate is encoded, it is unreadable. To view the file contents, you can use the following OpenSSL commands for your file type:

```
openssl x509 -in /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/cacerts/trustedcerts -text -noout
```

The following steps show how to add certificates to the trustedcerts.jks file (the Java truststore). Use the `keytool` command to add the certificates to the Java truststore. In this example, we assume that you have obtained a certificate from a CA not included in the truststore.

- 1 Locate the default truststore for your Java applications.
 - In a SAS Viya deployment, the trusted certificates are found at `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts/trustedcerts.jks`.
 - In a SAS 9.4 deployment, the trusted certificates are found at `<SASHome>/SASSecurityCertificateFramework/1.1/cacerts/trustedcerts.jks`.
- 2 Import the CA certificate into the default truststore. In our example, we are assuming that the file `root_ca.pem` contains the CA's certificate. Use the following commands to import a root CA certificate (`root_ca.pem` in our example) into the default truststore.

```
$ keytool -importcert -file /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tls/certs/cas/shared/default/custom.crt
-alias CAroot -keystore
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/
cacerts/trustedcerts.jks -storepass changeit
```

Note: Do not delete the trustedcerts files.

Note: Add your root certificate to the trustedcerts.pem and trustedcerts.jks files on every machine in the deployment.

For more information about how to manage your certificates and protect your keys, see [“Manage Certificates” on page 390](#).

Add Certificates to the Certificate Chain

The list of TLS certificates, from the root certificate to the end-user certificate, represents the TLS certificate chain. We take the server identity certificates and chain the intermediate certificates with them. In a SAS Viya deployment, these chained certificate files are placed in the `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs` directory.

For detailed information about how to manage your certificates and protect your keys, see [“Manage Certificates” on page 390](#).

The following steps show how to add your intermediate certificates to the server identity certificates file to create a chain:

- 1 You can use a text editor to add your certificates to a file in PEM format called `server.crt`.

Chaining files contain the server certificate first, the intermediate certificate that validates the server certificate next, the intermediate certificate that validates the first intermediate certificate next, and so on, down the chain all the way to the root certificate. The root certificate does not need to be in this file.

Here is a template of what a chained `server.crt` file might contain.

```
(Your Server Id Certificate)
- - - - -BEGIN CERTIFICATE- - - - -
<PEM encoded certificate>
- - - - -END CERTIFICATE- - - - -
(Intermediate Certificate(s))
```

```

- - - -BEGIN CERTIFICATE- - - -
<PEM encoded certificate>
- - - -END CERTIFICATE- - - -
(Intermediate Certificate(s))
- - - -BEGIN CERTIFICATE- - - -
<PEM encoded certificate>
- - - -END CERTIFICATE- - - -

```

The content of the digital certificate in this example is represented as `<PEM encoded certificate>`. The content of each digital certificate is delimited with a `- - - -BEGIN CERTIFICATE- - - -` and `- - - -END CERTIFICATE- - - -` pair. All text outside the delimiters is ignored. Therefore, you might not want to use delimited lines for descriptive comments.

You can also concatenate the certificate files. For example, you can concatenate an intermediate authority certificate file and a server certificate file into a single PEM file. Here is an example of concatenating certificates:

```
cat int_ca.crt >> server_id.crt
```

- 2 Because the digital certificate is encoded, it is unreadable. To view the file contents, you can use the following OpenSSL commands for your file type:

```
openssl x509 -in /config/etc/SASSecurityCertificateFramework/
tls/certs/server_id.crt -text -noout
```

Generate Site-Signed or Third-Party-Signed Certificates in PEM Format

You need to create two files, a private key file and a certificate file.

private key

This private key is in RSA format and is saved in ASCII (Base64-encoded) PEM (Privacy Enhanced Mail) format.

third-party-signed certificate

A certificate authority (CA) is a trusted third party. This certificate contains the CA's public key in X.509 certificate form and is saved in ASCII (Base64-encoded) PEM format.

SAS recommends the following best practices for managing certificates and securing your private keys for the CAS server.

- Place your server identity certificates in the `/config/etc/SASSecurityCertificateFramework/tls/certs` directory.

Intermediate certificates need to be added to the server identity certificate in a certificate chain. The server identity certificate must be the first certificate in the chain. The intermediate certificate must be second. This order is important to allow validation with the private key to be successful.

- If your custom root certificate is site-signed or is not already included in the Mozilla bundle of trusted CA certificates, then you need to manually add the root certificate to the `trustedcerts` files under the `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts` directory.

You should also place a copy of the root certificate that you are adding to the `trustedcerts` files in the same directory. The root certificate should have a `.crt` file extension. This ensures that if the playbook needs to be rerun to update the installation, then this root certificate is automatically included in the regeneration of the `trustedcerts` files.

For information, see [“Add Your Certificates to the Trust List or to a Certificate Chain” on page 392](#).

Note: Do not delete the `trustedcerts.jks` and the `trustedcerts.pem` files.

Note: Add your root certificate to the `trustedcerts.pem` and `trustedcerts.jks` files on every machine in the deployment.

- Place your private server keys in the `/config/etc/SASSecurityCertificateFramework/private` directory structure and reference this directory location in the environment variables that you are setting.
- Encrypt your private key when possible.

Note: This example is one way of possibly several to generate certificates for use with TLS. Consult your administrator for details about what is required for your site.

Generate site-signed or third-party-signed certificates in PEM format.

- 1 Decide which type of CA to use at your site.
 - site-signed
 - third-party-signed
- 2 Change the directory to where your OpenSSL commands reside. For example:

```
cd /usr/bin
```

- 3 Use the following OpenSSL command to generate a new private key in RSA format and a CA certificate signing request in PEM format. Store your private key in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private`.

```
openssl req -new -out /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tls/certs/certreq.csr
-newkey rsa:2048 -keyout /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/private.key -nodes
```

It is recommended that you supply an encrypted password on the key file. To do so, submit the following request.

```
openssl rsa -aes128 -in /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/private.key
-out /opt/sas/viya/config/etc/SASSecurityCertificateFramework/
private/tempprivate.key -passout pass:password

mv opt/sas/viya/config/etc/SASSecurityCertificateFramework/
private/tempprivate.key /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/private.key
```

- 4 Verify your Certificate Signing Request (CSR).


```
openssl req -noout -text -in /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tls/certs/certreq.csr
```
- 5 Submit your CSR file (`certreq.csr`) to your CA. This CA can be a CA at your site or a third party. You should receive the following certificates from your CA.
 - signed certificate (containing the CA's public key)
 - CA root certificate
 - One or more CA intermediate certificates
- 6 Store the signed certificates from your CA in `opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs`.
- 7 Add your site-signed root CA certificates to the truststore. See [“Add Your Certificates to the Trust List or to a Certificate Chain” on page 392](#).

See Also

For an example of using OpenSSL to generate site-signed or third-party-signed certificates in PEM format, see [“Create Site-Signed or Third-Party-Signed Certificates in PEM Format” on page 447](#).

Generate Site-Signed or Third-Party-Signed Certificates in Java Keystore Format

The following steps create site-signed or third-party-signed certificates in Java keystore (JKS) format. Details of each step are shown after this summary.

- 1 Create the machine's keystore.
- 2 Create a certificate signing request (CSR).
- 3 Submit a .csr file to a CA.
- 4 Receive a signed certificate, CA root certificate, and one or more CA intermediate certificates.
- 5 Add the server's identity certificate to the keystore.
- 6 Add the CA intermediate certificate to the keystore.

Note: This example is one way of possibly several to generate certificates for use with TLS. Consult your administrator for details about what is required for your site.

The keystore contains private keys and certificates used by TLS servers to authenticate themselves to TLS clients. By convention, such files are referred to as keystores.

SAS recommends the following best practices for managing certificates for Java.

- The signed certificate and private key are contained in one JKS format file. Add your certificates to the `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/java/jks` directory.
- Password-protect the private key.
- Password-protect the keystore. In the following example the keystore file is named `keystore.jks`.
- Make the keystore file readable only by members of the appropriate group.
- Make the file where the keystore password is referenced readable only by members of the appropriate group. For example, you might make the `init_usermods.properties` file (where the password is referenced by a keystore password property)) readable only by members of the appropriate group.

You can obtain site-signed or third-party-signed certificates using the Java Keytool. In the following scenario that we are using a certificate authority (CA) as our third party.

- 1 Log on to your machine as a user with root or sudo privileges.
- 2 Change the directory to where your `keytool` command resides. For example:

```
cd $JAVA_HOME/bin
```

- 3 Use the `keytool` command to create a new private key and keystore and store the information in the keystore file named `keystore.jks`. In the following example, we are first generating a private key `server.key`. We are also using alias `server`.

```
keytool -genkey -alias server -keyalg RSA -keystore
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/
java/jks/keystore.jks -storepass changeit -keypass password
-validity 360 -keysize 2048
```

The keystore password (which protects the keystore as a whole) and the key password (which protects the private key stored in the server entry) are set using the `-storepass` and `-keypass` options respectively.

Change the permissions on the keystore file (`keystore.jks`) to be readable only by members of the appropriate group. Use `chmod` or `sudo` to change the permissions.

```
chmod 600 keystore.jks
```

When you list the file, you see the permissions are Read/Write only `-rw-----` .

- 4 To query the contents of your Java keystore file, you can use the following command:

```
keytool -list -v -keystore /opt/sas/viya/config/etc
/SASSecurityCertificateFramework/java/jks/keystore.jks
-storepass changeit -keypass password
```

- 5 Use the `keytool` command to create certificate signing request (CSR) for an existing keystore. Here is an example command:

```
keytool -certreq -alias server -keystore
/opt/sas/viya/config/etc/SASSecurityCertificateFramework
/java/jks/keystore.jks -file
/opt/sas/viya/config/etc/SASSecurityCertificateFramework
/java/jks/server.csr -storepass changeit -keypass password
```

This command generates the CSR and stores it in a file called `server.csr`.

- 6 Submit your CSR file to your CA. For our example, we have provided a name for each of the signed certificates that we might receive, `server_ca.pem`, `root_ca.pem`, and `int_ca.pem`. You should receive the following from your CA:
- signed identity certificate (`server_ca.pem`)
 - CA root certificate (`root_ca.pem`)
 - one or more CA intermediate certificates (`int_ca.pem`)
- 7 After you have submitted your CSR to the CA and received the CA's reply (containing the signed certificate), import the reply into your keystore, located at `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/java/jks`, using the following `keytool` options.

This step imports the signed server identity certificate and one or more intermediate certificates in PEM format into the keystore.

- a Add the server identity certificate to your keystore. In this example, `server_cert.pem` is the server identity certificate.

```
keytool -importcert -file /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/java/jks/server_ca.pem
-keystore /opt/sas/viya/config/etc/SASSecurityCertificateFramework/
java/jks/keystore.jks -storepass changeit -keypass password
-trustcacerts -alias server_ca
```

- b If your server certificate is signed by an intermediate CA, import the intermediate certificate into your keystore file. In this example, `int_ca.pem` is the CA intermediate certificate.

```
keytool -importcert -file /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/java/jks/int_ca.pem -keystore
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/java/
jks/keystore.jks -storepass changeit -keypass password
-trustcacerts -alias int_ca
```

- c Verify that the certificates that you added to your keystore are present.

```
keytool -v -list -keystore /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/java/jks/keystore.jks
-storepass changeit -keypass password
```

Generate Self-Signed Certificates

Self-signed certificates are signed by your own private key, rather than by an external CA. You can generate self-signed certificates or root certificates in PEM format using RSA or HMAC encryption or in Java keystore format.

A private key file and a self-signed certificate are needed.

private key

This private key is in RSA format and is saved in ASCII (Base64-encoded) PEM format.

self-signed certificate

This certificate contains a public key in X.509 certificate form and is saved in ASCII (Base64-encoded) PEM format.

SAS recommends the following best practices for managing certificates and securing your private keys.

- Place your server identity certificates in the `/config/etc/SASSecurityCertificateFramework/tls/certs` directory.

Intermediate certificates are added to the server identity certificate in a certificate chain. The server identity certificate must be the first certificate in the chain. The intermediate certificate must be second. This order is important to allow validation with the private key to be successful.

- Place your root certificates and trusted CA certificates in the `/config/etc/SASSecurityCertificateFramework/cacerts` directory.
- Place your private server keys in the `/config/etc/SASSecurityCertificateFramework/private` directory structure and reference this directory location in the environment variables that you are setting.
- Encrypt your private key when possible.
- For Java, store your private key and signed certificates in `opt/sas/viya/config/etc/SASSecurityCertificateFramework/java/jks`.

Generate self-signed certificates or root certificates in PEM format using RSA encryption.

Note: This example is one of several possible ways to generate certificates for use with TLS. Consult your administrator for details about what is required for your site.

- 1 Change the directory to the directory where your OpenSSL commands reside. For example:

```
cd /usr/bin
```

- 2 Use the following OpenSSL command to generate a new private key using RSA encryption and a self-signed certificate:

```
openssl req -x509 -newkey rsa:2048 -keyout
/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/private.key
-out /opt/sas/viya/config/etc/SASSecurityCertificateFramework/
tls/certs/certreq.csr -days 1000
```

It is recommended that you supply an encrypted password on the key file. To do so, submit the following request:

```
openssl rsa -aes128 -in /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/private.key
-out /opt/sas/viya/config/etc/SASSecurityCertificateFramework/
private/tempprivate.key -passout pass:password
mv opt/sas/viya/config/etc/SASSecurityCertificateFramework/
private/tempprivate.key /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/private.key
```

- 3 If you need to add your certificates to the trusted list of certificates, see [“Add Your Certificates to the Trust List or to a Certificate Chain”](#) on page 392..

Generate self-signed certificates in Java keystore format.

Note: This example is one of several possible ways to generate certificates for use with TLS. Consult your administrator for details about what is required for your site.

- 1 For servers based on Java, generate a self-signed certificate using `keytool -genkeypair`. This command creates a public/private key pair and wraps the public key into a self-signed certificate. For example, the following command creates a self-signed test certificate for the host and stores it in a keystore. For this example, we are using alias `javahost`.

```
$ keytool -genkeypair -keystore
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/
java/jks/javahost.jks -keyalg RSA -alias javahost
-dname "CN=javahost.example.com,O=Hadoop" -storepass changeit
-keypass password -validity 1000
```

Note: By default, self-signed certificates are valid for only 90 days. To increase this period, replace the previous command's `-validity <val_days>` parameter to specify the number of days for which the certificate should be considered valid.

- 2 If you need to add your certificates to the trusted list of certificates, see [“Add Your Certificates to the Trust List or to a Certificate Chain” on page 392](#).

Convert Digital Certificate File Formats Using OpenSSL

In OpenSSL, you can use many parameters to convert between the different digital certificate file formats. The following are some examples of a few ways to convert files from one format to another. See [OpenSSL TLS Toolkit](#) for more commands that can be used.

Convert DER to PEM File Format

Many certificate authorities provide certificates in DER format. If you have a DER formatted file, but need a PEM formatted file, you can convert the DER formatted file to PEM format using OpenSSL.

Note: You must convert a DER formatted file to PEM format before you can include it in a trust list on Linux.

Here is an example of how to convert a server digital certificate from DER input format to PEM output format:

```
x509 -inform DER -outform PEM -in certificate.cer -out certificate.pem
```

Convert PEM Encoded Certificate to DER File Format

If you have a PEM formatted file, but need a DER formatted file, you can convert the PEM formatted file to DER using OpenSSL.

Here is an example of how to convert a server digital certificate from PEM input format to DER output format:

```
x509 -outform der -in certificate.pem -out certificate.der
```

Convert PEM to PK12 File Format

If you are using a Java application that accepts only PKCS#12 format, you might need to convert your PEM formatted file that includes certificates and the separate key file to one file that includes both the certificate and the key file.

If you have a PEM formatted certificate file, but need a PK12 formatted file, you can convert the PEM format certificate to a PK12 format using OpenSSL. Here is one way of converting a PEM to a PK12 formatted file for non-FIPS (Federal Information Processing Standard) libraries.

```
pkcs12 -export -out path/certificate.p12 -inkey path/privatekey.pem
-in path/certificate.pem -certfile certs.pem
```

```
pkcs12 -export -in server.cer -inkey server.key -out keystore.p12
-name server
```


Secure Credentials in the CAS Server (cas.servicesbaseurl)

Note: This section is applicable only if you have a full deployment. If you have a programming-only deployment, skip this section.

The URL that enables a CAS server to use SAS Viya services is set using the `cas.SERVICESBASEURL=` option. For example, CAS client credentials are passed to the SASLogon service at the address specified in the `cas.SERVICESBASEURL=` option in order to obtain an OAuth token.

This option is set in the `casconfig.lua` file located at `/opt/sas/viya/config/etc/cas/default/`.

- 1 In the `casconfig.lua` file, ensure that the HTTPS URL is used to access the Apache HTTP server machine.

```
cas.servicesbaseurl='https://webserver-host-name'
```

Note: The host name in the URL is the same as the Common Name used in the server identity certificate that Apache HTTP Server is using.

Note: In a SAS Viya full deployment, the `cas.SERVICESBASEURL=` option defaults to port 443 for HTTPS access.

- 2 When you set the `cas.SERVICESBASEURL=` option to use HTTPS, the `CAS_CALISTLOC=` environment variable needs to be set in the `casconfig_usermods.lua` file to point to the CA certificates that the Apache HTTP Server is using.

```
env.CAS_CALISTLOC=
'/path-to-CA-chain-used-for-Apache-HTTP-Server-certificate'
```

Note: If the CA certificates are already imported in the OpenSSL truststore, setting the `env.CAS_CALISTLOC=` environment variable is not necessary.

- 3 If you are setting the `CAS_CALISTLOC=` environment variable, you should copy the change made to this environment variable to the `vars.yml` file. This change ensures that your settings are not changed when upgrades are made to the deployment.

Note: See [“Modify the vars.yml File” in SAS Viya for Linux: Deployment Guide](#) for more details.

Add the following highlighted variables and their respective values:

```
CAS_CONFIGURATION:
  env:
    #CAS_DISK_CACHE: /tmp
    CAS_CLIENT_SSL_REQUIRED: 'true'
    CAS_CALISTLOC: path-to-CA-chain-used-for-Apache-HTTP-Server-certificate
  cfg:
    #gcport: 5580
    #httpport: 8777
    #port: 5570
    #colocation: 'none'
    servicesbaseurl: 'https://http-proxy-host-name'
```

Save and close the `vars.yml` file.

For information about using `cas.SERVICESBASEURL=`, see See [“Configuration File Options Reference” on page 627](#).

Manage Tokens and Create JWT Signing Keys

Generate Signing Keys for JSON Web Tokens

Overview

A JSON Web Token (JWT) is a JSON object that is defined in [RFC 7519](#) as a safe way to pass a set of information between two parties. Access tokens issued by SAS Logon Manager are also OpenID Connect ID tokens, which are JWTs.

The token consists of three parts: a header, claims, and a signature. All of these parts are base64 encoded. Here is what an example token might look like. Each part is separated by a period to create header.claims.signature.

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjMONTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoiOnRydWV9.TjVA95OrM7E2cBab30RMHrHDcEfxjoYZgeFONFh7HgQ
```

In a new Viya deployment, SAS Logon Manager generates RSA keys and CAS gets the public key that it needs from SAS Logon Manager automatically if the `cas.SERVICESBASEURL=` property is set. You can configure your own signing keys, overriding the SAS Logon Manager behavior. You can also supply a passphrase that is then hashed using HMAC.

To configure your JWT keys, the following configuration file options and properties need to be set for CAS and SAS Logon Manager.

`cas.OAUTHSIGNINGKEY=`

The signing key must be either a Base64-encoded RSA private key that is used to digitally sign tokens, or a passphrase. See [“Configuration File Options” on page 627](#).

Note: CAS gets the public key that it needs from SAS Logon Manager automatically if the `cas.SERVICESBASEURL=` is set.

`cas.OAUTHSIGNINGCERTIFICATE=`

The RSA public key corresponding to the RSA private key being used to digitally sign tokens. See [“Configuration File Options” on page 627](#).

`sas.logon.jwt`

The set of properties that are used to secure JSON web tokens with RSA digital signatures or hashed message authentication codes (HMACs). For a description of the properties, see [“Configuration Properties: Reference \(Applications\)” on page 242](#).

Generate JWT Signing Key

The following example uses OpenSSL to generate an RSA signing key.

Note: This example is one way of many to generate RSA signing keys. Consult your administrator for details about what is required for your site.

- 1 Change the working directory to the directory where SAS stores keys. SAS stores keys in the `/opt/sas/viya/config/etc/SASSecurityCertificateFramework` directory structure. For example:

```
cd /opt/sas/viya/config/etc/SASSecurityCertificateFramework
```

- 2 Use the following OpenSSL command to generate a new RSA private key.

```
openssl genrsa -out /opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/jwt-private.key 2048
```

- 3 Create a public directory.

```
mkdir public/
```

- 4 Extract the RSA public key. Submit the following request:

```
openssl rsa -in "./private/jwt-private.key" -out
"./public/jwt-public.key" -pubout/public/jwt-public.key
```

- 5 Copy the public key to the signingKey property using SAS Environment Manager. See [“Configure the SAS Logon Manager with New JWT Signing Key”](#) on page 402.

- 6 You can add the following environment variables to casconfig_usermods.lua file.

- Add cas.OAUTHSIGNINGCERTIFICATE= to the casconfig_usermods.lua file at /opt/sas/viya/config/etc/cas/default.

```
cas.oauthsigningcertificate="/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/public/jwt-public.key"
```

- Add cas.OAUTHSIGNINGKEY= to the keys.lua file located at /opt/sas/viya/config/etc/cas/default. The keys.lua file has limited access and is more secure.

```
cas.oauthsigningkey='zAcSGqF23Fu85e7qz7ZN2U4ZRhfV3W\
pwPPAoE3Z7kBw&SsiodoUaIvY8ltyTt5jkRh4J50vUPVWHaR7YPi5jC'
```

Ensure that the permissions on the keys.lua file is 600 and is readable only by CAS service account.

```
chmod 600 keys.lua
```

For information about the CAS configuration file options, see [“Configuration File Options”](#) on page 627.


The following example uses a simple passphrase. The server will use HMAC to generate the digital signature from the passphrase.

```
cas.oauthsigningkey="57443a4c052350a44638835d64fd66822f813319"
```

Configure the SAS Logon Manager with New JWT Signing Key

Use the SAS Environment Manager to set configuration properties that are used by the SAS Logon Manager. If you created a new JWT signing key, paste the key into the signingKey property.

- 1 From the side menu , select **SAS Environment Manager**.


- 2 In the navigation bar, click .

The Configuration page is an advanced interface. It is available to only SAS Administrators.

- 3 The default view is **Basic Services**. Select **Definitions** from the drop-down box.

- 4 In the **Definitions** list, select **sas.logon.jwt**.

- 5 If the definition has no properties configured, complete the following:

- a In the top right corner of the window, click .

- b In the New sas.logon.jwt Configuration dialog box, paste the PEM-encoded RSA private key or passphrase into the value for the signingKey property.

For a description of the properties, see [“Configuration Properties: Reference \(Applications\)”](#) on page 242.

- c Click **Save**.

Note: The system will take a few minutes to recognize the new key before starting to use the new key.

Obtain an Access Token to Register a New Client ID

You can use a curl command to obtain a token. You can then use that token to register a new client ID.

Consul tokens can be found at `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/consul/default` and is named `client.token`.

An example curl command to request a registration token for a new client follows. In this example, the client is named APP.

```
curl -X POST "http://localhost/SASLogon/oauth/clients/consul?callback=false&serviceId=app"
-H "X-Consul-Token: 29c4700f-ea89-41cd-8bc4-4198ccaa5bf9"
```

Note:

This request must pass a `callback=false` query string parameter and authenticate directly by passing a SAS Configuration Server (Consul) token. If the Consul token is valid, SASLogon returns the registration token in the response.

By default, tokens are valid for 12 hours. When you register the client ID, you can configure the amount of time that the token is valid. See how to register a client ID in “[Authentication: How To](#)” on page 31.

For more information about using curl, see [Curl Documentation](#).

Replace Tokens for SAS Configuration Server (Consul)

Overview

At installation, tokens are generated and placed in the `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/consul/default` directory. Client, encryption, and management tokens are provided. The owner and group of these files is SAS.

`client.token`

is the ACL client token that is used by all services to access values in the Key/Value store.

`management.token`

is the ACL management token (`acl_master_token`) that is used to administer the ACLs.

`encryption.token`

specifies the secret key that is used for encryption of Consul network traffic. Used for gossip communication.

You must use the value of an ACL token that is of type management to administer Consul ACLs. The value of this management token is created by the Ansible playbook and stored in the `management.token` file at `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/consul/default`.

Replace ACL Tokens

You must use the value of an ACL token that is of type management to administer Consul ACLs. In the following example, the `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/consul/default/management.token` file contains the ID of a management ACL that we want to change.

We are using the `sas-bootstrap-config` CLI to replace ACL tokens.

Note: You can also use the Consul ACL HTTP CLI to manage ACL tokens. For more information, see [ACL HTTP Endpoint](#).

- 1 Source the `/etc/profile.d/lang.sh` to set the LANG environment variable. It will be set to a value such as `en_US.UTF-8`

```
source /etc/profile.d/lang.sh
```

- 2 The `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/consul/default/management.token` file contains the ID of a management ACL that we want to change.

```
sudo cat /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tokens/consul/default/management.token
0329addc-bb72-489c-9f0a-5421890dd2fb
```

- 3 Create a backup copy of the original `management.token`.

```
sudo cp /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tokens/consul/default/
management.token /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tokens/consul/default/
management.token.OLD
```

- 4 List the ACLs using the `sas-bootstrap-config` CLI.

```
sudo /opt/sas/viya/home/bin/sas-bootstrap-config
--token-file /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tokens/consul/default/
management.token acl list
```

These are the ACLs listed.

```
{
  "CreateIndex": 4,
  "ModifyIndex": 4,
  "ID": "0329addc-bb72-489c-9f0a-5421890dd2fb",
  "Name": "Master Token",
  "Type": "management",
  "Rules": ""
},
{
  "CreateIndex": 3,
  "ModifyIndex": 64718,
  "ID": "anonymous",
  "Name": "Anonymous Token",
  "Type": "client",
  "Rules": "{\"service\":{\"\":{\"Policy\":\"read\"}}}"
},
{
  "CreateIndex": 19,
  "ModifyIndex": 64702,
  "ID": "eaa6de8a-3824-4c8f-a73a-dbd835c5cc97",
  "Name": "client",
  "Type": "client",
  "Rules": "{\"key\":{\"\":{\"Policy\":\"write\"}},\"service\":
{\"\":{\"Policy\":\"write\"}},\"event\":{\"\":{\"Policy\":\"write\"}},
\"query\":{\"\":{\"Policy\":\"write\"}}}"
}
```

- 5 Clone the management token using the following command. The `c43b7d1a-ccee-3792-a1d8-9576a9dbe7d2` ID is returned by the execution of the following code. This ID is the new value that is inserted into the `management.token` file at `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/consul/default`.

```
sudo /opt/sas/viya/home/bin/sas-bootstrap-config --token-file
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/
tokens/consul/default/management.token acl clone --acl-id
$(sudo cat /opt/sas/viya/config/etc/SASSecurityCertificateFramework/
```

```
tokens/consul/default/management.token)
{
  "ID": "c43b7d1a-ccee-3792-a1d8-9576a9dbe7d2"
}
```

6 List the ACLs again to verify that the new management ACL has been created.

```
sudo /opt/sas/viya/home/bin/sas-bootstrap-config --token-file
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/
consul/default/management.token acl list
```

Here are the ACLs listed now.

```
{
  "CreateIndex": 4,
  "ModifyIndex": 4,
  "ID": "0329addc-bb72-489c-9f0a-5421890dd2fb",
  "Name": "Master Token",
  "Type": "management",
  "Rules": ""
},
{
  "CreateIndex": 3,
  "ModifyIndex": 64899,
  "ID": "anonymous",
  "Name": "Anonymous Token",
  "Type": "client",
  "Rules": "{\"service\":{\"\":{\"\":{\"Policy\": \"read\"}}}"
},
{
  "CreateIndex": 64927,
  "ModifyIndex": 64927,
  "ID": "c43b7d1a-ccee-3792-a1d8-9576a9dbe7d2",
  "Name": "Master Token",
  "Type": "management",
  "Rules": ""
},
{
  "CreateIndex": 19,
  "ModifyIndex": 64897,
  "ID": "eaa6de8a-3824-4c8f-a73a-dbd835c5cc97",
  "Name": "client",
  "Type": "client",
  "Rules": "{\"key\":{\"\":{\"Policy\": \"write\"}},
  \"service\":{\"\":{\"Policy\": \"write\"}}, \"event\":
  {\"\":{\"Policy\": \"write\"}}, \"query\":{\"\":{\"Policy\": \"write\"}}}"
}
```

7 Replace the value in the management.token file with the value that was returned from the clone command.

```
sudo bash -c 'echo c43b7d1a-ccee-3792-a1d8-9576a9dbe7d2 >
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/
consul/default/management.token'
```

8 Destroy the old management ACL.

```
sudo /opt/sas/viya/home/bin/sas-bootstrap-config
--token-file /opt/sas/viya/config/etc/SASSecurityCertificateFramework/
tokens/consul/default/management.token acl destroy --acl-id $(sudo cat
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/
tokens/consul/default/management.token.OLD)
```

- 9 After the new ACLs have been created in Consul and the `management.token` and `client.token` files have been updated with the new values, copies of the original `.token` files can be deleted.

Replace ACL Tokens Using the `sas-crypto-management` Tool

You can use the `sas-crypto-management` application located at `/opt/sas/viya/home/SASSecurityCertificateFramework/bin/` to generate a value that can be used as the ID for an ACL.

```
sudo /opt/sas/viya/home/bin/SASSecurityCertificateFramework/
bin/sas-crypto-management uuid --out-file /opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tokens/consul/default/client.token
```

You can then use the value of the ACL ID that was generated using the `sas-crypto-management` tool (instead of the value that Consul generates using its `clone` command as shown in [“Replace ACL Tokens” on page 403](#)). Then, use the `create` command to specify the ID that should be used.

Replace an Encryption Token

All Consul agents that are running as servers or clients need to have an encryption key. The Consul agent supports encrypting all of its network traffic. The `SASSecurityCertificateFramework` provides the encryption token that is used for gossip communication. Enabling gossip encryption requires only that you set an encryption key when starting the Consul agent.

The Consul RPM start script generates a file named `config-gossip.json` in `/opt/sas/viya/config/etc/consul.d`. The Consul RPM uses the value obtained from the `gossip.token` file in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/consul/default`. You can see the type of information contained in the file by submitting the following command:

```
sudo cat /opt/sas/viya/config/etc/consul.d/config-gossip.json
```

The generated file contains encryption information that looks like the following. The encryption key is 16-bytes and Base64 encoded.

```
{ "encrypt": "y/k+KRpeZZVmzHCVrvbR6A==" }
```

The `-encrypt` option specifies the secret key to use for encryption of Consul network traffic. This key must be 16-bytes that are Base64-encoded. All nodes within a cluster must share the same encryption key to communicate. The provided key is automatically persisted to the data directory and loaded automatically whenever the agent is restarted. More information about this option can be found at [Consul Configuration Command-line Options](#).

There are situations when the encryption token might need to be replaced.

- 1 Sign on to the machine that runs the SAS Configuration Server (Consul) as the SAS install user (`sas`) or with `sudo` privileges.
- 2 Consul provides the `consul keygen` command to generate a new key.

```
consul keygen
```

- 3 Copy the value that is generated (for example, `X4SYOinf2pTAcAHRhpj7dA==`) and open `config-gossip.json` located at `/opt/sas/viya/config/etc/consul.d/`.

Use the copied string as the value for the `encrypt` parameter:

```
"encrypt": "X4SYOinf2pTAcAHRhpj7dA=="
```

- 4 Stop Consul.

```
sudo service sas-viya-consul-default stop
```

- 5 Delete the `local.keyring` and `remote.keyring` files in `/opt/sas/viya/config/data/consul/serf`.

All nodes within a cluster must share the same encryption key to communicate. The provided key is automatically persisted to the data directory and loaded automatically whenever the agent is restarted. This

option is provided on each agent's initial start-up sequence. The value of this secret key is persisted to the `/opt/sas/viya/config/data/consul/serf` directory to files `local.keyring` and `remote.keyring`.

Note: If a key is provided after Consul has been initialized with an encryption key, then the provided key is ignored and a warning is displayed.

6 Restart SAS Configuration Server (Consul).

```
sudo service sas-viya-consul-default restart
```

When Consul is restarted, the Consul RPM start script regenerates the `config-gossip.json` file and Consul reads this value and re-creates the `local.keyring` and `remote.keyring` files.

You can read about how this is done for Consul at [Encryption for Consul](#).

Concepts

SAS/SECURE

SAS/SECURE Overview

Refer to “[NETENCRYPT System Option](#)” on page 418 and “[NETENCRYPTALGORITHM= System Option](#)” on page 419 for details. You can specify various encryption algorithms as well as TLS to secure data in motion.

Linux supports the following encryption algorithms:

- RC2
- RC4
- DES
- TripleDES
- AES

Refer to “[Encryption Algorithms](#)” on page 416 for more information about encryption algorithms supported for use with SAS/SECURE.

SAS/SECURE Software Availability

For software delivery purposes, SAS/SECURE is a product within the SAS System. SAS/SECURE is included with the SAS Viya software. In prior releases, SAS/SECURE was an add-on product that was licensed separately. This change makes strong encryption available in all deployments (except where prohibited by import restrictions).

SAS/SECURE Export Restrictions

For U.S. export purposes, SAS designates each product based on the encryption algorithms and the product's functional capability. SAS/SECURE is available to most commercial and government users inside and outside the U.S. However, some countries (for example, Russia, China, and France) have import restrictions on products that contain encryption, and the U.S. prohibits the export of encryption software to specific embargoed or restricted destinations.

SAS/SECURE for Linux includes the following encryption algorithms.

- RC2 using up to 128-bit keys
- RC4 using up to 128-bit keys

- DES using up to 56-bit keys
- TripleDES using up to 168-bit keys
- AES using 256-bit keys

SAS/SECURE Installation and Configuration

SAS/SECURE is installed and delivered on every installation. Whether SAS/SECURE is used depends on the options that are set.

To use encryption provided by SAS/SECURE for communications and networking, specify the NETENCRYPT system option and set the NETENCRALG= system option to a value of RC2, RC4, DES, TRIPLEDES, AES, or SSL. Refer to [“NETENCRYPT System Option” on page 418](#) and [“NETENCRYPTALGORITHM= System Option” on page 419](#).

Transport Layer Security (TLS)

Transport Layer Security (TLS) Overview

Transport Layer Security (TLS) and its predecessor, Secure Sockets Layer (SSL), are cryptographic protocols that are designed to provide communication security. TLS and SSL are protocols that provide network data privacy, data integrity, and authentication.

Note: All discussion of TLS is also applicable to the predecessor protocol, Secure Sockets Layer (SSL).

TLS uses X.509 certificates and hence asymmetric cryptography to assure the party with whom they are communicating and to exchange a symmetric key. As a consequence of choosing X.509 certificates, certificate authorities and a public key infrastructure are necessary to verify the relation between a certificate and its owner, as well as to generate, sign, and administer the validity of certificates. For information about certificates, see [“Certificates” on page 410](#).

In addition to providing encryption services, TLS performs client and server authentication, and it uses message authentication codes to ensure data integrity. The client requests a certificate from the server, which it validates against the public certificate of the certificate authority used to sign the server certificate. The client then verifies the identity of the server and negotiates with the server to select a cipher (encryption method). The cipher that is selected is the first match between the ciphers that are supported on both the client and the server. All subsequent data transfers for the current request are then encrypted with the selected encryption method.

TLS System Requirements

SAS supports TLS on the Linux operating environment.

TLS Software Availability

SAS Viya supports TLS version 1.2 on Linux.

The default minimum protocol for OpenSSL is TLS 1.2.

SAS Viya uses the OpenSSL libraries provided for the operating systems on your system and OpenSSL libraries installed on your machine.

TLS Installation and Configuration

SAS Viya supports TLS on Linux using the operating system's OpenSSL libraries. However, in SAS Viya, the Apache HTTP Server is used as the web server and uses `mod_ssl` and the `sas-ssl.conf` file provided by SAS to provide TLS configuration.

In SAS Viya, the deployment provides the following support for TLS configuration.

- On the Apache HTTP Server, the module called `mod_ssl` provides TLS support. This module relies on OpenSSL to provide the cryptography engine. In addition, SAS Viya includes customizations to support SAS internal standards for developing software that protects data in motion.
- The Apache HTTP Server (web server) has localhost certificate and key files that allow HTTPS access to SAS Studio, CAS Server Monitor, and SAS Visual Analytics.
- Each machine in the deployment has a Mozilla bundle of trusted CA certificates in the `SASSecurityCertificateFramework` that is used by SAS and Java processes if TLS for CAS is turned on.
- The `SASSecurityCertificateFramework` also generates encrypted self-signed certificates for the CAS controller machine in the deployment that can be used to turn on TLS for CAS. These self-signed certificates are part of the Mozilla bundle of trusted CA certificates (`trustedcerts` files).

For more information about Certificates and how SAS Viya uses them with TLS, see [“Certificates” on page 410](#).

For information about configuring TLS, see [“Configure TLS and HTTPS ” on page 363](#).

TLS Terminology

The following concepts are fundamental to understanding TLS:

certificate authorities (CAs)

Cryptography products provide security services by using digital certificates, public-key cryptography, private-key cryptography, and digital signatures. Certificate authorities (CAs) create and maintain digital certificates, which also help preserve confidentiality.

Various commercial CAs, such as VeriSign and Thawte, provide competitive services for the e-commerce market. You can also develop your own CA by using products from companies such as RSA Security and Microsoft or from the Open-Source Toolkit OpenSSL.

digital signatures

A digital signature affixed to an electronic document or to a network data packet is like a personal signature that concludes a hand-written letter or that validates a credit card transaction. Digital signatures are a safeguard against fraud. A unique digital signature results from using a private key to encrypt a message digest. A document that contains a digital signature enables the receiver of the document to verify the source of the document. Electronic documents are said to be verified if the receiver knows where the document came from, who sent it, and when it was sent.

Another form of verification comes from message authentication codes (MAC), which ensure that a signed document has not been changed. A MAC is attached to a document to indicate the document's authenticity. A document that contains a MAC enables the receiver of the document (who also has the secret key) to know that the document is authentic.

digital certificates

Digital certificates are electronic documents that ensure the binding of a public key to an individual or an organization. Digital certificates provide protection from fraud.

Usually, a digital certificate contains a public key, a user's name, and an expiration date. It also contains the name of the certificate authority (CA) that issued the digital certificate and a digital signature that is generated by the CA. The CA's validation of an individual or an organization allows that individual or organization to be accepted at sites that trust the CA.

public and private keys

Public-key cryptography uses a public and a private key pair. The public key can be known by anyone, so anyone can send a confidential message. The private key is confidential and known only to the owner of the key pair, so only the owner can read the encrypted message. The public key is used primarily for encryption, but it can also be used to verify digital signatures. The private key is used primarily for decryption, but it can also be used to generate a digital signature.

symmetric key

In symmetric key encryption, the same key is used to encrypt and decrypt the message. If two parties want to exchange encrypted messages securely, they must both have a copy of the same symmetric key. Symmetric

key cryptography is often used for encrypting large amounts of data because it is computationally faster than asymmetric cryptography. Typical algorithms include DES, TripleDES, RC2, RC4, and AES.

asymmetric key

Asymmetric or public key encryption uses a pair of keys that have been derived together through a complex mathematical process. One of the keys is made public, typically by asking a CA to publish the public key in a certificate for the certificate-holder (also called the subject). The private key is kept secret by the subject and never revealed to anyone. The keys work together where one is used to perform the inverse operation of the other: If the public key is used to encrypt data, only the private key of the pair can decrypt it. If the private key is used to encrypt, the public key must be used to decrypt. This relationship allows a public key encryption scheme where anyone can obtain the public key for a subject and use it to encrypt data that only the user with the private key can decrypt. This scheme also specifies that when a subject encrypts data using its private key, anyone can decrypt the data by using the corresponding public key. This scheme is the foundation for digital signatures.

Certificates

About Certificates

Certificates are required for configuring TLS and HTTPS.

Digital certificates are used in a network security system to guarantee that the two parties exchanging information are really who they claim to be. Certificates are used to authenticate a server process or a human user. Digital certificates are issued by a certificate authority (CA).

A CA is an organization that verifies the information or the identity of computers on a network and issues digital certificates of authenticity and public keys. As part of a public key infrastructure (PKI), a CA checks with a registration authority to verify information provided by the requestor of a digital certificate. If the registration authority verifies the requestor's information, the CA can then issue a certificate.

There are three types of certificates that can be used to authenticate entities.

- third-party-signed

You can go to a commercial third-party certificate authority (VeriSign, GeoTrust, Thawte, DigiCert, Comodo, and so on), or a company can create their own CA and then use it to generate server and client certificates.

- site-signed

You go to the IT department at your site to obtain a certificate.

- self-signed

You serve as your own certificate authority.

After generating a digital certificate for the CA, the server, and the client (optional), you must identify for the client application one or more CAs that are to be trusted. This list is called a *trust list* or certificate chain.

A certificate chain is a sequence of certificates, where each certificate in the chain is signed by the subsequent certificate. The purpose of a certificate chain is to establish a chain of trust from a peer certificate to a trusted CA certificate. The CA vouches for the identity in the peer certificate when it signs it. If the CA is one that you trust (a copy of the CA certificate is in your root certificate directory), you can trust the signed peer certificate as well.

Mozilla Trusted Certificate Authority Certificate Bundle

SAS ships Viya with a default list of certification authority (CA) certificates from Mozilla that are known as the Mozilla trusted certificate authority (CA) certificate bundle. These are root certificates. The purpose of the root certificate is to establish a digital chain of trust. The root is the trust anchor.

These Mozilla trusted CA certificates are located in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts`. There are two files that contain the trusted list of

certificates. These are the `trustedcerts.pem` and the `trustedcerts.jks` files. These files are located on every machine in the SAS Viya deployment.

The Mozilla trusted bundle of CA Certificates is the basis for the trusted list of certificates. The `trustedcerts` files include the following certificates:

- Mozilla trusted certificate authority (CA) certificate bundle
- the Vault issued CA certificates
- the SAS generated certificates
- any certificate chain pointed to by `HTTPD_CERT_PATH` in the `vars.yml` file.

Your web browser will inherently trust all certificates that have been signed by any root that has been embedded in the browser itself or in an operating system on which it relies.

Certificates Issued by Vault (Full Deployment)

In a full deployment of SAS Viya, HashiCorp Vault is used to generate and sign root and intermediate TLS certificates. These TLS certificates are used to secure communication between various SAS Viya processes. Vault provides a point of contact for services requiring certificates needed to maintain secured communication.

Note: SAS recommends installing a full deployment, which includes the product visual interfaces and microservices.

Vault provides certificates that are part of the secured deployment. They are signed by a CA root and CA intermediate certificate created by Vault. Certificates issued by Vault and key files are placed on the CAS Controller, SAS/CONNECT Server, SAS Configuration Server (Consul), SAS Launcher Server, SAS Message Broker (RabbitMQ), and SAS Infrastructure Data Server (PostgreSQL). For more information about servers in a deployment, see [“Infrastructure Servers: Overview” on page 679](#).

Note: All the microservices have certificates signed by the Vault CA that are stored in Vault.

Certificate files and key files provided at deployment are as follows:

`vault-ca.crt`

The file `vault-ca.crt` contains the CA's certificates. It contains two certificates: The CA's Root certificate, and the CA's Intermediate certificate. This file is placed on all machines in the deployment to allow those machines to trust this machine when they connect to it.

The `vault-ca.crt` file is located at the following location: `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts/vault-ca.crt`

`sas_encrypted.crt`

A root CA certificate issued by Vault for the deployment. Sometimes referred to as the machine certificate. Each machine in a deployment has its own root CA certificate. This file is placed on all other machines in the deployment to allow those machines to trust this machine when they connect to it. This certificate file has a plaintext private key contained in file `sas_encrypted.key`. This file is located at `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs/sas_encrypted.crt`

Note: CAS uses a certificate that resides in the following directory: `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs/cas/default/sas_encrypted.crt`

`sas_encrypted.key`

The RSA private key associated with the public key. This key is embedded within file `sas_encrypted.crt`. This RSA private key is encrypted. Its decryption key is the contents of file `encryption.key`. This file is located at the following location: `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/cas/default/sas_encrypted.key`

`encryption.key`

The passphrase (or key) used to encrypt and decrypt the RSA private key in file `sas_encrypted.key`. This file is located at the following location: `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/cas/default/encryption.key`

In SAS Environment Manager, the interface for managing certificates in Vault is the SAS Secret Manager. The SAS Secret Manager is based on HashiCorp Vault 0.6.4. SAS Secret Manager uses Vault to store and generate secrets such as Transport Layer Security (TLS) certificates.

Note: A programming-only deployment does not use SAS Secret Manager. SAS Secret Manager is installed on the same machines where SAS Configuration Server resides. SAS Configuration Server must be running for SAS Secret Manager to be operational.

For information about the configuration properties, see [“Configuration Properties: Reference \(Services\)” on page 223](#).

For more information, see [“SAS Secret Manager” on page 681](#).

Default Certificates Provided for Apache HTTPD

SAS Viya uses an Apache HTTP server as a reverse proxy server to secure your environment. Apache provides default security settings using `mod_ssl` to secure the server with self-signed certificates. By default, the SAS Viya deployment installs Apache `httpd` on the machines that you designate as targets for the HTTP proxy installation unless the proxy server has already been installed.

SAS recommends that you install Apache `httpd` and configure the Apache HTTP Server to use certificates (your custom certificates) that comply with the security policies at your enterprise before you start the deployment process. This task can be performed pre-deployment or post-deployment of SAS Viya. If you configure Apache `httpd` pre-deployment and use your custom certificates, when you run the Ansible playbook to deploy SAS Viya, the custom certificates are automatically distributed to secure the server.

If you choose to use the Apache `httpd` default settings, Apache provided certificates are used to bring up the server. These settings are reasonably secure, but they are not compliant with SAS security standards. SAS recommends replacing the default certificates with custom certificates that comply with the security policies at your enterprise. If you do not add compliant certificates and instead keep the default security settings and certificates, end users will see a standard web browser warning message. SAS recommends replacing the certificates before giving end users access to SAS Viya.

Note: The default security is the only security that is available in a programming-only deployment.

When the Ansible playbook runs at installation, it inspects any existing certificates and the CA chain to determine whether they comply with SAS security requirements. If compliant certificates are found, the certificates are used without changes. If only the default `mod_ssl` is found, the playbook generates self-signed certificates and configures `mod_ssl` to use it. You can add your own certificates after the completion of the deployment process, which will require a brief outage.

The default Self-signed certificates and key files provided by default for the Apache HTTP Server are specified in the `ssl.conf` file. The location of the `ssl.conf` file is `/etc/httpd/conf.d/`. The certificate and key files are specified using the following directives. The default filenames are `localhost.crt` and `localhost.key`.

- `SSLCertificateFile /etc/pki/tls/certs/localhost.crt`
- `SSLCertificateKeyFile /etc/pki/tls/private/localhost.key`

SAS Issued Certificates

In a programming-only deployment of SAS Viya, self-signed certificates are provided for configuring TLS with CAS. These certificates are provided for machines in the deployment as follows:

- Server identity certificates are placed in the `sas_encrypted.crt` file `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs`. These are encrypted and unencrypted certificates for the CAS controller and unencrypted certificates for the CAS worker nodes.
- Private keys are placed in the `sas_encrypted.key` file in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private`. These are encrypted and unencrypted keys for the CAS controller and unencrypted keys for the CAS worker nodes.

- CA certificates are placed in the `trustedcerts.pem` file (Mozilla bundle of trusted certificates) in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts`. There are unencrypted certificates and encrypted certificates. Only encrypted certificates are provided for the CAS controller. These certificates are copied to all of the machines in the deployment.

Self-signed certificates and key files are also provided for the Apache HTTP Server. The location and filenames are associated with the following directives in the `ssl.conf` file located at `/etc/httpd/conf.d/`. Certificates are located in `localhost.crt` and in `localhost.key` files.

- `SSLCertificateFile /etc/pki/tls/certs/localhost.crt`
- `SSLCertificateKeyFile /etc/pki/tls/private/localhost.key`

Certificate File Formats

There are many file formats used to structure certificates. Here are some of them:

- encodings (also used as extensions)

PEM

Privacy Enhanced Email (`.pem`) is a container format (Base64 Encoded x.509). The `.pem` extension is used for different types of X.509v3 files, which contain ASCII (Base64) armored data prefixed with a “`—BEGIN ...`” line.

Examples are CA certificate files or an entire certificate chain. This file can contain an issued public certificate, a public key, a private key, and intermediate and root certificates.

The PEM file format is preferred by open-source software. It can have a variety of extensions (`.pem`, `.key`, `.cer`, `.cert`, and so on). For information about converting between file formats, see [“Convert Digital Certificate File Formats Using OpenSSL” on page 399](#).

DER

Distinguished Encoding Rules (`.der`) is used for binary DER encoded certificates. A PEM file is just a Base64-encoded DER file. OpenSSL can convert these to PEM. DER supports storage of a single certificate. These files can also bear the `.cer` extension or the `.crt` extension. For information about converting between file formats, see [“Convert Digital Certificate File Formats Using OpenSSL” on page 399](#).

JKS

JKS is a file format that is specific to Java. It is the Java keystore implementation. A keystore is a storage facility for cryptographic keys and certificates. Keytool is a key and certificate management utility that uses JKS as the file format of the key and certificate databases (KeyStore and TrustStores).

PKCS12 .P12

Public-Key Cryptography Standards (`.pkcs12`) is a file format that has both public and private keys in the file and all certificates in a certification path. This container file is fully encrypted with a password-based symmetric key. PFX is a predecessor to PKCS#12.

Note: The PKCS#12 format is the only file format that can be used to export a certificate and its private key.

For information about converting between file formats, see [“Convert Digital Certificate File Formats Using OpenSSL” on page 399](#).

- common extensions

CRT

The CRT extension is used for certificates. It supports storage of a single-certificate. The certificates can be encoded as binary DER or as ASCII PEM. The CER and CRT extensions are nearly synonymous.

Note: The only time CRT and CER can safely be interchanged is when the encoding type can be identical. For example, PEM-encoded CRT is the same as PEM-encoded CER.

CSR

This is a certificate signing request. Some applications can generate these for submission to certificate authorities. It includes some of the key details of the requested certificate, such as subject, organization, and state, as well as the public key of the certificate that will be signed. These are signed by the CA and a certificate is returned. The returned certificate is the public certificate. Note that this public certificate can be in a couple of formats.

KEY

The KEY extension is used both for public and private keys. The keys can be encoded as binary DER or as ASCII PEM.

SSH (Secure Shell)

SSH (Secure Shell) Overview

SSH is an abbreviation for Secure Shell. SSH is a protocol that enables users to access a remote computer via a secure connection. SSH is available through various commercial products and as freeware. OpenSSH is a free version of the SSH protocol suite of network connectivity tools.

Although SAS software does not directly support SSH functionality, you can use the tunneling feature of SSH to enable data to flow between a SAS client and a SAS server. Port forwarding is another term for tunneling. The SSH client and SSH server act as agents between the SAS client and the SAS server, tunneling information via the SAS client's port to the SAS server's port.

Linux operating systems can access an OpenSSH server on another Linux system. To access an OpenSSH server, Linux systems require OpenSSH software.

Windows systems require PuTTY software.

Currently, SAS supports the OpenSSH client and server that supports protocol level SSH-2 in Linux environments. Other third-party applications that support the SSH-2 protocol currently are untested. Therefore, SAS does not support these applications.

To understand the configuration options that are required for the OpenSSH and PuTTY clients and the OpenSSH server, it is recommended that you have a copy of the book *SSH, the Secure Shell: The Definitive Guide* by Daniel J. Barrett, Richard E. Silverman, and Robert G. Byrnes. This book is an invaluable resource when you are configuring the SSH applications, and it describes in detail topics that include public key authentication, SSH agents, and SSHD host keys.

SSH System Requirements

SAS supports SSH in the Linux operating environment.

SAS supports SSH in these operating environments:

- UNIX
- Windows
- z/OS

SSH Software Availability

OpenSSH supports SSH protocol versions 1.3, 1.5, and 2.0.

To build the OpenSSL software, refer to the following resources:

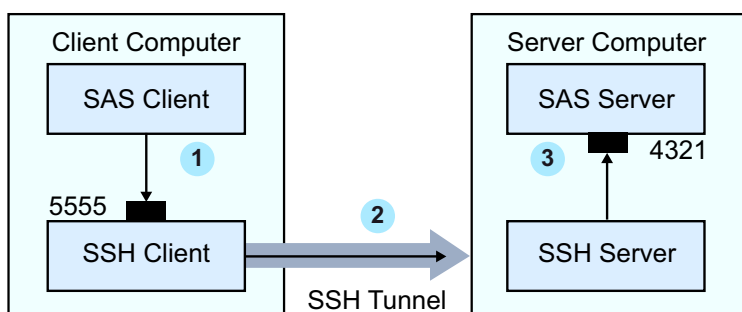
- www.openssh.com
- www.ssh.com

- [PuTTY Download Page](#)
- Barrett, Daniel J., Richard E. Silverman, and Robert G. Byrnes. 2005. *SSH, the Secure Shell: The Definitive Guide*. Sebastopol, CA: O'Really.

SSH Tunneling Process

An inbound request from a SAS client to a SAS server is shown as follows:

Figure A.2 SSH Tunneling Process



- 1 The SAS client passes its request to the SSH client's port 5555.
- 2 The SSH client forwards the SAS client's request to the SSH server via an encrypted tunnel.
- 3 The SSH server forwards the SAS client's request to the SAS server via port 4321.

Outbound, the SAS server's reply to the SAS client's request flows from the SAS server to the SSH server. The SSH server forwards the reply to the SSH client, which passes it to the SAS client.

SSH Tunneling: Process for Installation and Setup

SSH software must be installed on the client and server computers. Exact details about installing SSH software at the client and the server depend on the particular brand and version of the software that is used. See the installation instructions for your SSH software.

The process for setting up an SSH tunnel consists of the following steps:

- 1 SSH tunneling software is installed on the client and server computers. Details about tunnel configuration depend on the specific SSH product that is used. On Linux, you use OpenSSH software to access your Linux OpenSSH server.
- 2 The SSH client is started as an agent between the SAS client and the SAS server.
- 3 The components of the tunnel are set up. The components are a listen port, a destination computer, and a destination port. The SAS client accesses the listen port, which is forwarded to the destination port on the destination computer. SSH establishes an encrypted tunnel that indirectly connects the SAS client to the SAS server.

Encrypting ODS Generated PDF Files

You can use ODS to generate PDF output. When these PDF files are not password protected, any user can use Acrobat to view and edit the PDF files. You can encrypt and password-protect your PDF output files by specifying the PDFSECURITY= system option. Valid security levels for the PDFSECURITY= option are NONE or HIGH. SAS encrypts PDF documents using a 128-bit encryption algorithm. With PDFSECURITY=HIGH, at least one password must be set using the PDFPASSWORD= system option. A password is required to open a PDF file that has been generated with ODS.

Table A.35 PDF System Options

Task	System Option
Specifies whether text and graphics from PDF documents can be edited.	PDFACCESS NOPDFACCESS
Controls whether PDF documents can be assembled.	PDFASSEMBLY NOPDFASSEMBLY
Controls whether PDF document comments can be modified.	PDFCOMMENT NOPDFCOMMENT
Controls whether the contents of a PDF document can be changed.	PDFCONTENT NOPDFCONTENT
Controls whether text and graphics from a PDF document can be copied.	PDFCOPY NOPDFCOPY
Controls whether PDF forms can be filled in.	PDFFILLIN NOPDFFILLIN
Specifies the page layout for PDF documents.	PDFPAGELAYOUT=
Specifies the page viewing mode for PDF documents.	PDFPAGEVIEW=
Specifies the password to use to open a PDF document and the password used by a PDF document owner.	PDFPASSWORD=
Controls the resolution used to print the PDF document.	PDFPRINT=
Controls the printing permissions for PDF documents.	PDFSECURITY=

Encryption Algorithms

The following encryption algorithms are provided with SAS Viya:

SAS Proprietary for SAS data set encryption with passwords

is a cipher that uses parts of the passwords that are stored in the SAS data set as part of the 32-bit rolling key encoding of the data. This encryption provides a medium level of security. With the speed of today's computers, it could be subjected to a brute force attack on the 2,563,160,682,591 possible combinations of valid password values, many of which must produce the same 32-bit key.

SAS Proprietary Encryption for communications

is a cipher that provides basic fixed encoding services under all operating environments that are supported by SAS. The algorithm expands a single message to approximately one-third by using 32-bit fixed encoding. This encoding is used for passwords in configuration files, login passwords, internal account passwords, and so on.

RC2

is a block cipher that encrypts data in blocks of 64 bits. A *block cipher* is an encryption algorithm that divides a message into blocks and encrypts each block. The RC2 key size ranges from 8 to 256 bits. SAS/SECURE uses a configurable key size of 40 or 128 bits. (The NETENCRYPTKEYLEN system option is used to

configure the key length.) The RC2 algorithm expands a single message by a maximum of 8 bytes. RC2 is an algorithm developed by RSA Data Security, Inc.

RC4

is a stream cipher. A *stream cipher* is an encryption algorithm that encrypts data one byte at a time. The RC4 key size ranges from 8 to 2048 bits. SAS/SECURE uses a configurable key size of 40 or 128 bits. (The NETENCRYPTKEYLEN system option is used to configure the key length.) RC4 is an algorithm developed by RSA Data Security, Inc.

DES (Data Encryption Standard)

is a block cipher that encrypts data in blocks of 64 bits by using a 56-bit key. The algorithm expands a single message by a maximum of 8 bytes. DES was originally developed by IBM but is now published as a U.S. Government Federal Information Processing Standard (FIPS 46-3).

TripleDES

is a block cipher that encrypts data in blocks of 64 bits. TripleDES executes the DES algorithm on a data block three times in succession by using a single 56-bit key. This has the effect of encrypting the data by using a 168-bit key. TripleDES expands a single message by a maximum of 8 bytes. TripleDES is defined in the American National Standards Institute (ANSI) X9.52 specification.

AES (Advanced Encryption Standard)

is a block cipher that encrypts data in blocks of 128 bits by using a 256-bit key. AES expands a single message by a maximum of 16 bytes. Based on its DES predecessor, AES has been adopted as the encryption standard by the U.S. Government. AES is one of the most popular algorithms used in symmetric key cryptography. AES is published as a U.S. Government Federal Information Processing Standard (FIPS 197).

DSA (Digital Signature Algorithm)

The Digital Signature Algorithm (DSA) is a public-key (or asymmetric-key) cryptography algorithm. A digital signature is a mathematical scheme for demonstrating the authenticity of a digital message or document. A DSA is used to compute and verify digital signatures. Essentially, the DSA helps verify that data has not been changed after it is signed, thus providing message integrity.

In 1994, the National Institute of Standards and Technology (NIST) issued a Federal Information Processing Standard for digital signatures, known as the DSA or DSS. This was adopted as FIPS 186 in 1993.

Elliptic Curve (ECC)

is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. ECC requires smaller keys compared to non-ECC cryptography (based on plain Galois fields) to provide equivalent security. Elliptic curves are applicable for encryption, digital signatures, pseudo-random generators and other tasks.

MD5 (Message Digest)

is a series of byte-oriented algorithms that produce a 128-bit hash value from an arbitrary-length message. It is an algorithm used for hashing. It was developed by Rivest.

Note: This algorithm is not FIPS 140-2 compliant.

RSA (Rivest-Shamir-Adleman)

RSA is a public-key (or asymmetric-key) cryptography algorithm and is widely used for secure data transmission. It is used for both encryption and authentication. Encryption and decryption are carried out using two different keys, the public key and the private key. A public-key system means that the algorithm for encrypting a message is publicly known, but the algorithm to decrypt the message is only privately known. In RSA, the public key is a large number that is a product of two primes, plus a smaller number. The private key is a related number.

SHA-1 (Secure Hash Algorithm)

produces a 160-bit (20-byte) hash value. A SHA-1 hash value is typically rendered as a hexadecimal number 40 digits long. This algorithm was developed by the U.S. National Security Agency (NSA) and published in 2001 by the NIST as a U.S. Federal Information Processing Standard (FIPS) PUB 180-1.

SHA-256 (Secure Hash Algorithm)

is essentially a 256-bit block cipher algorithm that encrypts the intermediate hash value using the message block as key. This algorithm was developed by the U.S. National Security Agency (NSA) and published in 2001 by the NIST as a U.S. Federal Information Processing Standard (FIPS) PUB 180-4.

SHA-384 (Secure Hash Algorithm)

SHA384 is a truncated version of SHA512. It is essentially a 384-bit block cipher algorithm that encrypts the intermediate hash value using the message block as key. SHA-512 uses 64-bit words.

SHA-512 (Secure Hash Algorithm)

is essentially a 512-bit block cipher algorithm that encrypts the intermediate hash value using the message block as key. SHA-512 uses 64-bit words.

Reference

SAS System Options for Encryption

This section contains the SAS System options that can be used to configure encryption. These options can be specified in a number of different ways: in configuration files, in properties files, in SAS programs in the OPTIONS statement, on the SAS/CONNECT spawner command line, and in the SAS System Options window in SAS 9.

These system options are used for SAS/CONNECT and workspace servers. In a SAS Viya programming deployment, these system options are still used. In a SAS Viya deployment, TLS is configured by default. TLS is now turned on and off by [enabling TLS Ports](#).

NETENCRYPT System Option

Specifies whether encryption is required for the connection.

NETENCRYPT | NONETENCRYPT

NETENCRYPT

specifies that encryption is required.

NONETENCRYPT

specifies that encryption is not required, but is optional.

Client	Optional
Server	Optional
Valid in	Configuration file, OPTIONS statement, SAS System Options window in SAS 9, SAS invocation, SAS/CONNECT spawner command line, connectserver_usermods.sh script
Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Default	NONETENCRYPT
Operating environment	Linux
See	“NETENCRYPTALGORITHM= System Option” on page 419

The default for this option specifies that encryption is used if the NETENCRYPTALGORITHM option is set and if both the client and the server are capable of encryption. If encryption algorithms are specified but either the client or the server is incapable of encryption, then encryption is not performed.

Encryption might not be supported at the client or at the server in these situations:

- You are using a release of SAS (prior to SAS 8) that does not support encryption.
- Your site (the client or the server) does not have a security software product installed.
- You specified encryption algorithms that are incompatible in SAS sessions on the client and the server.

NETENCRYPTALGORITHM= System Option

Specifies the algorithm or algorithms to be used for encrypted client/server data transfers.

NETENCRYPTALGORITHM= algorithm | (“algorithm-1”... “algorithm-n”)

algorithm | (“algorithm-1”... “algorithm-n”)

specifies the algorithm or algorithms that can be used for encrypting data that is transferred between a client and a server across a network. These algorithms are specified on the Server.

When you specify two or more encryption algorithms, use a space or a comma to separate them, and enclose the algorithms in parentheses.

The following algorithms can be used:

- AES
- DES
- RC2
- RC4
- TripleDES
- SASProprietary
- SSL

Client	Optional
Server	Optional
Valid in	Configuration file, OPTIONS statement, SAS System Options window in SAS 9, SAS invocation, SAS/CONNECT spawner command line, connectserver_usermods.sh script
Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Alias	NETENCALG=
Default	No algorithm is defined
Operating environment	Linux
Tip	The NETENCRYPTALGORITHM= option must be specified in the server session.
See	“NETENCRYPT System Option” on page 418
Example	options netencryptalgorithm=(ssl);

Use the NETENCRYPTALGORITHM option to specify one or more encryption algorithms that you want to use to protect the data that is transferred across the network. If more than one algorithm is specified, the client session negotiates the first specified algorithm with the server session. If the client session does not support that algorithm, the second algorithm is negotiated, and so on.

If either the client session or the server session specifies the NETENCRYPT option (which makes encryption mandatory) but a common encryption algorithm cannot be negotiated, the client cannot connect to the server.

If the NETENCRYPTALGORITHM= option is specified in the server session only, then the server's values are used to negotiate the algorithm selection. If the client session supports only one of multiple algorithms that are specified in the server session, the client can connect to the server.

There is an interaction between either NETENCRYPT or NONETENCRYPT and the NETENCRYPTALGORITHM option.

Table A.36 Client/Server Connection Outcomes

Server Settings	Client Settings	Connection Outcome
NONETENCRYPT NETENCRALG= <i>alg</i>	No settings	If the client is capable of encryption, the client/server connection is encrypted. Otherwise, the connection is not encrypted.
NETENCRYPT NETENCRALG= <i>alg</i>	No settings	If the client is capable of encryption, the client/server connection is encrypted. Otherwise, the client/server connection fails.
No settings	NONETENCRYPT NETENCRALG= <i>alg</i>	A client/server connection is not encrypted.
No settings	NETENCRYPT NETENCRALG= <i>alg</i>	A client/server connection fails.
NETENCRYPT or NONETENCRYPT NETENCRALG= <i>alg-1</i>	NETENCRALG= <i>alg-2</i>	Regardless of whether NETENCRYPT or NONETENCRYPT is specified, a client/server connection fails.

NETENCRYPTKEYLEN= System Option

Specifies the key length that is used by the encryption algorithm for encrypted client/server data transfers.

NETENCRYPTKEYLEN= 0 | 40 | 128

0

specifies that the maximum key length that is supported at both the client and the server is used.

40

specifies a key length of 40 bits for the RC2 and RC4 algorithms.

128

specifies a key length of 128 bits for the RC2 and RC4 algorithms. If either the client or the server does not support 128-bit encryption, the client cannot connect to the server.

Client

Optional

Server

Optional

Valid in	Configuration file, OPTIONS statement, SAS System Options window in SAS 9, SAS invocation, SAS/CONNECT spawner command line, connectserver_usermods.sh script
Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Alias	NETENCRKEY=
Default	0
Operating environment	Linux

The NETENCRYPTKEYLEN= option supports only the RC2 and RC4 algorithms. The SAS Proprietary, DES, TripleDES, SSL, and AES algorithms are not supported.

By default, if you try to connect a computer that is capable of only a 40-bit key length to a computer that is capable of both a 40-bit and a 128-bit key length, the connection is made using the lesser key length. If both computers are capable of 128-bit key lengths, a 128-bit key length is used.

Using longer keys consumes more CPU cycles. If you do not need a high level of encryption, set NETENCRYPTKEYLEN=40 to decrease CPU usage.

SSLCACERTDIR= System Option

Specifies the location of the trusted certificate authorities (CA) found in OpenSSL format.

SSLCACERTDIR="file-path"

"file-path"

specifies the location where the public certificates for all of the trusted certificate authorities (CA) in the trust chain are filed. There is one file for each CA. Each CA certificate file must be PEM-encoded (base64). For more information, see ["Certificate File Formats" on page 413](#).

The names of the files are the value of a hash that OpenSSL generates.

Note: OpenSSL generates different hash values for each OpenSSL version. For example, OpenSSL 0.9.8 generates different hash values than does OpenSSL 1.0.2.

OpenSSL looks up the CA certificate based on the x509 hash value of the certificate. SSLCACERTDIR= requires that the certificates are located in the specified directory where the certificate names are the value of a hash that OpenSSL generates.

If you are upgrading from a version of OpenSSL that is older than 1.0.0, you need to update your certificate directory links. Starting with code base 1.0.0, SHA hashing is used instead of MD5. You can use the OpenSSL C_REHASH utility to re-create symbolic links to files named by the hash values.

You can discover the hash value for a CA and then create a link to the file named after the certificate's hash value. Note that you must add ".0" to the hash value.

```
ln -s cacert1.pem 'openssl x509 -noout -hash -in
/u/myuser/sslcerts/cacert1.pem'.0
```

If you list the CA file, you see the link between the file named after the certificate's hash value and the CA file.

```
lrwxrwxrwx 1 myuser rnd 10 Apr 7 14:42 6730c6a9.0 -> cacert1.pem
```

To verify the path of the server certificate file (cacert1.pem for our example), use the following OpenSSL command:

```
openssl verify -CApath /u/myuser/sslcerts cacert1.pem
```

Client	Optional
Server	Optional
Valid in	Configuration file, SAS invocation, SAS/CONNECT spawner start-up, connectserver_usermods.sh script
Categories	Communications: Networking and Encryption System Administration: Security
Default	The default file and location for certificates is <code>/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts/trustedcerts.pem</code> . You can point to a different file and location using the <code>SSLCALISTLOC=</code> system option or the <code>SSLCACERTDIR</code> system option. There is one trusted certificate file pointed to by the <code>SSLCALISTLOC=</code> system option. By contrast, the <code>SSLCACERTDIR</code> system option allows the customer to specify a location where multiple certificate files reside. See “SSLCALISTLOC= System Option” on page 422 .
Operating environment	Linux
Examples	<p>The <code>SSLCACERTDIR</code> system option points to the directory where the CA certificate is located. Export the environment variable on Linux hosts for the Bourne Shell:</p> <pre>export SSLCACERTDIR=/u/myuser/sslcerts/</pre> <p>Set the environment variable on Linux hosts for the C Shell directory where the CA certificates are located:</p> <pre>SETENV SSLCACERTDIR /u/myuser/sslcerts/</pre> <p>Set the environment variable at SAS invocation for Linux hosts:</p> <pre>-set "SSLCACERTDIR=/u/myuser/sslcerts/"</pre>

For Foundation Servers such as workspace servers and stored process servers, if certificates are used, SAS searches for certificates in the following order:

For Foundation Servers such as workspace servers and stored process servers, if certificates are used, SAS searches for certificates in the following order:

- 1 SAS looks for SAS system option `SSLCALISTLOC=` to find the file `trustedcerts.pem`.
- 2 SAS looks for the `SSLCALISTLOC=` environment variable to find the file `trustedcerts.pem`.
- 3 If the `SSLCALISTLOC=` system option or environment variable is not used, the `trustedcerts.pem` file located in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts` is used as the default.
- 4 If `trustedcerts.pem` exists, and the `SSL_CERT_DIR` and `SSLCACERTDIR` environment variables are set, SAS checks `trustedcerts.pem` first before it searches the directory.
- 5 If `trustedcerts.pem` does not exist, but the certificates are in the directory defined by `SSL_CERT_DIR` or `SSLCACERTDIR`, then SAS ignores `SSLCALISTLOC=`.
- 6 If `trustedcerts.pem` does not exist, and the `SSL_CERT_DIR` and `SSLCACERTDIR` environment variables are not set, SAS reports an error.

Note: A trusted CA certificate is required at the client in order to validate a server's digital certificate. The trusted CA certificate must be from the CA that signed the server certificate.

SSLCALISTLOC= System Option

Specifies the location of the public certificate(s) for trusted certificate authorities (CA).

SSLCALISTLOC="file-path"**"file-path"**

specifies the location of a single file that contains the public certificate(s) for all of the trusted certificate authorities (CA) in the trust chain.

Note: Specify this option on the client. Optionally, specify this option on the server.

Client	Optional
Server	Optional
Valid in	Configuration file, OPTIONS statement, SAS System Options window in SAS 9, SAS invocation, SAS/CONNECT spawner command line, connectserver_usermods.sh script
Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Operating environment	Linux
Notes	<p>If the SSLCALISTLOC= system option is not specified, SAS defaults to a file named <code>trustedcerts.pem</code> located in</p> <pre>/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts</pre> <p>The <code>trustedcerts.pem</code> file contains the list of trusted CA certificates provided by SAS at installation.</p> <p>If you use this option, it must be specified on the client, but does not have to also be specified on the server.</p>

The SSLCALISTLOC= system option specifies the location of a single file that contains the public certificate(s) for all of the trusted certificate authorities (CA) in the trust list. The CA file must be PEM-encoded (base64).

The location of the trusted certificate file specified by the SSLCALISTLOC= system option or SSLCACERTDIR= system option or the `trustedcerts.pem` file is needed on the spawner to verify the certificate from the SAS/CONNECT server.

The default path set for the SSLCALISTLOC= system option on the workspace server is `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts/trustedcerts.pem`. By default, the `trustedcerts.pem` file contains a managed set of trusted root certificates (Mozilla bundle of certificates and others) provided at SAS installation.

Note: The SSLCACERTDIR= system option can be used instead of using the SSLCALISTLOC= system option. SSLCACERTDIR= points to a directory that contains all of the public certificate file(s) of all CA(s) in the trust list. One file exists for each CA in the trust list. For more information, see ["SSLCACERTDIR= System Option" on page 421](#).

For Foundation Servers such as workspace servers and stored process servers, if certificates are used, SAS searches for certificates in the following order:

- 1 SAS looks for SAS system option SSLCALISTLOC= to find the file `trustedcerts.pem`.
- 2 SAS looks for the SSLCALISTLOC= environment variable to find the file `trustedcerts.pem`.
- 3 If the SSLCALISTLOC= system option or environment variable is not used, the `trustedcerts.pem` file located in `/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts` is used as the default.

- 4 If `trustedcerts.pem` exists, and the `SSL_CERT_DIR` and `SSLCACERTDIR` environment variables are set, SAS checks `trustedcerts.pem` first before it searches the directory.
- 5 If `trustedcerts.pem` does not exist, but the certificates are in the directory defined by `SSL_CERT_DIR` or `SSLCACERTDIR`, then SAS ignores `SSLCALISTLOC=`.
- 6 If `trustedcerts.pem` does not exist, and the `SSL_CERT_DIR` and `SSLCACERTDIR` environment variables are not set, SAS reports an error.

Note: A trusted CA certificate is required at the client in order to validate a server's digital certificate. The trusted CA certificate must be from the CA that signed the server certificate. The `SSLCALISTLOC=` option is required at the server only if the `SSLCLIENTAUTH` option is also specified at the server.

Note: Unless the `SSLCACERTDIR=` system option is set or the default `trustedcerts.pem` file is used, the `SSLCALISTLOC=` system option is needed on the spawner to verify the certificate from the SAS/CONNECT server.

SSLCACERTDATA= System Option

Specifies the name of the issuer of the digital certificate that TLS should use.

SSLCACERTDATA="encoded-string"

"encoded-string"

specifies the base64-encoded x509 text that represents a single certificate authority (CA) certificate. This string is in PEM format. The text string starts with the line "-----BEGIN CERTIFICATE-----" and ends with the line "-----END CERTIFICATE-----".

This option provides a way to programmatically specify a CA certificate rather than having to point to a file that contains the certificate information. The certificate must be PEM-encoded (base64) format.

Here is an example of how you might use the `SSLCACERTDATA=` system option to specify a certificate.

```
data _null_;
  length certInfo $3200.;
  input txt $67.;
  retain certInfo;

  if _N_ = 1 then
    certInfo=txt;
  else
    certInfo=catx('0a'x, certInfo, txt);
  call symput('certInfo', trim(left(certInfo)));
  datalines;
-----BEGIN CERTIFICATE-----
MIICbzCCAafagAwIBAgIJAP7q5/tk7+1aMAoGCCqGSM49BAMCMHYxCzAJBgNVBAYT
AlVTMQswCQYDVQQIDAJQJzENMAsgA1UEBwwEQ2FyeTEWMBQGA1UECgwNU0FTIElu
c3RpdHV0ZTEEMMAoGA1UECwwDSURCMSUwIwYDVQQDDDBxkZW1vUm9vdENBLUVDRFNB
LVAzODQtU0hBMjU2MB4XDTE2MTEwNDE4MDMzMVoXDTE2MTEwMjE4MDMzMVowdjlEL
MAkGA1UEBhMCVVMxMzAJBgNVBAGMAk5DMQ0wCwYDVQQHDARDYXJ5J5MRyWFAyDVQQK
DA1TQVMgSW5zdG10dXRlMQwwCgYDVQQQLDANJREIeJTAjBgNVBAMMHGR1bW9Sb290
Q0EtRUNEU0EtUDM4NC1TSEEyNTYwdjAQBgcqhkJOPQIBBgUrgQQAIGNiAAQghfjE
5iiiPQtB/Ors/GeNuLRXWnUhqnPWw4X0veIQT5rXFWZmiwReIjaYt9KChhmFkPno
cQ1m3HpdVnP86cPLPpLSvcAG/d06o2W2SakiOWa1cA1UKsRhy/kUMnTSGJSjUDBO
MB0GA1UdDgQWBBSXhRRVQTNHpe1A9NsdUa+Y/IxhTTAfBgNVHSMEGDAWgBSXhRRV
QTNHpe1A9NsdUa+Y/IxhTTAMBgNVHRMEBTADAQH/MAoGCCqGSM49BAMCA2cAMGQC
MFJf5/2+eRSwCxrOyVjgyI4Teiofggrji5StKyQzHhDnXPljdYRss0WxxhbdBcxo
8wIwDjX8Yx611Y52U/h0q8ZkuJWu0gJ8ZmrOVttkUBYUUD1Cer6pd14gQd6mUz
oXrB
-----END CERTIFICATE-----
```



```

;
run;

options SSLCACERTDATA="&certInfo";

```

SSLCERTLOC= System Option

Specifies the location of the digital certificate for the machine's public key. This is used for authentication.

SSLCERTLOC="file-path"

"file-path"

specifies the location of a file that contains a digital certificate for the machine's public key. The certificate must be PEM-encoded (base64). This is used by servers to send to clients for authentication.

Client	Optional
Server	Optional
Valid in	Configuration file, OPTIONS statement, SAS System Options window in SAS 9, SAS invocation, SAS/CONNECT spawner command line, connectserver_usermods.sh script
Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Operating environment	Linux
Note	If you use this option, it must be specified on the server, but does not have to also be specified on the client.

The SSLCERTLOC= option is required for a server. It is required at the client only if the SSLCLIENTAUTH option is specified at the server. In order for a TLS connection to succeed, the SAS/CONNECT server needs to be started with the -SSLCERTLOC= and -SSLPVTKEYLOC= system options set in the SAS/CONNECT spawner. Alternatively, the -SSLPKCS12LOC= system option can be used.

In SAS Viya, set the SSL options on the spawner and the server in the connectserver_usermods.sh file (`/opt/sas/viya/config/etc/connectserver/default`) and in the connect_usermods.sh file (`/opt/sas/viya/config/etc/connect/default`). For configuration information, see ["Sign On to a SAS/CONNECT Spawner Using TLS" on page 382](#).

SSLCIPHERLIST= System Option

Specifies the ciphers that can be used on Linux for OpenSSL.

SSLCIPHERLIST=openssl_cipher_list

openssl-cipher-list

The SSLCIPHERLIST= system option specifies the ciphers that can be used on Linux for OpenSSL. Refer to the OpenSSL Ciphers document to see how to format the *openssl-cipher-list* and for a complete list of the ciphers that work with your TLS version. The OpenSSL Ciphers information can be found at <https://www.openssl.org/docs/man1.1.0/apps/ciphers.html>.

Note: SAS does not support CAMELLIA, IDEA, MD2, and RC5 ciphers.

Note: The protocol and cipher information for the actual connection can be seen by setting `dumpCurrentCipherInfo` at the SAS DEBUG level.

Note: If you set a minimum protocol that does not allow some ciphers, you might get an error.

Client	Optional
Server	Optional
Valid in	Configuration file, command line
Categories	Communications: Networking and Encryption System Administration: Security
Restriction	If the SSLMODE= option is set, this option is ignored.
Operating environment	Linux
Notes	This option can also be specified as an environment variable. This system option must be set before TLS is loaded. It cannot be changed after TLS is loaded. You must set the environment variable before the SAS/CONNECT spawner is started and before SAS is started on the client.
Example	Specify the system option: -SSLCIPHERLISTS= HIGH

SSLCLIENTAUTH System Option

Specifies whether a server should perform client authentication.

SSLCLIENTAUTH | NOSSLCLIENTAUTH

SSLCLIENTAUTH

specifies that the server should perform client authentication. Server authentication is always performed, but the SSLCLIENTAUTH option enables a user to control client authentication. This option is valid only when used on a server.

TIP If you enable client authentication, a certificate for each client is needed.

NOSSLCLIENTAUTH

specifies that the server should not perform client authentication.

Default NOSSLCLIENTAUTH is the default.

Server	Optional
Valid in	Configuration file, OPTIONS statement, SAS System Options window in SAS 9, SAS invocation, SAS/CONNECT spawner command line, connectserver_usermods.sh script
Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Operating environment	Linux
Note	If you use this option, it is specified on the server.

SSLCRLCHECK System Option

Specifies whether a Certificate Revocation List (CRL) is checked when a digital certificate is validated.

SSLCRLCHECK | NOSSLCRLCHECK**SSLCRLCHECK**

specifies that CRLs are checked when digital certificates are validated.

NOSSLCRLCHECK

specifies that CRLs are not checked when digital certificates are validated.

Client	Optional
Server	Optional
Valid in	Configuration file, OPTIONS statement, SAS System Options window in SAS 9, SAS invocation, SAS/CONNECT spawner command line, connectserver_usermods.sh script
Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Operating environment	Linux
Note	If you use this option, it can be specified on the client and server.
See	“SSLCRLLOC= System Option” on page 427

A certificate revocation list (CRL) is published by a certificate authority (CA) and contains a list of revoked digital certificates. The list contains only the revoked digital certificates that were issued by a specific CA.

The SSLCRLCHECK option is required at the server only if the SSLCLIENTAUTH option is also specified at the server. Because clients check server digital certificates, this option is relevant for the client.

SSLCRLLOC= System Option

Specifies the location of a certificate revocation list (CRL).

SSLCRLLOC=“file-path”**“file-path”**

specifies the location of a file that contains a certificate revocation list (CRL).

Client	Optional
Server	Optional
Valid in	Configuration file, OPTIONS statement, SAS System Options window in SAS 9, SAS invocation, SAS/CONNECT spawner command line, connectserver_usermods.sh script
Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Operating environment	Linux

Notes	If you use this option, it can be specified on the client and server.
	The SSLCRLLOC= option is required only when the SSLCRLCHECK option is specified.
See	“SSLCRLCHECK System Option” on page 427

SSLMINPROTOCOL= System Option

Specifies the minimum TLS or SSL protocol that can be negotiated when using OpenSSL.

SSLMINPROTOCOL=*protocol*

protocol

specifies the minimum TLS or SSL protocol version that is negotiated between Linux servers when using OpenSSL. SAS Viya supports specifying TLS1.2 and TLSv1.2. The following other values can be specified, but are less secure: SSL3, SSLV3, TLS, TLS1, TLSV1, TLS1.0, TLSV1.0, TLS1.1, and TLSV1.1.

CAUTION! TLS versions 1.0 and 1.1 are insecure. It is highly recommended that you use TLS 1.2 or later.

Note: A message is written to the SAS log when an invalid value is specified.

During the first TLS handshake attempt, the highest supported protocol version is offered. If this handshake fails, earlier protocol versions are offered instead. TLS1.2 is the default minimum OpenSSL protocol. By default, the SSLMODE= option is set to SSLMODESP800131A, which uses TLS 1.2 to negotiate between client and servers. You can specify an earlier fallback value, but it is not recommended.

See the [SAS SAS Statement Regarding OpenSSL Security Advisories](#) for the most current information about the versions of OpenSSL used in SAS products and about the advisories under consideration.

Client	Optional
Server	Optional
Valid in	Configuration file, command line, SAS/CONNECT spawner start-up if this option is used as an environment variable
Categories	Communications: Networking and Encryption System Administration: Security
Default	TLS 1.2.
Restriction	If the SSLMODE= option is set, this option is ignored.
Operating environment	Linux
Notes	This option can also be specified as an environment variable. This environment variable must be set before TLS or SSL are loaded. It cannot be changed after TLS or SSL are loaded. You must set the environment variable before the SAS/CONNECT spawner is started and before SAS is started on the client.
Example	Specify the system option as follows: -SSMINPROTOCOL="TLS1.2"

SSLMODE= System Option

Sets the allowed TLS version and cipher suites to be used for TLS.

SSLMODE=*ssl-mode*

ssl-mode

SSLMODESUITEB128

is the mode of operation that uses the cipher-suites specified in the NIST Suite B Cryptography using 128 AES encryption.

SSLMODESUITEB192

is the mode of operation that uses the cipher-suites specified in the NIST Suite B Cryptography using 192 AES encryption.

SSLMODESP800131A

is the DEFAULT configuration mode for TLS communication.

SSLMODEDEPRECATED

is the mode of operation that uses the cipher-suites specified in the NIST Special Publication 800-131A.

When system option SSLMODE= is set, system option SSLMINPROTOCOL= is ignored. If SSLMODE= is not set, SAS checks the SSLMINPROTOCOL= system option and uses the protocol set. If neither system option is set, SAS uses the default cipher mode SSLMODESP800131A.

Client	Optional
Server	Optional
Valid in	Configuration file, command line, SAS/CONNECT spawner start-up if this option is used as an environment variable, connectserver_usermods.sh script
Categories	Communications: Networking and Encryption System Administration: Security
Default	SSLMODESP800131A
Restriction	If the SSLMODE= option is set, this option is ignored.
Interaction	When system option SSLMODE= is set, system option SSLMINPROTOCOL= is ignored.
Operating environment	Linux
See	For a list of ciphers that are supported for each of the modes that can be specified for the SSLMODE= system option, see SSLMODE= System Option Supported Ciphers .
Example	Specify the system option as follows: <pre>-sslmode SSLMODESP800131A</pre>

SAS uses the National Institute of Standards and Technology (NIST) Special Publication 800-131A (SP800-131A) as the minimum compliance standard for TLS and to extend the FIPS standards. TLS version 1.2 is the default version of TLS that SAS supports. However, SAS does provide the ability to specify less secure TLS 1.1 if needed (SSLMODEDEPRECATED). For details of SP800-131A, see [NIST Special Publication 800-131A, Revision 1](#).

Suite B cryptography allows TLS client and server applications to specify a profile compliant with Suite B Cryptography as defined in [RFC 5430: Suite B Profile for Transport Layer Security \(TLS\)](#). Suite B cryptography specifies the cryptographic algorithms that can be used in a “Suite B Compliant” TLS V1.2 session. Suite B requires the key establishment and authentication algorithms that are used in TLS V1.2 sessions to be based on Elliptic Curve Cryptography, and the encryption algorithm to be AES.

For a list of ciphers that are supported for each of the modes that can be specified for the SSLMODE= system option, see [SSLMODE= System Option Supported Ciphers](#).

SSLPKCS12LOC= System Option

Specifies the location of the PKCS #12 encoding package file.

SSLPKCS12LOC=“file-path”

“file-path”

specifies the location of the PKCS #12 DER encoding package file that contains the certificate and the private key.

Client	Optional
Server	Optional
Valid in	Configuration file, OPTIONS statement, SAS System Options window in SAS 9, SAS invocation, SAS/CONNECT spawner command line, connectserver_usermods.sh script
Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Operating environment	Linux
Notes	If you use this option, it can be specified on the client and server. You must specify both the -SSLPKCS12LOC option and the -SSLPKCS12PASS option together.
See	“SSLPKCS12PASS= System Option” on page 430

If the SSLPKCS12LOC= option is specified, the PKCS #12 DER encoding package must contain both the certificate and private key. The SSLCERTLOC= and SSLPVTKEYLOC= options are ignored.

You must specify both the -SSLPKCS12LOC option and the -SSLPKCS12PASS option in order for the SAS/CONNECT server to access the appropriate server scripts. In SAS Viya, set the SSL options on the spawner and the server in the connectserver_usermods.sh file (`/opt/sas/viya/config/etc/connectserver/default`) and in the connect_usermods.sh file (`/opt/sas/viya/config/etc/connect/default`). For configuration information, see [“Sign On to a SAS/CONNECT Spawner Using TLS” on page 382](#).

SSLPKCS12PASS= System Option

Specifies the password that TLS requires for decrypting the private key.

SSLPKCS12PASS=password

password

specifies the password that TLS requires in order to decrypt the PKCS #12 DER encoding package file. The PKCS #12 DER encoding package is stored in the file that is specified by using the SSLPKCS12LOC= option.

Client	Optional
Server	Optional
Valid in	Configuration file, OPTIONS statement, SAS System Options window in SAS 9, SAS invocation, SAS/CONNECT spawner command line
Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Operating environment	Linux
Notes	If you use this option, it can be specified on the client and server. You must specify both the -SSLPKCS12LOC option and the -SSLPKCS12PASS option together.
See	“SSLPKCS12LOC= System Option” on page 430

The SSLPKCS12PASS= option is required only when the PKCS #12 DER encoding package is encrypted.

You must specify both the -SSLPKCS12LOC option and the -SSLPKCS12PASS option in order for the SAS/CONNECT server to access the appropriate server scripts. In SAS Viya, set the SSL options on the spawner and the server in the connectserver_usermods.sh file (`/opt/sas/viya/config/etc/connectserver/default`) and in the connect_usermods.sh file (`/opt/sas/viya/config/etc/connect/default`). For configuration information, see [“Sign On to a SAS/CONNECT Spawner Using TLS” on page 382](#).

SSLPVTKEYLOC= System Option

Specifies the location of the private key that corresponds to the digital certificate.

SSLPVTKEYLOC=“file-path”

“file-path”

specifies the location of the file that contains the private key that corresponds to the digital certificate that was specified by using the SSLCERTLOC= option.

Client	Optional
Server	Optional
Valid in	Configuration file, OPTIONS statement, SAS System Options window in SAS 9, SAS invocation, SAS/CONNECT spawner command line, connectserver_usermods.sh script
Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Operating environment	Linux
Notes	If you use this option, it can be specified on the client and server. You must specify the -SSLCERTLOC option if you specify the -SSLPVTKEYLOC option. -SSLPVTKEYPASS is required only when the private key is encrypted.

See [“SSLCERTLOC= System Option” on page 425](#) and [“SSLPVTKEYPASS= System Option” on page 432](#).

The SSLPVTKEYLOC= option is required at the server only if the SSLCERTLOC= option is also specified at the server.

The key must be PEM-encoded (base64). For more information, see [“Certificate File Formats” on page 413](#).

You must specify both the -SSLCERTLOC option and the -SSLPVTKEYLOC option in order for the SAS/CONNECT server to access the appropriate server scripts. In SAS Viya, set the SSL options on the spawner and the server in the connectserver_usermods.sh file (`/opt/sas/viya/config/etc/connectserver/default`) and in the connect_usermods.sh file (`/opt/sas/viya/config/etc/connect/default`). For configuration information, see [“Sign On to a SAS/CONNECT Spawner Using TLS” on page 382](#).

SSLPVTKEYPASS= System Option

Specifies the password that TLS requires for decrypting the private key.

SSLPVTKEYPASS=“password”

“password”

specifies the password that TLS requires in order to decrypt the private key. The private key is stored in the file that is specified by using the SSLPVTKEYLOC= option.

Client	Optional
Server	Optional
Valid in	Configuration file, OPTIONS statement, SAS System Options window in SAS 9, SAS invocation, SAS/CONNECT spawner command line, connectserver_usermods.sh script
Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Operating environment	Linux
Notes	If you use this option, it can be specified on the client and server. You must specify the -SSLCERTLOC= option if you specify the -SSLPVTKEYLOC= option. -SSLPVTKEYPASS= is required only when the private key is encrypted.
See	“SSLCERTLOC= System Option” on page 425 and “SSLPVTKEYPASS= System Option” on page 432 .

The SSLPVTKEYPASS= option is required only when the private key is encrypted. OpenSSL performs key encryption.

Note: No SAS system option is available to encrypt private keys.

SSLREQCERT= System Option

Specifies what checks to perform on server certificates in a TLS session.

SSLPVTKEYLOC=ALLOW | DEMAND | NEVER | TRY**ALLOW**

specifies that the client requests a server certificate, but the session proceeds normally even if no certificate is provided or an invalid certificate is provided.

DEMAND

specifies that a server certificate is requested, and if no valid certificate is provided, the session terminates. DEMAND is the default setting.

NEVER

specifies that the authentication server does not ask for a certificate.

TRY

specifies that the client requests a server certificate, and if no certificate is provided, the session proceeds normally. If an invalid certificate is provided, the session terminates.

If you do not add the SSLREQCERT= option to your configuration file, then the default value is DEMAND. If you specify SSLREQCERT=, then the value of SSLREQCERT= applies to all of your authentication providers.

Note: To ensure proper security, SSLREQCERT=DEMAND should be specified.

Client	Optional
Server	Optional
Valid in	Configuration file, SAS invocation, SAS/CONNECT spawner command line, connectserver_usermods.sh script
Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Operating environment	Linux
Example	export SSLREQCERT=DEMAND

SSLSNIHOSTNAME= System Option

Enables the client to specify the Server Name Indication (SNI) in the TLS handshake that identifies the server name that it is trying to connect to.

SSLSNIHOSTNAME= "hostname"

specifies the host name that is used for the Server Name Indication (SNI) TLS extension. If it is not specified, the target host name is used. The client uses SNI in the first message of the TLS handshake (connection setup) to identify the server name that it is trying to connect to.

The client uses SNI in the TLS handshake to identify the server name that it is trying to connect to. When making a TLS connection, the client requests a digital certificate from the web server. After the server sends the certificate, the client examines it and compares the name that it was trying to connect to with the name or names included in the certificate. If a match is found, the connection proceeds as normal.

Client	Optional
Server	Optional
Valid in	Configuration file, SAS invocation, SAS/CONNECT spawner start-up if this option is used as an environment variable, connectserver_usermods.sh script

Category	Communications: Networking and Encryption
PROC OPTIONS GROUP=	Communications
Default	The default is the name of the host being contacted.
Operating environment	Linux
Notes	This option can also be specified as an environment variable. The TLS SNI extension is always sent to the web server.
Example	Specify the system option as follows: <code>-SSLSNIHOSTNAME="www.example.org"</code>

SAS Environment Variables for Encryption

Overview of Environment Variables

Linux environment variables are variables that apply to both the current shell and to any subshells that it creates. The way in which you define an environment variable depends on the shell that you are running. For more information, see [Defining Environment Variables in UNIX Environments](#).

SSL_USE_SNI Environment Variable

Disables the use of Server Name Indication (SNI) in the TLS handshake for the client.

SSL_USE_SNI

SSL_USE_SNI

Linux clients and servers support TLS Server Name Indication (SNI). The client uses SNI in the first message of the TLS handshake (connection setup) to identify the server name that it is trying to connect to.

The client uses SNI in the TLS handshake to identify the server name that it is trying to connect to. When making a TLS connection, the client requests a digital certificate from the web server. After the server sends the certificate, the client examines it and compares the name that it was trying to connect to with the name or names included in the certificate. If a match is found, the connection proceeds as normal.

Client	Optional
Server	Optional
Valid in	SAS invocation, configuration file
Categories	Communications: Networking and Encryption System Administration: Security
Default	By default, the TLS SNI extension is sent as part of the TLS handshake.
Restriction	System option SSLSNIHOSTNAME= is used to specify the Server Name Indication (SNI) that identifies the server name that it is trying to connect to. This environment variable is now used to turn off SNI which is sent by default.

Operating environment

Linux

Examples

Export the environment variable on Linux hosts for the Bourne Shell:

```
export SSL_USE_SNI=1
```

Set the environment variable at SAS invocation for Linux hosts:

```
SETENV SSL_USE_SNI
```

CAS TLS Environment Variables

CAS server options are stored in configuration files. During deployment, `casconfig_deployment.lua` is created in the `/opt/sas/viya/config/etc/cas/default` directory. This file contains CAS configuration settings that are created during deployment by Ansible from `vars.yml`.

The `casconfig_usermods.lua` file is now used by the SAS administrator to add CAS configuration options. During redeployments or upgrades, `casconfig_usermods.lua` is not modified. Therefore, to preserve your CAS configurations, always use `casconfig_usermods.lua` for changes.

When you start the server with the `service sas-viya-cascontroller-default start` command, the `casconfig_deployment.lua`, `casconfig.lua`, and the `casconfig_usermods.lua` configuration files are processed. For more information, see [“SAS Cloud Analytic Services: Reference” on page 627](#).

These are the configuration options that can be used for configuring TLS on CAS servers and clients.

env.CAS_CALISTLOC=<'path/CA-list-file'>

Specifies the location of a single file that contains the public certificate(s) for all of the trusted certificate authorities (CA) in the trust list. This is the CA list location when CAS is acting as a client

Client	Optional
Valid in	Server configuration file, cas.settings file, cas configuration files, and operating system command line
Used by	CAS Server
Category	Security
Requirement	The certificate files and the key files being referenced by these environment variables must be PEM-encoded (Base64 ASCII).
Example	<code>env.CAS_CALISTLOC='/opt/sas/viya/config/etc/SASSecurityCertificateFramework/cacerts/trustedcerts.pem'</code>

env.CAS_CERTLOC=<'path/certificate-file'>

Specifies the path and filename of the file that contains the PEM-formatted certificate to be used for TLS communications.

This environment variable can also point to a certificate chain that starts with the server identity certificate and includes one or more intermediate CA certificates in the order in which they were signed.

Valid in	Server configuration file, cas.settings file, cas configuration files, and operating system command line
Used by	CAS REST API
Category	Security

Requirement Use with [env.CAS_PVTKEYLOC](#) on page 441 and [env.CAS_PVTKEYPASS](#) on page 441.

Example

```
env.CAS_CERTLOC='/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tls/certs/cas/shared/
default/sas_encrypted.crt'
```

env.CAS_CLIENT_SSL_CA_LIST=<'path/certificates-file'>

Specifies the path and filename of the file that contains the list of trusted certificate authorities (CAs). This environment variable can be used by the CAS server or by the client connecting to the CAS server. For the server, this environment variable points to the trust list used to accept connections to the server. For the client, this environment variable points to the trust list that the client uses to connect to the server.

Client Optional

Server Optional

Valid in Server configuration file, cas.settings file, CAS Lua configuration files, and operating system command line

Used by CAS client, Lua client, Python Client, CAS server, SAS 9.4 client

Category Security

Example

```
export CAS_CLIENT_SSL_CA_LIST='/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/cacerts/trustedcerts.pem'
export CAS_CLIENT_SSL_CA_LIST='/opt/sas/viya/
config/etc/SASSecurityCertificateFramework/cacerts/vault-ca.crt
export <SASHome>/SASSecurityCertificateFramework/1.1/
cacerts/trustedcerts.pem
```

env.CAS_CLIENT_SSL_CERT=<'path/certificate-file'>

Specifies the path and filename of the file that contains the certificate that the client uses to connect to the server for TLS communications. This environment variable is used when accepting connections to the CAS server.

This environment variable can also point to a certificate chain that starts with the server identity certificate and includes one or more intermediate CA certificates in order that they are signed.

Server Optional

Valid in Server configuration file, cas.settings file, and operating system command line

Used by CAS server

Category Security

Notes Environment Variables CAS_CLIENT_SSL_KEY=, CAS_CLIENT_SSL_KEYPW=, CAS_CLIENT_SSL_CERT= are specified together.

The contents of this file are not confidential.

Example

```
env.CAS_CLIENT_SSL_CERT='/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tls/certs/sas_encrypted.crt '
env.CAS_CLIENT_SSL_CERT='/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/tls/certs/cas/default/sas_encrypted.crt '
```

env.CAS_CLIENT_SSL_CLIENT_CERT=<'path/certificate-file'>

Specifies the path and filename of the file that contains the certificate that the client uses to connect to the server for TLS communications.

This environment variable is specified when the client presents a certificate to the server. In most configurations, only the server presents a certificate to the client.

This environment variable can also point to a certificate chain that starts with the server identity certificate and includes one or more intermediate CA certificates in order that they are signed.

Client Optional

Valid in Server configuration file, cas.settings file, and operating system command line

Used by CAS client

Category Security

Notes The contents of this file are not confidential.

Environment Variables CAS_CLIENT_SSL_CLIENT_KEY=, CAS_CLIENT_SSL_CLIENT_KEYPW=, CAS_CLIENT_SSL_CLIENT_CERT= are specified together.

Example

```
env.CAS_CLIENT_SSL_CLIENT_CERT='/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs/CASClient.crt'
env.CAS_CLIENT_SSL_CLIENT_CERT='/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs/cas/default/CASClient.crt'
```

env.CAS_CLIENT_SSL_CLIENT_KEY=<'path/key-file'>

Specifies the path and filename of the file that contains the private key for the client to use to connect to the server for TLS communications.

This environment variable is specified when the client presents a certificate to the server. In most configurations, only the server presents a certificate to the client.

Client Optional

Valid in Server configuration file, cas.settings file, and operating system command line

Used by CAS client

Category Security

Note Environment Variables CAS_CLIENT_SSL_CLIENT_KEY=, CAS_CLIENT_SSL_CLIENT_KEYPW=, CAS_CLIENT_SSL_CLIENT_CERT= are specified together.

Tip The contents of this file should be kept confidential.

Example

```
env.CAS_CLIENT_SSL_CLIENT_KEY='/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/CASClient.key'
env.CAS_CLIENT_SSL_CLIENT_KEY='/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/cas/default/CASClient.key'
```

env.CAS_CLIENT_SSL_CLIENT_KEYPW=<'password'>

Specifies the password for the client's private key. The password in this variable should match the password used to generate the private key file specified by the CAS_CLIENT_SSL_CLIENT_KEY= environment variable.

Note: This password is not encoded.

This password should be set in a Lua configuration file that is readable only by the CAS service account.

Client	Optional
Valid in	Server configuration file, cas.settings file, and operating system command line
Used by	CAS client
Category	Security
Note	Environment Variables CAS_CLIENT_SSL_CLIENT_KEY=, CAS_CLIENT_SSL_CLIENT_KEYPW=, CAS_CLIENT_SSL_CLIENT_CERT= are specified together.
Example	<code>env.CAS_CLIENT_SSL_CLIENT_KEYPW='encryptedpassword'</code>

env.CAS_CLIENT_SSL_KEY=<path/key-file>

Specifies the path and filename of the file that contains the private key for the client to be used for TLS communications. This key is used when accepting connections to the server.

Server	Optional
Valid in	Server configuration file, cas.settings file, and operating system command line
Used by	CAS server
Category	Security
Note	Environment Variables CAS_CLIENT_SSL_KEY=, CAS_CLIENT_SSL_KEYPW=, CAS_CLIENT_SSL_CERT= are specified together.
Tip	The contents of this file should be kept confidential.
Example	<code>env.CAS_CLIENT_SSL_KEY='/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/sas_encrypted.key'</code> <code>env.CAS_CLIENT_SSL_KEY='/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/cas/default/sas_encrypted.key'</code>

env.CAS_CLIENT_SSL_KEYPW=<'password'>

Specifies the password for the server's private key. The password in this variable should match the password used to generate the private key file specified by the CAS_CLIENT_SSL_KEY= environment variable.

Note: This password is not encoded.

Server	Optional
Valid in	Server configuration file, cas.settings file, and operating system command line
Used by	CAS server
Category	Security
Note	Environment Variables CAS_CLIENT_SSL_KEY=, CAS_CLIENT_SSL_KEYPW=, CAS_CLIENT_SSL_CERT= are specified together.
Example	<code>env.CAS_CLIENT_SSL_KEYPW='password'</code>

env.CAS_CLIENT_SSL_KEYPWLOC=<'path/certificate-file'>

Specifies the location of the password file for the server's private key. The password in this variable should match the password used to generate the private key file specified by the CAS_CLIENT_SSL_KEY= environment variable.

Server Optional

Valid in Server configuration file, cas.settings file, and operating system command line

Used by CAS server

Category Security

Note Environment Variables CAS_CLIENT_SSL_KEY= and CAS_CLIENT_SSL_CERT= are specified together.

Example

```
env.CAS_CLIENT_SSL_KEYPWLOC='/opt/sas/viya/
SASSecurityCertificateFramework/private/cas/default/encryption.key'

env.CAS_CLIENT_SSL_KEYPWLOC='/opt/sas/viya/
SASSecurityCertificateFramework/private/encryption.key'
```

env.CAS_CLIENT_SSL_REQUIRED=<'true' | 'false' >

Determines whether encryption is used between the client and the server.

Valid in Server configuration file, cas.settings file, CAS Lua config files, and operating system command line

Used by CAS server

Category Security

Example env.CAS_CLIENT_SSL_REQUIRED='true'

env.CAS_INTERNODE_DATA_SSL=<true | false>

Enables encryption for the analytics cluster when set to `true`. This value must be the same on every node in the cluster.

Valid in Server configuration file, cas.settings file, and operating system command line

Category Security

Example env.CAS_INTERNODE_DATA_SSL=true

env.CAS_INTERNODE_SSL_CA_LIST=<'path/keystore'>

Specifies the path and filename of the file that contains the list of trusted certificate authorities (CAs). This setting is likely to be the same for all nodes in the grid.

Valid in Server configuration file, cas.settings file, and operating system command line

Category Security

Example

```
env.CAS_INTERNODE_SSL_CA_LIST="/opt/sas/viya/config
/etc/SASSecurityCertificateFramework/cacerts/trustedcerts.pem"
```

env.CAS_INTERNODE_SSL_CERT=<'path/certificate-file'>

Specifies the path and filename of the file that contains the certificate to be used for TLS communications for the certificate specific to the node being configured.

This environment variable can point to a certificate chain that starts with the server identity certificate and includes the intermediate CA certificates in the order in which they are signed.

Valid in Server configuration file, cas.settings file, and operating system command line

Category Security

Example

```
env.CAS_INTERNODE_SSL_CERT="/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs/sas_encrypted.crt"
env.CAS_INTERNODE_SSL_CERT="/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs/cas/default/sas_encrypted.crt"
```

env.CAS_INTERNODE_SSL_KEY=<'path/key-file'>

Specifies the path and filename of the file that contains the private key used to sign the certificate specific to the CAS node being configured. This setting is likely to be different on every machine.

Valid in server configuration file, cas.settings file, and operating system command line

Category Security

Tip The contents of this file should be kept confidential.

Example

```
env.CAS_INTERNODE_SSL_CERT="/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/cas/default/sas_encrypted.key"
env.CAS_INTERNODE_SSL_CERT="/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/sas_encrypted.key"
```

env.CAS_INTERNODE_SSL_KEYPW=<'password'>

Specifies the password for the private key.

The setting is the password for the encrypted private key used to sign the certificate specific to the node being configured.

Note: This password is not encoded.

Valid in Server configuration file, cas.settings file, and operating system command line

Category Security

Example

```
env.CAS_INTERNODE_SSL_KEYPW='encryptedpassword'
```

env.CAS_INTERNODE_SSL_KEYPWLOC =<'path/certificate-file'>

Specifies the location of the password/key file for the encrypted private key used to sign the certificate specific to the node being configured.

Valid in Server configuration file, cas.settings file, and operating system command line

Category Security

Example

```
env.CAS_INTERNODE_SSL_KEYPWLOC='opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/cas/default/encryption.key'
env.CAS_INTERNODE_SSL_KEYPWLOC='opt/sas/viya/config/SASSecurityCertificateFramework/private/encryption.key'
```

env.CAS_PKCS12LOC=<'path/certificate-file'>

Specifies the path and filename of the PKCS12 (DER formatted binary) file that contains the certificate and private key.

Valid in Server configuration file, cas.settings file, and operating system command line

Category Security

Example `env.CAS_PKCS12LOC='/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs/sas_encrypted.p12'`
`env.CAS_PKCS12LOC='/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tls/certs/cas/default/sas_encrypted.p12'`

env.CAS_PKCS12PASS=<'path/password-file'>

Specifies the password for the private key specified by env.CAS_PKCS12LOC.

Note: This password is not encoded.

Valid in Server configuration file, cas.settings file, and operating system command line

Category Security

Example `env.CAS_PKCS12PASS='password'`

env.CAS_PVTKEYLOC=<'path/key-file'>

Specifies the path and filename of the file that contains the private key that corresponds to the digital certificate.

Valid in Server configuration file, cas.settings file, and operating system command line

Used by CAS REST API

Category Security

Tip The contents of this file should be kept confidential.

Example `env.CAS_PVTKEYLOC='/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/sas_encrypted.key'`
`env.CAS_PVTKEYLOC='/opt/sas/viya/config/etc/SASSecurityCertificateFramework/private/cas/default/sas_encrypted.key'`

env.CAS_PVTKEYPASS=<'password'>

Specifies the password for the private key specified by env.CAS_PVTKEYLOC.

Note: This password is not encoded.

Valid in Server configuration file, cas.settings file, and operating system command line

Used by CAS REST API

Category Security

Example `env.CAS_PVTKEYPASS='password'`

env.CAS_PVTKEYPASSLOC=<'path/key-file'>

Specifies the path and filename of the file that contains the private key that corresponds to the digital certificate.

Valid in Server configuration file, cas.settings file, and operating system command line

Used by CAS REST API

Category Security

Tip The contents of this file should be kept confidential.

Example

```
env.CAS_PVTKEYLOC='/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/encryption.key'

env.CAS_PVTKEYLOC='/opt/sas/viya/config/etc/
SASSecurityCertificateFramework/private/cas/default/encryption.key'
```

env.CAS_SSLCLIENTAUTH=<true>

When set to any value, causes client certificates to be validated when TLS connections are initiated.

Valid in Server configuration file, cas.settings file, and operating system command line

Category Security

Example env.CAS_SSLCLIENTAUTH=true

env.CAS_SSLCRLCHECK=<false>

When set to any value, causes the certificate revocation list (CRL) to be checked when TLS connections are initiated.

Valid in Server configuration file, cas.settings file, and operating system command line

Category Security

Example env.CAS_SSLCRLCHECK='false'

env.CAS_SSLNAMECHECK=<true>

When set to any value, causes the name of the server to be checked against the host name specified in the server identity certificate pointed to by env.CAS_CERTLOC to validate the server's identity.

Valid in Server configuration file, cas.settings file, and operating system command line

Used by CAS REST API

Category Security

Example env.CAS_SSLNAMECHECK=true

env.CAS_SSLREQCERT=<"NEVER | ALLOW | TRY | DEMAND">

Specifies what the client should do with the information sent by the server.

The variable env.CAS_SSLREQCERT must specify one of the following values:

■ **DEMAND**

The client asks for a server certificate. For the connection to continue, the server must provide a certificate, and the certificate must pass validation.

CAUTION! For security purposes, DEMAND is the setting that should be specified.

■ **NEVER**

The client never asks the CAS server for a certificate.

■ **ALLOW**

The client asks the server for a certificate. If the server does not provide a certificate, or if the certificate does not pass validation, the TLS connection continues.

■ **TRY**

The client asks the server for a certificate. If the server does not provide a certificate, the TLS connection continues. However, if the certificate does not pass validation, the TLS connection fails.

Valid in	Server configuration file, cas.settings file, and operating system command line
Used by	CAS REST API
Category	Security
Example	<code>env.CAS_SSLREQCERT='DEMAND'</code>

env.CAS_USE_HTTPS_ALL=<'TRUE' | 'FALSE'>

When set to TRUE, causes connections using the CAS REST API to use HTTPS.

Valid in	Server configuration file, cas.settings file, CAS Lua config files, and operating system command line
Used by	CAS REST API
Category	Security
Default	false
Example	<code>env.CAS_USE_HTTPS="FALSE"</code>

Configuration File Options for Data Transfer

CAS Configuration File Options for Data Transfer with the SAS Data Connect Accelerator

CAS server options are stored in a configuration file. During deployment, this configuration file, `casconfig_deployment.lua`, is created in the `/opt/sas/viya/config/etc/cas/default` directory. When you start the server with the `service sas-viya-cascontroller-default start` command, the options are read.

For more information about the CAS server configuration files, see see, [“Understanding Configuration Files and Start-up Files” on page 622](#).

These are the configuration options that can be used for data transfer encryption with the SAS Data Connect Accelerator. For a complete list of CAS configuration file options, see [SAS Viya Administration: SAS Cloud Analytic Services](#).

cas.DCSSLCERTISS='issuer-of-digital-certificate'

Specifies the name of the issuer of the digital certificate that TLS should use.

cas.DCSSLCERTLOC='pathname'

Specifies the location of a file that contains the digital certificate for the machine's public key also known as the identity certificate. This is used by servers to send to clients for authentication.

Requirement The certificate file must be PEM-encoded (base64).

See [“Certificate File Formats” on page 413](#)

cas.DCSSLCERTSERIAL='serial-number'

Specifies the serial number of the digital certificate that TLS should use.

cas.DCSSLCERTSUBJ='subject-name'

Specifies the subject name of the digital certificate that TLS should use.

cas.DCSSLPKCS12LOC=*pathname*

Specifies the location of the PKCS #12 DER encoding package file that contains the identity certificate and the private key.

Requirement If the cas.DCSSLPKCS12LOC= option is specified, the PKCS #12 DER encoding package must contain both the certificate and private key. The cas.DCSSLCERTLOC= and cas.DCSSLPVTKEYLOC= options are ignored.

See [“Certificate File Formats” on page 413](#)

[cas.DCSSLPKCS12PASS on page 444](#)

cas.DCSSLPKCS12PASS=*password*

Specifies the password that TLS requires in order to decrypt the PKCS #12 DER encoding package file.

Interaction The PKCS #12 DER encoding package is stored in the file that is specified by using the cas.DCSSLPKCS12LOC= option.

Note The cas.DCSSLPKCS12PASS= option is required only when the PKCS #12 DER encoding package is encrypted.

See [cas.DCSSLPKCS12LOC on page 444](#)

cas.DCSSLPVTKEYLOC=*pathname*

Specifies the location of the file that contains the private key that corresponds to the digital certificate that was specified by using the cas.DCSSLCERTLOC= option.

Requirement The key must be PEM-encoded (base64).

Note The cas.DCSSLPVTKEYLOC= option is required at the server only if the cas.DCSSLCERTLOC= option is also specified at the server.

See [“Certificate File Formats” on page 413](#)

[cas.DCSSLCERTLOC on page 443](#)

[cas.DCSSLPVTKEYPASS on page 444](#)

cas.DCSSLPVTKEYPASS=*password*

Specifies the password that TLS requires in order to decrypt the private key.

Interaction The private key is stored in the file that is specified by using the cas.DCSSLPVTKEYLOC= option.

Note The cas.DCSSLPVTKEYPASS= option is required only when the private key is encrypted. OpenSSL performs key encryption.

See [cas.DCSSLPVTKEYLOC on page 444](#)

cas.DCSSLPVTKEYPASSLOC=*pathname*

Specifies the location of the file that contains the password that TLS requires in order to decrypt the private key.

Interactions The private key is stored in the file that is specified by using the cas.DCSSLPVTKEYLOC= option.

If the DCSSLPVTKEYPASS option is specified, it is used. Otherwise, the DCSSLPVTKEYPASSLOC option is used.

See [cas.DCSSLPVTKEYPASS](#) on page 444

cas.DCTCPMENCRYPT='YES' | 'NO' | 'OPT'

Specifies whether encryption is required for the connection.

'YES' means that data encryption is required.

'NO' means that data will be sent as plaintext.

'OPT' means that data encryption is preferred but not required.

Aliases 'REQ' or 'REQUIRED' for 'YES'

'OPTIONAL' for 'OPT'

Default No value. However, if you specify `cas.DCTCPMENCRYPTALGORITHM='SSL'` and `cas.DCTCPMENCRYPT=` is not specified, `cas.DCTCPMENCRYPT=` defaults to 'YES'.

Requirement The option values must be uppercase.

Interactions Encryption is determined by the setting of this option on both the client (data provider) and server (CAS) side. For more information, see [“DCTCPMENCRYPT Option Setting Interaction” on page 388](#).

If you specify `cas.DCTCPMENCRYPTALGORITHM='SSL'` and `cas.DCTCPMENCRYPT` is not specified, `cas.DCTCPMENCRYPT` defaults to 'YES'.

Note If you have multiple clusters and you set the `DCTCPMENCRYPT=` option on the client (data provider) side to YES for one cluster and NO for another cluster, you might want to set the server (CAS) side `cas.DCTCPMENCRYPT=` option to 'OPT'.

cas.DCTCPMENCRYPTALGORITHM='SSL'

Specifies the algorithm to be used for encrypted data transfers using the SAS Data Connect Accelerator.

Default SSL

Requirement The option value, SSL, must be uppercase.

Interaction If you specify `cas.DCTCPMENCRYPTALGORITHM='SSL'` and `cas.DCTCPMENCRYPT=` is not specified, `cas.DCTCPMENCRYPT=` defaults to 'YES'.

Note SSL (TLS) is the only algorithm available at this time for encrypted data transfers using the SAS Data Connect Accelerator.

dcsecurity.properties File Options for Data Transfer with the SAS Data Connect Accelerator

You can set the following options in the `dcsecurity.properties` file. The `dcsecurity.properties` file is located in the following directory on your cluster.

- For Teradata, `/opt/SAS/SASTKInDatabaseServerForTeradata/14.00000/security`
- For Hadoop, `EPInstallDir/sasexe/SASEPHOME/security`

The syntax for setting the properties is as follows:

```
-option-name
option-setting
```

Here is an example.

-DCTCPMENCRIPTALGORITHM SSL

These are the options that you can set in the dcsecurity.properties file options for data transfer encryption with the SAS Data Connect Accelerator.

DCSSLCACERTDIR '*pathname*'

Specifies the directory where the public certificates for all of the trusted certificate authorities (CA) in the trust list are filed.

Requirement Each CA certificate file must be PEM-encoded (base64).

Interaction The DCSSLCALISTLOC= option can be used instead of or in conjunction with the DCSSLCACERTDIR= option.

Note Different versions of OpenSSL generate different hash values. For example, OpenSSL 0.9.8 generates different hash values from those generated by OpenSSL 1.x.

See [“Certificate File Formats” on page 413](#)

DCSSLCALISTLOC '*pathname*'

Specifies the location of a single file that contains the public certificate(s) for all of the trusted certificate authorities (CA) in the trust list.

Requirement The CA file must be PEM-encoded (base64).

Interaction The DCSSLCACERTDIR= option can be used instead of or in conjunction with the DCSSLCALISTLOC= option.

See [“Certificate File Formats” on page 413](#)

DCTCPMENCRYPT YES | NO | OPT

Specifies whether encryption is required for the connection.

YES means that data encryption is required.

NO means that data will be sent as plaintext.

OPT means that data encryption is preferred but not required.

Aliases REQ or REQUIRED for YES

OPTIONAL for OPT

Default NO. However, if you specify the DCTCPMENCRIPTALGORITHM option and DCTCPMENCRYPT is not specified, DCTCPMENCRYPT defaults to YES.

Requirement The option values must be uppercase.

Interaction Encryption is determined by the setting of this option on both the client (data provider) and server (CAS). For more information, see [“DCTCPMENCRYPT Option Setting Interaction” on page 388](#).

DCTCPMENCRIPTALGORITHM SSL

Specifies the algorithm to be used for encrypted data transfers using the SAS Data Connect Accelerator.

Default SSL

Requirement The option value, SSL, must be uppercase.

Note TLS is the only algorithm available at this time for encrypted data transfers using the SAS Data Connect Accelerator.

Examples

Create Site-Signed or Third-Party-Signed Certificates in PEM Format

Generate a Private Key in RSA Format and a Certificate Signing Request

The tasks that you perform to request a digital certificate for the CA, the server, and the client are similar. However, the values that you specify are different.

In this example, Proton, Inc. is the organization that is applying to become a CA. A certificate request is sent to a certificate authority to get it signed, thereby becoming a CA. After Proton, Inc. becomes a CA, it can serve as a CA for issuing other digital certificates to clients and servers on its network. The certificates generated by the Proton, Inc. CA are considered site-signed certificates.

Note: You can also sign the certificate yourself if you have your own certificate authority or create a self-signed certificate.

To create a site-signed certificate using OpenSSL, first you need to generate a private key in RSA format. This file is not protected with a passphrase and is saved in the ASCII (Base64-encoded) PEM format.

- 1 Edit your existing `openssl.cnf` file or create an `openssl.cnf` file. OpenSSL by default looks for a configuration file in `/usr/lib/ssl/openssl.cnf`. It is good practice to add `-config ./openssl.cnf` to the commands `OpenSSL CA` or `OpenSSL REQ` to ensure that OpenSSL is reading the correct file.

Note: You can find where the `openssl.cnf` file is located by submitting the OpenSSL command:

```
openssl version -d
```

.

Here is an example of some of the information that can be specified in the `openssl.cnf` file. You need to specify where OpenSSL should look for information. Here is a partial file example. Much more information about certificates can be specified.

Figure A.3 Example of an OpenSSL.cnf File

```

File Edit Format View Help
#
# openssl example configuration file.
# This is being used for generation of certificate requests.
#
#####
[ ca ]
default_ca = CA_default # The default ca section
#####
[ CA_default ]

dir = ./demoCA # where everything is kept
certs = $dir/certs # where the issued certs are kept
crl_dir = $dir/crl # where the issued crl are kept
database = $dir/index.txt # database index file.
new_certs_dir = $dir/newcerts # default place for new certs.

certificate = $dir/cacert.pem # The CA certificate
serial = $dir/serial # The current serial number
crl = $dir/crl.pem # The current CRL
private_key = $dir/private/cakey.pem # The private key
RANDFILE = $dir/private/.rand # private random number file

x509_extensions = usr_cert # The extensions to add to the cert

default_days = 365 # how long to certify for
default_crl_days = 30 # how long before next CRL
default_md = sha1 # which sha to use.
preserve = no # keep passed DN ordering
policy = policy_match

# For the CA policy
[ policy_match ]
policy = match

```

2 Select the `apps` subdirectory of the directory where OpenSSL was built.

3 Initialize OpenSSL.

```
$ openssl
```

4 Issue the appropriate command to request a digital certificate. In the example below, we are creating an RSA private key and generating a certificate signing request all at once.

Table A.37 OpenSSL Commands for Requesting a Private Key

Recipient of Certificate Request	OpenSSL Command
CA	<code>req -config ./openssl.cnf -new -out ca.csr -newkey rsa:2048 -keyout cakey.pem -sha256</code>
Server	<code>req -config ./openssl.cnf -new -out server.csr -newkey rsa:2048 -keyout serverkey.pem -sha256</code>
Client	<code>req -config ./openssl.cnf -new -out client.csr -newkey rsa:2048 -keyout clientkey.pem -sha256</code>

Table A.38 Arguments and Values Used in OpenSSL Commands

OpenSSL Arguments and Values	Functions
req	Requests a certificate.
-config ./openssl.cnf	Specifies the storage location for the configuration details for the OpenSSL program.
-new	Identifies the request as new.
-out ca.csr	Specifies the storage location for the certificate request.
-newkey rsa:2048	Generates a new private key along with the certificate request that is 2048 bits in length using the RSA algorithm.
-keyout cakey.pem	Specifies the storage location for the private key.
-nodes	Prevents the private key from being encrypted. Not Recommended. For best practice, encrypt the private key.
-sha256	Specifies the SHA-256 hash algorithm be used. Without this option, the default is SHA-1.

- 5 Informational messages are displayed and prompts for additional information appear according to the specific request.

To accept a default value, press the Enter key. To change a default value, type the appropriate information and press the Enter key.

Note: Unless the `-NODES` option is used in the OpenSSL command when creating a digital certificate request, OpenSSL prompts you for a password before allowing access to the private key.

Here is an example of a request for a digital certificate:

```
OpenSSL> req -config ./openssl.cnf -new -out ca.req -newkey rsa:2048
-keyout privkey.pem -nodes
```

```
Using configuration from ./openssl.cnf
```

```
Generating a 2048 bit RSA private key
```

```
.....+++++
```

```
.....+++++
```

```
writing new private key to 'cakey.pem'
```

```
-----
```

```
You are about to be asked to enter information that will be incorporated
into your certificate request.
```

```
What you are about to enter is what is called a Distinguished Name or a DN.
```

```
There are quite a few fields but you can leave some blank
```

```
For some fields there will be a default value,
```

```
If you enter '.', the field will be left blank.
```

```
-----
```

```
Country Name (2 letter code) [US]:
```

```
State or Province Name (full name) [North Carolina]:
```

```
Locality Name (city) [Cary]:
```

```
Organization Name (company) [Proton Inc.]:
```

```
Organizational Unit Name (department) [IDB]:
```

```
Common Name (YOUR name) []: proton.com
```

```
Email Address []: Joe.Bass@proton.com
```

Please enter the following 'extra' attributes to be sent with your certificate request
 A challenge password []:
 An optional company name []:
 OpenSSL>

The request for a digital certificate is complete.

Note: For the server, the Common Name must be the name of the computer that the server runs on. In our examples, we are using proton.com.

Generate a Public Certificate

- 1 Issue the appropriate command to generate a public certificate from the certificate signing request.

Table A.39 OpenSSL Commands for Generating Digital Certificates

Recipient of Generated Certificate	OpenSSL Command
CA	<pre>x509 -req -in ca.csr -signkey cakey.pem -out cacert.pem -sha256</pre> <p>Note: This command generates a self-signed certificate.</p>
Server	<pre>ca -config ./openssl.cnf -in server.csr -out server.pem -md sha256</pre> <p>Note: This command creates certificates signed by the CA. These are defined in the openssl.cnf file.</p>
Client	<pre>ca -config ./openssl.cnf -in client.csr -out client.pem -md sha256</pre> <p>Note: This command creates certificates signed by the CA. These are defined in the openssl.cnf file.</p>

Table A.40 Arguments and Values Used in OpenSSL Commands to Generate a Certificate

OpenSSL Arguments and Values	Functions
x509	Identifies the certificate display and signing utility. Typically used to generate a self-signed certificate.
-req	Specifies that a certificate be generated from the request.
ca	Identifies the Certificate Authority utility.
-config ./openssl.cnf	Specifies the storage location for the configuration details for the OpenSSL utility.
-in filename.csr	Specifies the storage location for the input for the certificate request.

OpenSSL Arguments and Values	Functions
-out filename.pem	Specifies the storage location for the certificate.
-signkey cakey.pem	Specifies the private key that is used to sign the certificate that is generated by the certificate request.
-md sha256	Specifies the SHA-256 hash algorithm be used. Without this option, the default is SHA-1.

- 2 Informational messages are displayed and prompts for additional information appear according to the specific request.

To accept a default value, press the Enter key. To change a default value, type the appropriate information, and press the Enter key.

Here is a sample of the messaging from a CSR for a server digital certificate:

Note: The password is for the CA's private key.

```
Using configuration from ./openssl.cnf
Enter PEM pass phrase: password
Check that the request matches the signature
Signature ok
The Subjects Distinguished Name is as follows
countryName           :PRINTABLE:'US'
stateOrProvinceName   :PRINTABLE:'NC'
localityName          :PRINTABLE:'Cary'
organizationName      :PRINTABLE:'Proton, Inc.'
organizationalUnitName:PRINTABLE:'IDB'
commonName            :PRINTABLE:'proton.com'
Certificate is to be certified until April 16 17:48:27 2016 GMT (365 days)
Sign the certificate? [y/n]:y
1 out of 1 certificate requests certified, commit? [y/n]:y
Write out database with 1 new entries Data Base Updated
```

The subject's Distinguished Name is obtained from the digital certificate request.

The generation of a digital certificate is complete.

Check Your Digital Certificate Using OpenSSL

To check a digital certificate, issue the following command:

```
openssl> x509 -text -in filename.pem
```

A digital certificate contains data that was collected to generate the digital certificate timestamps, a digital signature, and other information. However, because the generated digital certificate is encoded (usually in PEM format), it is unreadable.

Create a Certificate Chain in PEM Format Using OpenSSL

After generating a digital certificate for the CA, the server, and the client (optional), you must identify for the OpenSSL client application one or more CAs that are to be trusted. This list is called a *chain of trust*. This chain includes a set of certificates, where each one has been signed by the one that comes after it.

On the client, if there is only one CA to trust, specify in the client application the name of the file that contains the OpenSSL CA digital certificate. If multiple CAs are to be trusted, you can copy and paste into a new file the

contents of all the digital certificates of CAs to be trusted by the client application. These CAs can be primary, intermediate, or root certificates. Add the root CAs to the client's truststore.

For the server, do not include the Root CA in the server's certificate chain.

To manually create a new chain of trust, use the template shown below.

```
(Your Server Certificate - ssl.crt)

-----BEGIN CERTIFICATE-----

<PEM encoded certificate>

-----END CERTIFICATE-----

(Your Intermediate CA Certificate(s))

-----BEGIN CERTIFICATE-----

<PEM encoded certificate>

-----END CERTIFICATE-----

(Your Root CA Certificate)

-----BEGIN CERTIFICATE-----

<PEM encoded certificate>

-----END CERTIFICATE-----
```

The content of the digital certificate in this example is represented as `<PEM encoded certificate>`. The content of each digital certificate is delimited with a `-----BEGIN CERTIFICATE-----` and `-----END CERTIFICATE-----` pair. All text outside the delimiters is ignored. Therefore, you might not want to use delimited lines for descriptive comments.

Generally, OpenSSL returns `.pem` files, CAs return `.crt` files (Microsoft returns `.cer` files). Instead of manually cutting and pasting these files together (regardless of your file extension), you can also concatenate the certificate authority files. For example, you can take an intermediate authority certificate file, a root authority certificate file, and a primary certificate file and concatenate them into a single PEM file. Here is an example of concatenating certificates:

```
cat server.pem > certchain.pem
cat intermediateCA.pem >> certchain.pem
cat rootCA.pem >> certchain.pem
```

Because the digital certificate is encoded, it is unreadable. You will see a string of hexadecimal characters. To view the file contents, you can use the following OpenSSL commands for your file type:

```
openssl x509 -in cert.pem -text -noout
openssl x509 -in cert.cer -text -noout
openssl x509 -in cert.crt -text -noout
```

Use the following OpenSSL command to view a DER-encoded certificate:

```
openssl x509 -in certificate.der -inform der -text -noout
```

Note: If you are including a digital certificate that is stored in DER format into your certificate chain, you must first convert it to PEM format. For more information, see [“Convert DER to PEM File Format”](#) on page 399.

Verify Certificates in the Trust Chain Using OpenSSL

Clients and servers exchange and validate each other's digital certificates. All of the CA certificates that are needed to validate a server certificate compose a trust chain. All CA certificates in a trust chain have to be available for server certificate validation.

You can use the following OpenSSL command to verify that certificates are signed by a recognized certificate authority (CA):

```
openssl verify -verbose -CAfile <your-CA_file>.pem <your-server-cert>.pem
```

If your local OpenSSL installation recognizes the certificate or its signing authority and everything checks out (dates, signing chain, and so on), you get a simple OK message.

Encryption for Data at Rest

Encryption for Data at Rest: Overview

SAS Viya provides encryption in two contexts:

- Data at rest is data stored in databases, file servers, endpoint devices, and various storage networks. This data can be on-premises, virtual, or in the cloud. This data is usually protected in conventional ways by firewalls. Numerous layers of defense are needed, and encrypting sensitive data is another layer.

This document covers administrative tasks for encrypting files at rest in the SASHDAT format and it shows how tables that are imported into caslibs are encrypted. See [Concepts](#) for details.

- Data in motion is data that is being transmitted to another location. Data is most vulnerable while in transit. Sensitive data in transit should be encrypted. You can protect all traffic in transit between servers and clients. See [Encryption in SAS Viya: Data in Motion](#).

SAS Viya uses Advanced Encryption Standard (AES) algorithms with 256-bit keys to encrypt data at rest.

Refer to [SAS Cloud Analytic Services: Fundamentals](#) if you need additional background on data and caslib concepts.

Use one of the following interfaces to encrypt files at rest.

- To manage encryption of data files interactively, use [SAS Environment Manager](#).
- To programmatically encrypt data files, use the [CASLIB Statement](#).

How To (SAS Environment Manager)

Introduction

Authorized administrators use SAS Environment Manager to create and manage data security. The Domains area enables you to create a stored credential (an encryption key) that is available to designated identities to facilitate loading of encrypted files. By default, when you are creating a new caslib, encrypting that caslib is disabled. If you choose to enable encryption, this can be done by creating an encryption domain, and then associating that domain with your path-based caslib.

In the Domains area, you can perform the following tasks:

- Create a new encryption domain or use an existing encryption domain.
- Add users or group identities to the security domain.
- Create an encryption key (passphrase).

You must be a member of the *SAS Administrators* group and assume groups when you log on to SAS Environment Manager in order to create and manage Domains.

In the SAS Environment Manager Data area, you can create and manage caslibs. By default, caslibs are not encrypted. You can encrypt the tables in your caslibs by associating the caslibs with an encryption domain.



CAUTION! Be sure to keep a separate record of your encryption key. Once an encrypted caslib is created, it is not possible to change the encryption key setting or change the domain. The encryption key value cannot be retrieved through the software. If the caslib is deleted, the tables remain encrypted. To access those encrypted tables, you need to define a new caslib using the same path, domain, and encryption key.

Note: When the encrypted data is loaded into CAS, it is decrypted at load time. Caslib authorizations apply to accessing the loaded data. Identities that are set in the encryption domain provide authorization for who can request that encrypted tables be loaded. Identities can be custom groups or individual user IDs. Custom groups provide access control by simple group membership.

The following instructions explain how to encrypt the tables in your caslibs using SAS Environment Manager.

Navigation

The Domains area is available only if you are a member of the *SAS Administrators* group.

- 1 In the applications menu () , under **Administration**, select **Manage Environment**.
- 2 In the navigation bar, locate the **Security** section and click **Domains** .
- 3 You can select one of two views from the **Domains** page. The default view is **Domains**. From the **View** drop-down list, select one of the following views:

Domains

lists all domains. This is the default view. A SAS Administrator can see all domains. There are three types of domains: Authentication, Connection, and Encryption. On the Domains page, you can view the information for each domain that is defined, or you can create a new domain.


When you create an encryption domain, you cannot delete that domain.

Credentials

The credentials that represent the encryption keys are not currently shown in the Credentials view. For information about what is shown in the credentials view, see [SAS Viya Administration: External Credentials](#).

Manage Encryption Domains

Create a New Encryption Domain

- 1 In the **Domains** view, click .
- 2 In the New Domain window, specify general settings as follows:


ID	Create an ID name. Enter a unique ID for your encryption domain.
Type	Select the type of domain. There are three domain types, Encryption, Authentication, and Connection. Select Encryption from the list of available domains.
Identities	From the Select Identities window, you can select from users, groups, and custom groups. See below for how to add an identity.
Encryption key	Encryption passphrase or key.

Confirm encryption key	Enter the same passphrase as above.
Description	Add a description.


For additional information about identities, from the New Domain window, click ⓘ.

- To add identities, from the New Domain window, click +.

To add an Identities member, from the Select Identities window, perform the following tasks:




- In the left pane of the Select Identities window, select **Users**, **Groups**, or **Custom Groups** from the drop-down box.
 - Move the users, groups, or custom groups to the right. Click .
 - Click **OK** to save the information.
- After you have entered all of the parameter settings needed, click **Save**.

View Properties of an Encryption Domain

- In the **Domains** view, select an ID row for Type Encryption.
- Right-click, and select **Properties**. Or select  from the taskbar. From the **Domain Properties** window, the ID, Type, Description, Date created, Date modified, who created the domain, and who modified the domain is displayed.
- Click **close**.

Edit an Encryption Domain

If you are a member of the domain, you can add identities to and remove identities (users, groups, and custom groups) from an existing Encryption Domain and change the description. You cannot change the type of domain or change the Encryption key.

- In the **Domains** view, select a domain ID.
- Right-click, and select **Edit**. Or select  from the taskbar.
- To add or remove an Identity, from the Edit Domain window, click +.
 - To add more identities, in the left pane of the Select Identities window, select **Users**, **Groups**, or **Custom Groups** from the drop-down menu.
Select an identity, and move it to the right pane. Click .
 - To remove an identity, select an identity from the **Selected Identities** pane and move it to the left. Click .
 - Click **OK** to save the information.
- For additional information about identities, from the Edit Domain window, click ⓘ.
- Edit the **Description** of the Encryption Domain.
- After you have entered all of the parameter settings needed, click **Save**.

Delete Encryption Domains


Encryption domains cannot be deleted in the current release. If you try to delete an Encryption domain, you receive the following message: You cannot delete the encryption domain named 'domain-name'. Libraries associated with this domain will need to be recreated if the domain is deleted.

Manage Caslibs

For additional information, see [“Manage Caslibs” on page 280](#).

Create Caslibs with Encryption Enabled

If you want to encrypt the tables in a caslib, you must turn on encryption when you create the caslib. You cannot edit an existing caslib and encrypt it.

- 1 In the navigation bar, click **Data** .


Select **Libraries** from the drop-down view.

In the **Libraries** view, click .

- 2 In the New Caslib window, specify general settings as follows:

Note: A SAS administrator can see all of the properties and all domains.

Server	Select a server. Only servers to which you are authorized to add a global caslib are listed. See “SAS Cloud Analytic Services: Concepts” on page 618 .
Data source type	Select the type of data source. The Data Source area automatically displays the settings for the selected data source. Only PATH, HDFS, and DNFS can be encrypted.
Path	Specify data source-specific path information for the caslib.
Name	Specify a name for the caslib.
Description	Add a description.

- 3 In the **New Caslib** window, select the **Enable encryption** check box.
- 4 Select a domain from the list of available domains or create a new domain by selecting .


Note: Once you select **Enable encryption**, only the Encryption domains appear in the drop-down menu. If no encryption domains are defined, you must select the New icon to create one.

- 5 Click **Save**.


Modify a Caslib

You can change the caslib path or change the description of a caslib that is encrypted. The new path uses the same encryption domain defined in this caslib. It is not possible to change the encryption domain assigned to a caslib.

Note: Tables created in the previous topics are still encrypted and require the same key. Changing the path for a caslib with encryption enabled is not recommended. It is recommended that you create a new caslib using the same encryption domain and put a new path.

- 1 In the navigation bar, click **Data** .

Select **Libraries** from the drop-down view.

In the **Libraries** view, select a global caslib.
- 2 Right-click, and select **Edit**. Or select  from the taskbar.
- 3 In the **Edit Caslib** window, change the caslib path or description as needed.
- 4 Click **Save**.

When editing a caslib the following restrictions apply:

- Only **Path** based libraries can be edited.
- You cannot edit a personal caslib.
- You cannot change the encryption domain assigned to a caslib. If you need to change the encryption domain, you must create a new caslib.



View Properties of an Encrypted Caslib

- 1 In the **Libraries** view, select a caslib.
- 2 Right-click, and select **Properties**. Or select  from the taskbar. Read-only information is displayed.

Note: The Library Properties window shows the Encryption domain.


View and Edit Caslib Authorization

To manage tables (promote, create, drop, edit), delete source tables, alter tables and caslibs, and manage access to a selected caslib, you need the appropriate authorization. You can view your permissions or edit your permissions for a specified caslib. For additional information, see [“Manage Caslibs” on page 280](#).

- 1 In the **Libraries** view, select a caslib.
- 2 Right-click, and select **View Authorization**. Or select  from the taskbar, then **View Authorization** from the drop-down list.
- 3 You can edit who is authorized to import tables into a caslib and change the access level. Right-click, and select **Edit Authorization**. Or select  from the taskbar, then select **Edit Authorization** from the drop-down list.

To add identities, from the Edit Authorization window, click **+**.

To add an Identities member, from the Select Identities window, perform the following tasks:


- a In the left pane of the Edit Members window, select **Users**, **Groups**, or **Custom Groups** from the drop-down box.
- b Move the user, group, or custom group to the right. Click .
- c Click **OK** to save the information.

- a Here you can change the access level by sliding the control bar to the level of control that you want.
 - b Click **Save**.
- 4 Click **Close**.

Delete an Encrypted Caslib

You need the appropriate authorization to delete a caslib.

CAUTION! When you delete a caslib, all associated in-memory tables are immediately dropped.


- 1 In the **Libraries** view, select a caslib.
- 2 Right-click, and select **Delete**. Or select  from the taskbar.
- 3 In the confirmation window, click **Yes**.

Note: Deleting a caslib does not affect persisted files in the corresponding data source. The persisted encrypted SASHDAT files remain in the data source. If the user needs access to these files, the original encryption passphrase is required. To access these files, the user must create a new caslib with an encryption domain that uses the same passphrase.


Manage Tables

An encrypted caslib can contain a mix of encrypted and unencrypted tables. However, loading any table still requires the user to be a member of the encryption domain.


Display Encryption Status of Tables in a Caslib

- 1 In the **Libraries** view, select a caslib.
- 2 Right-click, and select **Tables**. Or select  from the taskbar.

A list of the tables that are assigned to the caslib is displayed.


- 3 To display which tables are encrypted, you need to display the Encryption column. The Encryption column displays whether a table is encrypted or not. Values are AES for encrypted tables and NONE if the table is not encrypted.
 - a Customize which columns are displayed in the Tables of a Caslib view. From the top right corner, select . Select **Manage Columns**. The Manage Columns window opens. It contains the columns that can be displayed for the selected table.
 - b Display the Encryption column. From the Hidden columns pane, select Encryption, and then choose to move it to the **Displayed columns** pane.
 - c Click **OK**.

Import an Unencrypted Table into an Encrypted Caslib


- 1 In the **Libraries** view, select your encrypted caslib.
- 2 Select the **Import** icon  from the taskbar.
- 3 In the Import Data To Caslib window, select a **Data Source** or select **Import**.

For more information, see [Getting Started with the Choose Data Window](#).

- 4 For this example, we are importing from a Local File. Select **Local File** and enter the path where the files are located or select a file to import.
 - a You can accept or enter the name of the **Target table name**. The **Target destination** is your encrypted caslib. In our case, we are importing a local table in Excel format into an encrypted caslib.
 - b Select **Import Item**.

Note: You will see a note that states that the item is being imported. When it is complete, you will see the following message: "The table was successfully imported on *Date/Time* and is ready for use."
 - c Select **Close**.
- 5 Right-click, and select **Tables**. Or select  from the taskbar.

A list of the tables that are imported into the caslib are displayed.

- 6 Verify that the unencrypted tables that were loaded into an encrypted caslib show as encrypted using AES. In the Library:*your Caslib* view, select  from the taskbar.

A list of the tables that are assigned to the caslib is displayed. Note that your new table shows the AES encryption status of the caslib if you have the **Encryption** column displayed. See "[Display Encryption Status of Tables in a Caslib](#)".

How To (Programming Tasks)

SAS Cloud Analytic Services supports encryption of SASHDAT files at the file level and at the directory level. As an administrator, you might want to simplify encryption for data at rest by configuring caslibs with an encryption password so that all files in a directory are encrypted. For information that describes how to set an encryption password at the directory level, see [CASLIB Statement](#), [DATASOURCE options](#) in *SAS Cloud Analytic Services: User's Guide*.

For an example of using the CASLIB statement to encrypt caslibs, see [Encrypt Tables in a Caslib](#) in *SAS Cloud Analytic Services: User's Guide*.

Encryption for Data at Rest: Concepts

Overview

When a caslib is created and designated as encrypted (an encrypted domain is associated), a table imported into that caslib is then encrypted in SASHDAT (.sashdat) format. A domain is associated with a caslib to provide access. Domains are used to store both the credentials (passwords and keys) that are required to access external data sources and the identities that are allowed to use those credentials.

When the encrypted tables are loaded into CAS (in-memory tables), these tables are identified as using an AES encrypted source. However, in-memory tables are not encrypted. The encryption applies to source tables, not to tables that are in memory.

You can specify encryption for a caslib only when it is created. To change it you must re-create the caslib.

SAS Viya supports encrypting files at rest in a path location. Only path-based (PATH, DNFS, HDFS) caslibs are supported.

Note: Cloud Foundry supports the path-based caslib types (PATH, DNFS, HDFS), as well as LASR and Hadoop.

SAS Viya uses Advanced Encryption Standard (AES) algorithms with 256-bit keys to encrypt data at rest.

Refer to [SAS Cloud Analytic Services: Fundamentals](#) if you need additional background on data and caslib concepts.

What Is a Domain?

Overview

Domains are used to store both the credentials (passwords and keys) that are required to access external data sources and the identities that are allowed to use those credentials. A domain contains one or more references to identities (users or groups) who have access to the credentials in the domain. A user can gain access to the credentials either directly with their user ID or indirectly as a member of a group that is defined as an Identity.

The ID, or name, of a Domain is used in the definition of a non-path-based caslib to access and load tables from external databases. A domain is associated with a caslib to provide access. External data sources include LASR, Oracle, Teradata, Hadoop, Postgres, and Impala. Users of a caslib with an associated domain do not have to know or enter database credentials to access or load external data.

Note: Cloud Foundry supports these caslib types: PATH, DNFS, HDFS, LASR, and Hadoop.

There are three Domain types: Authentication, Connection, and Encryption.

What Is an Encryption Domain?

An encryption domain is used to store an encryption key that is required to read data at rest in a path assigned to a caslib. The Identities selected in this encryption domain have access to the key. When you create a path-based caslib, you can choose to enable encryption. You then select or create an encryption domain to assign an encryption key. Tables imported to this caslib are now encrypted. If the path contains preexisting tables, those tables are not encrypted. Users who are not defined in Identities as individuals or as members of a group cannot load data from this caslib.

Note: It is recommended that you start with an empty caslib and import either encrypted or unencrypted tables. Mixing these types is not recommended.

Encryption domains are used to store encryption keys that can then be associated with a caslib type of PATH, HDFS, or DNFS.

What Is a Connection Domain?

A connection domain is used when the external database has been set up to require a user ID but no password. For information about using connection domains, see [SAS Viya Administration: External Credentials](#).

What Is an Authentication Domain?

An authentication domain is a name that facilitates the matching of logons with the servers for which they are valid. Authentication domains are used to store credentials that are used to access an external source (for example, an Oracle database) that can then be associated with a caslib of the appropriate type.

The software attempts to use only the credentials that it expects to be valid for a particular resource or system. The software's knowledge of which credentials are likely to be valid is based entirely on authentication domain assignments.

Authenticating to SAS can be done through SAS logon. For more information, see [SAS Viya Administration: Authentication](#).

For information about using Authentication Domains, see [SAS Viya Administration: External Credentials](#).

Defaults

In a new deployment, encryption for data at rest is not automatically enabled. You can configure encryption of data that is added to PATH, HDFS, and DNFS caslibs by using SAS Environment Manager and the programming interfaces.

Encrypting Caslibs

SAS Viya supports encryption as an option for tables in caslibs. The encryption applies to source tables, not to tables resident in memory. When you create a caslib, you set the encryption option and assign a key. The tables that you import into the library then become encrypted and all have the same key.

When you import a table (SAS table, .csv file, .txt file, Excel file, and so on) into a caslib, a .sashdat file is created in the same path location. If the caslib is enabled for encryption, those .sashdat files at rest are now encrypted. When these tables are loaded into CAS, these in-memory CAS tables are not encrypted. The in-memory tables can be used by any user with authorization to access the caslib.

If you delete the caslib, the tables in the associated path remain encrypted. To access those tables again, you must create a new caslib, enable encryption, and use the same encryption domain with the same key value, or create a new encryption domain with the same key value.

Considerations for Encrypting Tables in Caslibs

Here are a few best practices and considerations when encrypting data at rest:

- Only PATH, HDFS, or DNFS files can be encrypted.
- It is best not to mix encrypted and unencrypted tables in a caslib path. Only the user IDs and groups in the domain identities are able to read the encrypted data.

When you create a new caslib and enable encryption, only the newly imported tables are encrypted and stored in the path. A best practice is to make sure that the path is empty first before you import the tables that you want encrypted.

- Encryption of data at rest has some performance costs, and user and administrative overhead. You must balance the goals of security and performance at your site when deciding to encrypt data. Users and administrators must keep track of keys (passphrases and passwords) when accessing the data. The system uses additional CPU resources when loading and saving encrypted tables.

Reference

PROC PWENCODE

In the programming environment, the ENCRYPTIONPASSWORD= option in the CASLIB statement specifies a password for encrypting or decrypting tables. For additional password security, you can use the PWENCODE procedure to encode that password. Encoded passwords can be used in place of plaintext passwords in SAS programs that access databases and various servers. For information, see [Base SAS Procedures Guide](#).

Using SAS Environment Manager

What Is SAS Environment Manager?

SAS Environment Manager is a web application for managing a SAS Viya environment. It includes a dashboard view, which provides a quick overall look of your environment's health and status, as well as detailed views that enable you to examine and manage your environment in detail.

You can use the application to manage these areas of your environment:

Data

- CAS tables, caslibs, user-defined formats, and servers

User content

- Saved reports and data, favorites, and history

User information

- Users and groups from your directory service and SAS groups

License information

- Your SAS licenses and expiration dates.

System backups

- Backups and restores of system data

Configuration

- Configuration data for SAS Viya microservices

Contexts

- Settings such as environment variables and port ranges that are used when launching a compute process

Logs

- Log messages from SAS applications and services.

Machines

- Information and metric data for the machines and services

Scheduling

- Schedules for job service jobs

Domains

- Authentication domains (for storing a user ID and password), encryption domains (for storing an encryption key), and connection domains (for storing a user ID without a password)

Passwords

- System passwords

Mobile device access

- Lists that allow or prevent access to the system by specific mobile devices.

Rules

- Access controls and rules that control who can access resources and content in your system

Quality Knowledge Bases


Collections of files that store data and logic that define data quality operations such as parsing, standardization, and matching (available only if SAS Data Quality is installed)

Tenants

Information about tenants and status of tenant services (available only in a multi-tenant environment and only to provider administrators)

Note: If you are using only the SAS Viya programming interface, SAS Environment Manager is not deployed. See [“Deployment Types” on page 1](#) for more information.

Accessing SAS Environment Manager

To access SAS Environment Manager open SAS Home. You can click on the **Manage Environment** tile, or in the Applications menu () under **Administration**, select **Manage Environment**.


Note: If you are using only the SAS Viya programming interface, SAS Environment Manager is not deployed. See [“Deployment Types” on page 1](#) for more information.


When you log on to SAS Environment Manager, if you are a member of the SAS Administrators group, a prompt will appear asking you whether you want to opt in to your assumable groups. If you select **Yes**, your membership in the SAS Administrators group is in effect. See [“Assumable Custom Groups” on page 486](#) for more information.

To sign out of SAS, use the application bar. Click your name, and then click **Sign out**. When you click **Sign out**, you sign out of all SAS web applications.

Using the Dashboard

Overview

The Dashboard provides a quick view of the state of your system. It displays a set of tiles and reports, each of which summarizes an aspect of system status. The Dashboard is the default view when you first open SAS Environment Manager. You can return to the Dashboard from any page in SAS Environment Manager by clicking  **Dashboard** from the navigation menu.

Click  then **Refresh** to refresh the data. Some tiles are also automatically refreshed, as noted in the tile's description on this page.

By default, the **Dashboard** displays these tiles:

- **Availability**
- **CAS System Health**
- **Logged Issues**
- **Mobile Devices**

You can also display system performance reports on the Dashboard by clicking **Show Reports**.

Using the Dashboard Tiles

The Dashboard can include these tiles:

Availability

displays grids of color-coded boxes that correspond to the machines, services, and service instances in your environment. The colors reflect the status of each machine, service, and service instance. A green box indicates that the item is available, a yellow box indicates that it is partially available or in a warning state, and a red box indicates that it is unavailable.

The grids are updated every ten seconds.

Selecting a box in one of the grids highlights the corresponding boxes in the other two grids. The box you selected is outlined with a solid line, and the associated boxes are outlined with a dashed line. These are the associations between the selected boxes:

- When you click on a box in the **Machines** grid, the services and service instances that are running on that machine are highlighted in the **Services** and **Service Instance** grids.
- When you click on a box in the **Services** grid, the machines where that service is running are highlighted in the **Machines** grid, and the instances of the service are highlighted in the **Service instances** grid.
- When you click on a box in the **Service instances** grid, the machines where the service instance is running are highlighted in the **Machines** grid, and the service is highlighted in the **Services** grid.

Note: To deselect a box, hold down the Ctrl key and click the box. You can also hold down the Ctrl key and press the spacebar.

Place your cursor over a box to view the name of the machine, service, or service instance.

Double-click on a box in the **Machine** grid to open the Services dialog box, which lists the services that are running on that machine and their availability. You can open the Machines page for the selected machine from this dialog box.

Click on a box in the **Service instances** grid to view the machine address and port where the instance is running.

Use the **Filter** field to display only certain machines, services, and service instances. As you enter characters in the Filter field, the boxes displayed in the Availability area dynamically change. The boxes that are displayed either match the characters that you type, or are associated with the boxes that are displayed. For example, typing “**laun**” in the Filter field might cause two **Services** boxes to display (for the Launcher service and the Launcher server), only the **Service instance** boxes associated with the displayed services, and only the **Machines** boxes associated with the displayed services.

CAS System Health

displays graphs that give you a quick view of the state of the nodes (machines) in your SAS Viya cluster for a selected CAS server. Use the buttons at the top of the tile to select the graph that you want to view.

If your environment contains more than one CAS server, a menu above the graph enables you to select the server to view. When you log into SAS Environment Manager, this tile attempts to connect to the default CAS server. If the default server cannot be found, the tile displays information for the first server to which it can connect. If it can connect to the default server, but the server does not respond within five seconds, the tile displays a message. You can then retry the server or choose another server. If you log out of SAS Environment Manager, the later log back in, the tile displays information about the server you last selected. See “[SAS Cloud Analytic Services: Overview](#)” on page 607 for more information on CAS servers.

You define the default server in the casManagement configuration property. See “[Configuration Properties: How To Configure Services](#)” on page 215 for more information.


The **Node Memory Usage** graph displays the memory usage for each node in your cluster. Each bar represents a separate node. Position your pointer over a bar on the graph to view the name of the node and its memory usage. The vertical scale of the graph changes to match the memory usage of the most heavily used node.

The **CPU Load** graph displays the 1-minute CPU load average over the past five minutes for the nodes in your cluster. The chart updates every ten seconds. Each node is represented by a separate line on the graph. The vertical scale of the graph changes depending on the largest value being displayed in the chart.

Position your pointer on any of the lines on the graph to view the name of the node and the CPU load at the selected time.


The graphs update every ten seconds.

Logged Issues

displays a time series graph of the number of ERROR and FATAL level log messages captured by SAS Viya log files in the last 30 minutes. Only the top five sources of ERROR and FATAL messages are included. If there have been no ERROR or FATAL messages in the last 30 minutes, the message **No information is available** displays in place of the graph. To view details about the messages or to filter the displayed messages, click  and then select **Open** to display the **Logs** page.

Custom Groups

displays the name and number of members for the top five custom groups (by number of members). Custom groups are created to control access to SAS Viya features. If you have a sufficient authorization level, you can use the **Users** page to manage custom groups. See [“Getting Started with Identity Management” on page 476](#). This tile refreshes whenever the Dashboard is reloaded.

If you have the proper authorization, click on  and then click on **Open** to display the **Users** page.

Mobile Devices

displays the type of mobile device access control in use and the number of successful and unsuccessful logon attempts. You can use the **Mobile Devices** page to manage mobile device access and view detailed information about access attempts. See [“Mobile: How To” on page 557](#). This tile refreshes whenever the Dashboard is reloaded. If you do not have a sufficient authorization level, this tile does not appear.

If you have the proper authorization, click on  and then click on **Open** to display the **Mobile Devices** page.

Reports

You can display two types of reports on the Dashboard.

User-selected reports are selected by right-clicking on the report in the Content page and selecting **Pin to dashboard**. References to the selected reports are stored in the folder `Users/user_name/Application Data/SAS Environment Manager/Dashboard Items`. Any user can pin reports to their dashboard.

System performance reports are provided by default in SAS Viya, and are only available to SAS administrators. These reports are available:

Application Activity

performance and usage by application

CAS Activity

CPU, memory usage, and system performance for CAS

Disk Space

disk usage history and forecast

Infrastructure Data Server Tables

size and usage for SAS Infrastructure Data Server tables

Message Queue Activity

activity and traffic on the RabbitMQ message exchanges that are used by the operations infrastructure

System Activity

memory and CPU usage and network activity

User Activity


reports based on audit records

By default, the reports are hidden. Click **Show Reports** to display thumbnails for the reports. If you display the reports and then log out of SAS Environment Manager, the reports will display when you log back into SAS Environment Manager.

Select  in any report tile and then click on **Open** to display the full report in SAS Report Viewer.

Click › and ‹ to change the displayed reports.

Personalizing Your Dashboard

- To add or remove a tile, at the top of the window, select **your_user_name** ⇒ **Settings**. In the Settings dialog box, select **Dashboard**. Select the check boxes for the tiles that you want to display on the Dashboard.
- To remove a tile that is currently displayed on the Dashboard, click  and then click **Unpin**.
- To choose which system status reports to display on the Dashboard, at the top of the window select **your_user_name** ⇒ **Settings**. In the Settings dialog box, select **Public Dashboard Items**. Select the check boxes for the reports that you want to display on the Dashboard. These reports are available only to administrators.
- To add a system status report to those provided by default, place the report in the folder `/Products/SAS Environment Manager/Dashboard Items`. An administrator can then pin the report to the reports in their Dashboard.
- To add a report to your dashboard, navigate to the report in the **Content** area of SAS Environment Manager. Right-click on the report in the folder tree and select **Pin to dashboard** from the pop-up menu. The report is added to the report gallery and is copied to the folder `/user/Application Data/SAS Environment Manager/Dashboard Items`.
- To choose which of your reports to display on the **Dashboard**, at the top of the window select **your_user_name** ⇒ **Settings**. In the Settings dialog box, select **My Dashboard Items**. Select the check boxes for the reports that you want to display on the Dashboard.

SAS Environment Manager Functions

Accessing the Functions

To access a SAS Environment Manager page, select it from the navigation bar.

Depending on your organization's environment and authorization policies, you might not have access to all pages. If you are a member of the SAS Administrators group and opt in to the group when you sign in, you can access all of these pages. If you are not a member of the SAS Administrators group, or you do not opt in to the group, you can access only the **Dashboard**, **Data**, and **Content** pages. Your organization might also use other groups that could restrict your ability to access certain pages. See [“Getting Started with Identity Management” on page 476](#) for more information.

Data

Select  **Data** from the navigation menu to view and manage the tables, libraries and servers that contain your data. You can select four views of your data from the **View** menu.

The **Loaded Tables** view displays information about all of the data tables that have been loaded into CAS memory. The window displays basic information about each table, such as the state (loaded or unloaded), server location, size, and the dates when the table was created and modified. From this view, you can also unload or delete tables, manage authorization for each table, view table properties, and view information about a table's columns. You can edit the path and description for path-based libraries.

The **Libraries** view displays information about the libraries defined for the current servers. The window displays basic information about each library, such as the associated server, path for the data source, and status. While in the **Libraries** view, you can also view a library's properties, manage authorization for a library, view a list of all


tables (loaded and unloaded) associated with a library, add or delete a library, view tables that are included in a library, and import data into a library. You can also change the path and description for a library.

The **Servers** view displays information about the SAS Viya servers. The window displays basic information about the servers, such as the status, host, and port. While in the **Servers** view, you can also view detailed properties for the server, including the libraries in the server and the users and groups that are superusers for the server. If you have the proper authorization, you can also assume the superuser access for the server, which enables you to edit the server properties.

The **User Defined Formats** view displays information about all of the user-defined formats that are available for the data. The window displays a list of available user defined formats and enables you to add new formats, import new formats, or edit existing ones. This view is available only for administrators.

See [“Data Administration: Reference” on page 298](#) for more information.

Content

Select  **Content** from the navigation menu to display folders that contain items that users have saved. When you open the Content page, you have access to your own data in the My Folder folder. If you have administrative access, you can also view the folders of other users. From this page, you can create, delete, move, and rename folders, create shortcuts, and manage the authorization for any folder or item that you select (if you have sufficient authorization). You can also export the reports in a folder to a package file and import the reports from a package file into a folder.

Each user’s folder contains several subfolders:

My Favorites

contains references to items identified as favorites, to enable quick access to often-used reports and data.

My Folder

contains saved items.

Application Data


contains items used by SAS Viya applications, such as items pinned to a user’s Dashboard in SAS Environment Manager.

My History

contains a list of the most recent items that you have accessed. You can select entries in this folder to quickly return to items that you have worked with recently.

See [“Content Management: How To” on page 257](#) for more information.


Users

Select  **Users** from the navigation menu to view information about users and groups, and to manage custom groups. The information displayed for users and groups comes from your organization’s directory service (such as LDAP or Microsoft Active Directory). Because this information is managed by your identity provider, it is displayed as read-only data in SAS Environment Manager.

You can also manage custom groups on the **Users** page. Custom groups enable you to manage special permissions for groups of users.

See [“View User and Group Information” on page 476](#) for more information.


Licensed Products

Select  **Licensed Products** from the navigation menu to view information about the licenses for your products. You can view a list of all your currently licensed products and see the expiration date, grace period, warning period, and maximum CPU count for each one. You can filter the list by any of the displayed criteria to make it easier to find products in the table.



See [“Licensing: Overview” on page 499](#) for more information.

Note: This page is not available for a tenant administrator.

Backup and Restore


Select  **Backup and Restore** from the navigation menu to back up and restore your environment.

Configuration


Select  **Configuration** from the navigation menu to manage the configuration settings for SAS Viya services. You can select from a list of basic services, all services, or definitions. When you select a service, the service’s configuration properties are displayed on the right side of the window. Click  to change any of the displayed properties.

See [“Introduction” on page 215](#) for more information.

Contexts

Select  **Contexts** from the navigation menu to manage launcher contexts. A context is a collection of settings such as environment variables and port ranges that are used when launching a SAS Viya instance. You can also specify settings for a deployment, such as the deployment ID and the installation and configuration directories.


Logs

Select  **Logs** from the navigation menu to view information about messages that have been written to the logs. You can view a chart of the number of log messages and a table of the detailed messages. By default, the chart and table reflect the messages logged during the previous 30 minutes, but you can select a different time range. You can also search for a specific message or filter the messages by level and source.

See [“Logging: Overview” on page 505](#) for more information.

Note: This page is not available for a tenant administrator.

Machines


Select  **Machines** from the navigation menu to monitor the machines in your environment and the service instances running on those machines. You can view this information:

- charts of the percent of CPU utilization and memory used for each machine
- status of predefined checks (such as disk or memory usage) for the selected machine
- service instances running on the selected machine, along with their current state
- properties for the selected server


See [“Monitoring: How to \(SAS Environment Manager\)” on page 518](#) for more information.

Note: This page is not available for a tenant administrator.

Scheduling


Select  **Scheduling** from the navigation menu to schedule jobs. You can select time events to use as triggers to start jobs. You can run scheduled jobs immediately, edit schedules, and unschedule jobs. See [Scheduling on page 587](#) for more information.

Domains


Select  **Domains** from the navigation menu to manage domains used for authentication, encryption, and connection.

See [“About the Domains Page” on page 61](#) for more information.

My Passwords


Select  **My Passwords** from the navigation menu to manage passwords used to access domains in your system. See [“Manage Personal Passwords” on page 68](#) for more information.

Mobile Devices

Select  **Mobile Devices** from the navigation menu to manage how mobile devices access certain reports. You can use either a blacklist or a whitelist. If you use a whitelist, all mobile devices are blocked except for those listed in the whitelist. If you use a blacklist, all mobile devices are allowed except for those listed on the blacklist. The **Mobile Devices** page displays a table of recent access attempts by mobile devices, the devices listed in the blacklist, and the devices listed in the whitelist. The page indicates whether the blacklist or the whitelist is being enforced, and enables you to select which list to use. You can also add devices to either list.


See [“Mobile: How To” on page 557](#) for more information.

Rules


Select  **Rules** from the navigation menu to manage access to specific locations and content.

See [“General Authorization: How to \(Rules Page\)” on page 114](#) for more information.

Quality Knowledge Bases

Select  **Quality Knowledge Bases** from the navigation menu to view and manage Quality Knowledge Bases, which are collections of files that store data and logic that define data quality operations such as parsing, standardization, and matching. This area is available only if you have licensed SAS Data Quality. See [QKB Management on page 355](#) for more information.


Tenants

If you are the provider administrator of a multi-tenant environment, select  **Tenants** from the navigation menu to view information about tenants and tenant services.

How To

Work with Information Displayed in Tables

When viewing information that is displayed in a table in SAS Environment Manager, use these tips to control how data is displayed:

- To sort a table, right-click on a column header and select **Sort**. You can sort the table by the contents of the column, or add the column as a secondary sort criteria.
- To reorder the columns in a table, click on the column heading and drag the header to the new location.
- To prevent a column from being reordered, right-click on the column heading and select **Freeze**. The column is moved to the left of the table and cannot be reordered. To enable the column to be moved, right-click on the header and select **Unfreeze**.
- To select which columns are displayed, click on the **Column icon**  on the right side of the table header. The **Columns** window displays a list of hidden columns and displayed columns. Select the columns that you want to display and click **OK**.

Note: Not all of the tables in SAS Environment Manager use all of these features.

Manage Settings

Access Settings

To access the Settings window, select your user name in the upper right of the SAS Environment Manager window. Select **Settings** from the pop-up menu.

Set Locale

The locale specifies the language that is used for SAS Environment Manager text. It also specifies how text is sorted and the format used for displaying numbers and dates.

To specify the locale, select **Global** ⇒ **Region and Language**. Select the locale in the field **Locale for regional formats and sorting**. The default value is **Browser locale**, which specifies that the locale setting is inherited from the web browser that you use to display SAS Environment Manager. The new locale setting takes effect after you sign out and sign back in to all of the SAS applications that are running.

Choose a Theme

The theme is the collection of colors, graphics, and fonts used in SAS Environment Manager. Use the **Theme** options to select the default theme (as chosen by your system administrator) or another available theme. The theme that you choose takes effect after you close the Settings window.

Reset Messages

In SAS Environment Manager, some warning and confirmation messages include an option to not see the message again in the future. Select **Global** ⇒ **General** and click **Reset Messages** to clear all of those settings and to view all warning and confirmation messages.

Manage Accessibility Settings

Accessibility features are part of the global settings, which are applied to all SAS web applications. Accessibility features are not specific to SAS Environment Manager.

The following accessibility features are available:

Enable sounds

Issue sounds for actions in the application

Enable visual effects

Enable visual effects such as fades and wipes.

Rearrange column content when space is limited

Rearrange the information in a column if there is not enough space to display all of the information. If you are using a screen reader, clear this check box. Clearing the check box prevents data from flowing into adjacent columns, and enables screen readers to navigate tables more consistently.

Invert application colors

Use light-colored text on a dark background, which makes the user interface easier to see.

Customize the focus indicator settings

Makes the focus indicator easier to see by adjusting the color, thickness, and opacity.

What is Available to a Tenant Administrator?

If you are the tenant administrator in a multi-tenant system, these functions are not available:

- Licensed Products
- Tenants
- Logs
- Machines

By default, the Dashboard for a tenant administrator contains these items:

- **CAS System Health** tile
- **Mobile Devices** tile
- **Top 5 Custom Groups** tile
- Personal reports that are pinned to the Dashboard

Other Dashboard items that are listed in this document are not available to tenant administrators.

Identity Management

<i>Identity Management Overview</i>	475
<i>Getting Started with Identity Management</i>	476
Give Other Users Specialized Access	476
Give Users Access to Data and Content	476
<i>Identity Management How To</i>	476
View User and Group Information	476
Manage Custom Groups	477
Manage CAS Role Memberships	479
Manage Access to Functionality	481
<i>User Management: Guidelines and Best Practices</i>	482
<i>Identity Management Concepts</i>	483
Initial Users	483
Identity Providers	484
Custom Groups	485
Access to Functionality	487
CAS Server Roles	493
<i>Identity Management Reference</i>	495
Initial Rules for Access to Functionality	495
<i>User Management: Interfaces</i>	496
Interfaces for User Management	496
<i>Identity Management: Troubleshooting</i>	496
Cannot Sign In to SAS Studio	496
Cannot Access Cloud Analytic Services	496
Cannot Sign In to CAS Server Monitor	496
Cannot Administer SAS Home	497
Cannot View Users and Group Members	497
Cannot Access Esri Geographic Mapping Resources	497

Identity Management Overview

User and group identities are stored and managed by your organization's identity provider (for example, Microsoft Active Directory). Read-Only access to the provider enables SAS to authenticate users and obtain identity information at sign-on.

SAS identity management is not used in a programming-only deployment. In such deployments, your operating system user management is used.

SAS identity management includes the following:

- managing the membership of custom groups and CAS roles
- giving users, groups, and custom groups access to SAS functionality

Getting Started with Identity Management

After deploying, perform these tasks to set up your identities.

Give Other Users Specialized Access

In the initial deployment, authenticated users automatically have access to functionality that is appropriate for a typical user. See [“Initial Rules for All Authenticated Users” on page 495](#).

To give additional functionality to special categories of users (for example, administrative users), follow these steps:

- 1 Become familiar with the predefined custom groups and their associated levels of functionality. See [“Predefined Custom Groups” on page 485](#) and [“Access to Functionality” on page 487](#).
- 2 Become familiar with the CAS server roles. See [“CAS Server Roles” on page 493](#).
- 3 Based on this information, determine which of your users and groups to add to each role and each predefined custom group.
- 4 Add users and groups to the appropriate custom groups. See [“Add or Remove Custom Group Members” on page 477](#).
- 5 Add users and groups to the appropriate CAS server roles. See [“Manage CAS Role Memberships” on page 479](#).

Give Users Access to Data and Content



You use the SAS Viya authorization layer to give users access to the data and content that they need to do their jobs. See [Authorization Orientation on page 71](#) for more details.

See Also



- *SAS Viya Administration: Orientation to Authorization*

Identity Management How To

View User and Group Information

- 1 In the applications menu () , select **Administration** ⇒ **Manage Environment**. In the navigation bar, select .
- 2 On the Users page, you can do the following:
 - Select **Users**, **Groups**, or **Custom Groups** from the drop-down list in the toolbar. Custom groups are displayed when you first open the page.

Note: A custom group is a group that exists in SAS but not in your identity provider.

- Enter a string in the **Search** field to search for identities within the category that you selected (Users, Groups, or Custom Groups). To restore the complete list of identities, clear the search field.
- Click an identity in the left pane to see its properties in the right pane. An identity's properties include the following:
 - basic properties including name, ID, and description
 - contact information (for users only)
 - a list of members (for groups and custom groups only)
 - a list of groups that the identity is a member of.  indicates custom groups, and  indicates groups from your identity provider.

Note: Properties for users and groups (other than memberships in custom groups) are retrieved from your directory service and are read-only. Properties for custom groups are stored in SAS and can be edited using SAS Environment Manager.


- Access recently viewed identities by using the drop-down box at the top of the right pane.

Note: To add, edit, or delete users and groups (other than custom groups), use your organization's identity provider (for example, Microsoft Active Directory) to which SAS Viya is connected.



Manage Custom Groups

A custom group is a group that exists in SAS Viya but not in your identity provider. Your deployment includes a set of [predefined custom groups](#), which provide an easy way to give users access to specialized functionality. You can also create your own custom groups, which are useful if you do not want to (or do not have permission to) create groups in your identity provider.

Add or Remove Custom Group Members

- 1 On the Users page in SAS Environment Manager, select **Custom Groups** from the drop-down list in the toolbar.
- 2 In the left pane, click the name of the group whose members you want to update.
- 3 In the **Members** section of the right pane, click .

The Edit Members window displays the custom group's current members in the right pane.

- 4 To add a member, do the following:
 - a In the left pane of the Edit Members window, select **Users**, **Groups**, or **Custom Groups** from the drop-down box.
 - b In the left pane, click the name of a user, group, or custom group identity. The identity's properties are displayed in the far right pane.
 - c Click .
- 5 To remove a member, do the following in the Edit Members window:
 - a In the **Select Identities** list, click the user, group, or custom group identity that you want to remove. The identity's properties are displayed in the right pane.
 - b Click .


- When you are finished adding and removing members, click **OK**.

Note: If you add or remove a user, the change takes effect the next time that this user logs on to SAS Environment Manager. If the user is currently logged on, his or her previous memberships continue to apply.

Create a New Custom Group

Create custom groups to give members similar permissions.


There are many uses of custom groups, but there is one specific use-case to be aware of. By default, authenticated users who launch a CAS session will do so as the `cas` account. Files generated in such a session are saved in a folder belonging to the `cas` account, but in a directory path that includes the user's ID. If you prefer users to launch CAS sessions under their own account to cause their files to be saved to their UNIX directories, create and populate a custom group with ID `CASHostAccountRequired`. When members of the `CASHostAccountRequired` group launch a CAS session, that session runs under that user's host account, and the generated files are created under the user's home directories.

- On the **Users** page in SAS Environment Manager, select **Custom Groups** from the drop-down list in the toolbar.
- Click  in the toolbar.
- In the New Custom Group window, enter a unique name and ID for the group. You can also enter a description.


CAUTION! Do not use an apostrophe (') in a custom group ID. The use of an apostrophe (') interferes with the use of that group's identity on the **Users** page in SAS Environment Manager as well as accessing that group's identity when working with authorization.

TIP Create an ID that is easily recognizable. For example, for the group "Report Testers", you could use "ReportTesters" as the ID.

- Click **Save**.


TIP You can also create a custom group by copying an existing group or custom group. To do so, click the existing group (or custom group) and select . You can then edit the properties and members of the new custom group as needed.

Edit a Custom Group's Basic Properties

- On the **Users** page in SAS Environment Manager, select **Custom Groups** from the drop-down list in the toolbar.
- In the left pane, click the name of the group whose properties you want to edit.
- In the **Basic Properties** section of the right pane, click .
- In the Edit Custom Group window, enter your changes to the name or description.

Note: You cannot edit the ID of a custom group.
- Click **Save**.





Delete a Custom Group


- 1 On the Users page in SAS Environment Manager, select **Custom Groups** from the drop-down list in the toolbar.
- 2 Click the custom group that you want to delete. The group's properties are displayed in the right pane.
- 3 Click , and then click **Delete** in the confirmation window.

Manage CAS Role Memberships

For each CAS server, be sure to designate at least one user (other than the server's process owner) to the Superuser role. In CAS Server Monitor, you can also designate users for the Data role. In the initial deployment, users that you add to the SAS Administrators [predefined custom group](#) have membership in the Superusers role. If you want to designate a user to the role without providing the extra privileges of SAS Administrators, follow these instructions.


Add or Remove CAS Role Members (in SAS Environment Manager)

- 1 In the applications menu () , select **Administration** ⇒ **Manage Environment**. In the navigation bar, select .
- 2 In the **View** field, select **Servers**.
- 3 Right-click a CAS server, and select **Assume the Superuser role**.
- 4 Right-click the server again, and select **Properties**.
- 5 In the Superuser Role Membership section of the Properties page, click .
- 6 To add a member, do the following in the Select Identities window:
 - a In the left pane, select **Users**.
 - b In the left pane, click the name of a user. The user's properties are displayed in the far right pane.
 - c Click .
- 7 To remove a member, do the following in the Select Identities window:
 - a In the **Select Identities** list, click the user that you want to remove. The identity's properties are displayed in the right pane.

Note: You cannot change or remove the account that starts the server.
 - b Click .
- 8 Click **OK**.
- 9 Click **Relinquish** in the status bar to relinquish the Superuser role.



Add or Remove CAS Role Members (in CAS Server Monitor)

The CAS (Superuser) role provides unrestricted access to all CAS objects and actions within the associated CAS server. Members of the SAS Administrators group have access to all tasks, folders, objects, and application functionality.

- 1 Sign in to [CAS Server Monitor](#) with an account that is already a CAS (Superuser).
- 2 In the left navigation bar, select .
- 3 On the Configuration page, select the **Administrators** tab.
- 4 To add a member:
 - a Click **Add**.



Note: If the **Add** button is not present, you are not signed in as a CAS administrator (Superuser).
 - b In the Add Administrator window, enter a user or group name, select the appropriate identity type, and select the **CAS** or **Data** radio button.

TIP The user and group names that you enter are not validated. You can enter any user or group name from your identity provider.

- c Click **OK** to save your changes.
- 5 To change a role assignment:
 - a Click  in the appropriate row, and select **Modify**. You cannot change the assignment for the account that starts the server.
 - b In the Edit Administrator window, select **Data** or **CAS**, and click **OK**.
- 6 To remove a role assignment, click  in the appropriate row, and select **Delete**. You cannot remove the account that starts the server.
- 7 Under **Administrators**, review the results.
- 8 Verify that full administrative privileges are available when designated users sign in to CAS Server Monitor. For example, any user who sees the **Add** button on the **Administrators** tab is a CAS administrator (Superuser).

Assume the Superuser Role

In SAS Environment Manager, you become a Superuser only after you explicitly assume that role. For example, you might assume the role to troubleshoot and resolve an access issue. To assume the Superuser role:

- 1 In the applications menu () , select **Administration** ⇌ **Manage Environment**. In the navigation bar, select .
- 2 In the **View** field, select **Servers**.
- 3 In the list of servers, right-click the name of the server for which you want to assume the role, and select **Assume the Superuser role**.
The status message reminds you that you have assumed the role.
- 4 After you perform the task that required the role, click **Relinquish** in the status bar.

Note: Use the Superuser role only when it is required for a specific task. Be sure to relinquish the role when you are finished.

Manage Access to Functionality

Access to functionality determines the features that are available to a user. Initially, all authenticated users have access to functionality that is appropriate for a typical user.

Give Users Access to Additional Functionality

To give users access to additional functionality, you should begin by simply adding selected users and groups to the appropriate predefined custom group.

For details about these groups, see [“Predefined Custom Groups” on page 485](#).

Note: To manage access to CAS administrative functionality separately, see [“CAS Server Roles” on page 493](#).

See Also




- [“Access to Functionality” on page 487](#)
- [“Initial Rules for Access to Functionality” on page 495](#)
- [General Authorization on page 107](#)

Adjust Rules for Access to Functionality

You might identify the need for more granular control, based on your organization’s use of SAS Viya. If so, here are examples of steps that you can take to adjust the level of access for a given category of users.

Restrict a Function to a Particular Group





The principal in an authorization rule is the user, group, or construct to which the rule is assigned. By default, the Authenticated Users principal (a construct that includes all authenticated users) has access to a large number of functions. If you want to restrict one or more of these functions to a particular group, follow these steps:

- 1 In the applications menu () , select **Administration** ⇒ **Manage Environment**. In the navigation bar, select .
- 2 Select **Authenticated Users** in the **Principal** drop-down box, and click **Apply**.
- 3 From the subset of rules that apply to authenticated users, find the rule that corresponds to the functionality that you want to restrict. The Object URI and Description columns provide information to help identify the rule.
- 4 Select the rule, and click .
- 5 In the Edit Rule window, select **group** in the **Principal Type** field. Then select the appropriate group or custom group in the **Principal** field.
- 6 In the **description** field, update the description for the group for which you provided access.

CAUTION! It is strongly recommended that you leave all of the other fields unchanged.
- 7 Click **Save**.
- 8 On the Rules page, right-click the rule and select **Properties**. Verify that the elements of the edited rule are as you intended.

Grant an Administrative or Specialized Function to a Different Group

By default, access to administrative or specialized functions are granted to predefined custom groups (such as SAS Administrators and Application Administrators). If you need to grant one of these functions to a different group of users, follow these steps:

- 1 In the applications menu () , select **Administration** ⇨ **Manage Environment**. In the navigation bar, select .
- 2 Clear any previously selected principals in the **Principal** drop-down box, and select **Add Identities**.
- 3 Choose **Custom Groups** from the drop-down box on the Select Identities page. Enter the ID of the predefined custom group (for example, **SASAdministrators**) in the **filter** field.
- 4 Click .
- 5 Click **OK**.
- 6 From the Rules page, select the custom group that you just added from the **Principal** drop-down box. Click **Apply**.
- 7 From the subset of rules that apply to the group, find the rule that corresponds to the functionality that you want to reassign. The Object URI and Description columns provide information to help identify the rule.
- 8 Select the rule, and click .
- 9 In the **Principal type** field, select **group**. Then select the appropriate group or custom group in the **Principal** field.
- 10 In the **description** field, update the description for the name of the group for which you provided access.
CAUTION! It is strongly recommended that you leave all of the other fields unchanged.
- 11 Click **Save**.
- 12 On the Rules page, right-click the rule and select **Properties**. Verify that the elements of the edited rule are as you intended.

See Also

- [“CLI Examples: Identities” on page 207](#)

User Management: Guidelines and Best Practices

The following basic guidelines contribute to simplicity and security:

- Limit membership in administrative roles and groups.
- Assume administrative group memberships only when you need to perform tasks that require the extra permissions.
- Assume a CAS administrative role only when you need to perform tasks that require the extra permissions, and relinquish the role when you are finished.

- If you delete a custom group, any custom rules that you created still exist. Manually delete such rules.

Identity Management Concepts

Initial Users

sasboot Account

The sasboot account is an internal user account that is created during the deployment process. The account is known only to SAS. After the deployment process is completed, use this account to log on to SAS Environment Manager to [configure the connection to your identity provider](#) and [set up the administrative users](#).

The password for the account is expired by default. Each time the SASLogon service is started, a new URL is written to the service's log which, enables the password to be reset if necessary. The URL remains active for 24 hours. For security purposes, the URL also expires after you enter it in a browser, even if the password is not reset. For details, see, see [“Sign In as the sasboot User” in SAS Viya for Linux: Deployment Guide](#).

After you have set up the identity provider connection and the first administrative users, the sasboot account is generally used only if the connection to the identity provider fails. After performing the initial tasks, you should change the password. For additional security, you can then disable the password reset feature. This prevents password reset links from being written to the log each time the SASLogon service is started. See [“Post-Installation Tasks” in SAS Viya for Linux: Deployment Guide](#).

Note:

The sasboot account exists only in a Full deployment.

Operating System Accounts

During deployment, two required accounts (one service account and one user account) and one group are created for you in the operating system, unless the accounts already exist. Because these accounts are required for running services during product operation, do not delete them or change their names. These accounts do not run as root.

The following table identifies and describes the predefined accounts:

Account Name and Group	Parameters	Purpose
sas; member of sas group	UID: 1001 Group ID: 1001 Non-login service account without user restrictions. No password. You can add a password, if needed. The password does not expire. Any post-installation changes to this account do not prevent future software updates that use SAS RPM packaging.	This user account enables the required components to run, including the web application server for SAS Studio.

Account Name and Group	Parameters	Purpose
cas; member of sas group	UID: 1002 Group ID: 1001 Typical user account that is subject to user restrictions. No default password is assigned. Important: You must set a password for this account. The password eventually expires. You are prompted to set a new password. If the CAS server is running in a grid environment (with multiple CAS worker nodes), passwordless SSH is configured by default if you used an Ansible playbook for the deployment.	Required for managing the Cloud Analytic Services. Use this user account to log on to the CAS Server Monitor.

TIP If you must log on to any of these accounts, use sudo to access them.

Identity Providers

User and group identities are stored and managed in your organization's identity provider. SAS has Read-Only access to the provider, enabling SAS to authenticate users and obtain identity information at sign-on.

Supported Identity Providers

SAS Viya supports identity providers that are based on LDAP.

Identities Service Configuration

The SAS Identities service configuration has default values appropriate for Microsoft Active Directory. To enable SAS Viya to access your identity provider, you must update the SAS Identities service configuration with the following information:

- the provider's host, port, and connection credentials. If you are using Microsoft Active Directory, this is the only information you need to change.
- mappings of your provider's identity fields to the fields used in SAS.
- information to enable searching for users and groups.

See Also

- ["Post-Installation Tasks" in SAS Viya for Linux: Deployment Guide](#)
- ["Identities Service" on page 230](#)

Identity Filtering

When [configuring the connection](#) to your identity provider, you can specify a filter to limit the identities that SAS Viya returns. For example, you can create a filter to exclude identities whose accounts are disabled or expired, or to exclude objects that represent computer resources rather than actual users or groups. You can modify this filter at any time.

If you have a large number of users, using a filter can improve performance and reduce memory requirements. In addition, user management tasks can be performed more efficiently if only relevant identities are listed in SAS Environment Manager.

A default filter is provided for sites that use Active Directory. If you use another identity provider such as openLDAP, then you might need to modify the default filter. For more information about the default filter, see [“Identities Service” on page 230](#).

Identity Caching

Identity caching is available for enhanced performance. Search requests go to the cache, reducing the number of direct requests to the identity provider. You can configure the cache refresh interval, and enable or disable the cache. The cache is enabled by default. See [“Identities Service” on page 230](#).

Custom Groups

What Is a Custom Group?

A custom group is a group that exists in SAS Viya but not in your identity provider. These groups are persisted in a SAS database.

Your deployment includes a set of [predefined custom groups](#). You can also create your own custom groups. This feature is useful if you want to create new groups of SAS users, but you do not want to (or do not have permission to) create groups in your identity provider.

Predefined Custom Groups

The following custom groups are provided with your deployment. These groups provide an easy way to give users and groups access to the appropriate data, content, or functionality.

Note: The predefined groups below are a part of a deployment that contains SAS Visual Analytics, SAS Visual Statistics, and SAS Visual Data Mining and Machine Learning. Some products and solutions have additional predefined groups. See the documentation for these products and solutions for information about other predefined groups.

For example, if you have SAS Data Studio, then you have a predefined group called Data Builders. This group is not assumable, and there are no initial members.

Note: These groups are not supplied in a programming-only deployment.

SAS Administrators

Have access to the following::

- all tasks in SAS Environment Manager and CAS Server Monitor.
- all folders and all objects that the folders contain (for example, plans and reports).

Is an assumable group.

Members of CAS Superuser role are initial members.

Note: Access to data (CAS libraries) is not included. For example, users in this group can create, run, and view reports only if they have explicitly been granted access to the underlying data.

Esri Users

Can access Esri systems for geo map access.

Is not an assumable group.

Has no initial members.

Note: Esri requires that organizations pay for tokens to use the Esri geographic mapping services. You can add a user or group of users to the Esri Users group to control who has access to these tokens. Therefore, you can control the cost of using Esri geographic services.

Application Administrators

Can access the following items from SAS Home:

- Publish Tile
- Manage Published
- SAS Theme Designer

Is not an assumable group.

Has no initial members.

Note: An additional custom group is predefined, but not created. If you create a group with ID: *CASHostAccountRequired*, members of this group automatically run their CAS sessions under their own host account. By default, CAS sessions run using the `cas` account. For more information, see [“The CASHostAccountRequired Custom Group” on page 486](#).

Assumable Custom Groups

The SAS Administrators group is a predefined custom group. This group is *assumable*. When a user in an *assumable* group signs in to SAS Viya, a prompt appears asking `Do you want to opt in to all of your assumable groups?` A list of assumable groups to which the user belongs appears below the prompt.

If the user selects **Yes**, the user gets the extra permissions that are associated with the assumable groups. If the user selects **No**, the user does not get the extra permissions. The selection remains in effect until the user signs out.

As a best practice, users should select **Yes** only when they need to perform tasks that require the extra permissions

The CASHostAccountRequired Custom Group

The *CASHostAccountRequired* custom group is predefined, but not created. If you create a group with ID: *CASHostAccountRequired*, members of this group automatically run their CAS sessions under their own host account. By default CAS sessions run using the `cas` account.

Therefore, members of this group must have host accounts.

Note: If a user is a member of the *CASHostAccountRequired* custom group, but has no host account, then SAS Environment Manager cannot access information about the CAS Server. You might observe the following behavior:

- From SAS Environment Manager, the CAS server appears to be down even though it is not. No libraries or tables are displayed.
- From SAS Data Studio, you receive a `connection refused` or `access denied` error message when you attempt to select a CAS server.

When you modify the membership of this group, the users that have been added or removed must log off from their sessions before the changes can take effect.

If a user has previously created `sashdat` files and is then added to the *CASHostAccountRequired* custom group, the user can continue to work with data in memory. However, if certain triggering events occur, such as a CAS server restart, the same user can no longer see the `sashdat` files as the location of these files is different for members of this group. Users in this situation should copy the `sashdat` files from the default location to the host CAS user path.

The original default location is: `/opt/sas/viya/config/data/cas/default/formats/casuserlibraries/username` where user name is the user's host account.

The host CAS user location is: `~/casuser/`, where the `~` represents the users home directory.

Note that files should be copied in the opposite direction for users that are removed from the `CASHostAccountRequired` group.

Additional Documentation

Here is additional documentation related to custom groups:

- [“Manage Custom Groups” on page 477](#)
- [“Access to Functionality” on page 487](#)
- [“Initial Rules for Access to Functionality” on page 495](#)

Access to Functionality

Introduction

Access to functionality determines the features that are available to a user, such as the following:

- applications that the user can access
- menu items or pages that are visible to the user after an application is opened
- media types that the user can access, and the user’s permissions for that media type

Note: Access to CAS administrative functionality is managed separately. See [“CAS Server Roles” on page 493](#).

Access to functionality (other than CAS administration) is managed by rules that target a service, a service endpoint, a media type (for example, folders or reports), or a pseudo URI. These rules are created and enforced using the general authorization model. This is the same model that is used for rules that target specific objects (for example, specific folders or reports).

SAS provides an [initial set of rules](#) to control your user’s access to functionality, including the following:

- rules that give all authenticated users access to functionality that is appropriate for a typical user. These rules are applied automatically to any user who successfully signs in.
- rules that give special categories of users access to additional functionality (for example, access to administrative functions). To apply these rules, you simply add users or groups to a [predefined custom group](#) such as SAS Administrators.

In most cases, the initial rules provide a sufficient level of control. If (after gaining experience with SAS Viya) you identify the need for more granular control, you can make adjustments to the rules’ applicability.

See Also

- [“Initial Rules for Access to Functionality” on page 495](#)
- [“Adjust Rules for Access to Functionality” on page 481](#)

Supported Adjustments to Existing Rules

CAUTION! For rules that affect access to functionality, only certain modifications to certain rules are supported. Do not modify rules that target a service, service endpoint, media type, or pseudo URI, except as specified in this topic. For instructions, see [“Adjust Rules for Access to Functionality”](#).

You can make these modifications to the following functionality rules:

- replace the principal

- update the description
- copy a rule in order to create a new rule with a changed principal.

Table A.41 Rules

Target object URI: /deviceManagement_capabilities/manageMobileDevices
 Original principal type: Group
 Original principal ID: SASAdministrators
 Original granted permissions: Delete,Read,Create,Update
 Affected functionality: Manage the mobile device blacklist, whitelist, and device access history.

Target object URI: /SASEnvironmentManager/**
 Original principal type: Group
 Original principal ID: SASAdministrators
 Original granted permissions: Read
 Affected functionality: Access all functionality in SAS Environment Manager.

Target object URI: /SASEnvironmentManager/
 Original principal type: authenticated-users
 Original principal ID:
 Original granted permissions: Read
 Affected functionality: Access SAS Environment Manager.

Target object URI: /SASEnvironmentManager/dashboard
 Original principal type: authenticated-users
 Original principal ID:
 Original granted permissions: Read
 Affected functionality: Access the **Dashboard** page in SAS Environment Manager.

Target object URI: /SASEnvironmentManager/data
 Original principal type: authenticated-users
 Original principal ID:
 Original granted permissions: Read
 Affected functionality: Access the **Data** page in SAS Environment Manager.

Target object URI: /SASEnvironmentManager/content
 Original principal type: authenticated-users
 Original principal ID:
 Original granted permissions: Read
 Affected functionality: Access the **Content** page in SAS Environment Manager.

Target object URI: /SASEnvironmentManager/scheduling
 Original principal type: authenticated-users
 Original principal ID:
 Original granted permissions: Read
 Affected functionality: Access the **Scheduling** page in SAS Environment Manager.

Target object URI: /SASMobileBI/**

Original principal type: authenticated-users

Original principal ID:

Original granted permissions: Read

Affected functionality: Access SAS Mobile BI.

Target object URI: /SASMobileBI_capabilities/cacheMobileReportData

Original principal type: authenticated-users

Original principal ID:

Original granted permissions: Read

Affected functionality: Cache mobile report data from within the SAS Mobile BI application. This is required for offline access to reports. For users who do not have this capability, report data is retained only on the device while the report is open.

Target object URI: /SASMobileBI_capabilities/exemptFromOfflineTimeLimit

Original principal type: authenticated-users

Original principal ID:

Original granted permissions: Read

Affected functionality: Provides exemption from the SAS Mobile BI offline time-out.

Target object URI: /SASMobileBI_capabilities/exemptFromPasscodeRequirements

Original principal type: authenticated-users

Original principal ID:

Original granted permissions: Read

Affected functionality: Provides exemption from the requirement to enter a passcode to access the SAS Mobile BI application.

Note:

If any of the mobile server connections require a passcode, then it will still be required to access the application. This is true even if the exemption rule is in effect. In addition, users can enable a passcode even if the exemption rule is in effect.

Target object URI: /importVASpk/**

Original principal type: authenticated-users

Original principal ID:

Original granted permissions: Delete,Read,Create

Affected functionality: Import reports.

Target object URI: /SASReportViewer

Original principal type: authenticated-users

Original principal ID:

Original granted permissions: Read

Affected functionality: Access SAS Report Viewer.

Target object URI: /SASThemeDesigner/**

Original principal type: Group

Original principal ID: ApplicationAdministrators

Original granted permissions: Read

Affected functionality: Access SAS Theme Designer.

Target object URI: /SASVisualAnalytics

Original principal type: authenticated-users

Original principal ID:

Original granted permissions: Read

Affected functionality: Access SAS Visual Analytics.

Target object URI: /SASVisualAnalytics_capabilities/buildAnalyticalModel

Original principal type: authenticated-users

Original principal ID:

Original granted permissions: Read

Affected functionality: Create and modify analytical models in SAS Visual Analytics.

Target object URI: casManagement/servers/*/caslibs/*/tables

Original principal type: authenticated-users

Original principal ID:

Original granted permissions: Create

Affected functionality: Upload data files through the casManagement service.

Target object URI: /casManagement_capabilities/importData

Original principal type: authenticated-users

Original principal ID:

Original granted permissions: Read

Affected functionality: Access the Import Data window.

Target object URI: /webDataAccess/esri/user/token

Original principal type: Group

Original principal ID: EsriUsers

Original granted permissions: Read,Create

Affected functionality: Use the Esri service.

Target object URI: /reportRenderer/reports

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Delete, Create, Read, Update, Remove

Affected functionality: Export PDF.

Target object URI: /preferences/preferences/@currentUser

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Delete, Create, Read, Update

Affected functionality: Set preferences.

Target object URI: /folders/folders/@myHistory

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Delete, Secure, Create, Read, Update, Add, Remove

Affected functionality: Access personal history folder.

Target object URI: /folders/folders/@myFavorites

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Delete, Secure, Create, Read, Update, Add, Remove

Affected functionality: Manage personal favorites folder.

Target object URI: /comments/**

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Read, Create, Delete, Update

Affected functionality: Manage comments.

Target object URI: /reportImages/jobs

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Create, Read

Affected functionality: Create jobs to obtain report images (thumbnails, section images).

Target object URI: /reportData_capabilities/exportData

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Read

Affected functionality: Export data from reports.

Target object URI: /reportData_capabilities/exportDetailData

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Read

Affected functionality: Export detail data from reports.

Target object URI: /SASVisualAnalyticsCommon_capabilities/exportImage

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Read

Affected functionality: Export report images from SAS Visual Analytics and web or mobile report viewers.

Target object URI: /SASVisualAnalyticsCommon_capabilities/shareReport

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Read

Affected functionality: Email or share reports from SAS Visual Analytics and web or mobile report viewers.

Target object URI: /maps/providers

Original principal type: SAS Administrators

Original principal ID:

Original granted permissions: Create

Affected functionality: Add custom map provider.

Target object URI: /maps/providers/*

Original principal type: SAS Administrators

Original principal ID:

Original granted permissions: Delete, Update

Affected functionality: Manage custom map providers (update, delete).

Target object URI: /reportAlerts/*

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Delete, Secure, Create, Read, Update, Add, Remove

Affected functionality: Subscribe to report alerts.

Target object URI: /webDataAccess_capabilities/facebookImport

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Read

Affected functionality: Access import Facebook data.

Target object URI: /webDataAccess_capabilities/googledriveImport

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Read

Affected functionality: Access import Google Drive data.

Target object URI: /webDataAccess_capabilities/twitterImport

Original principal type: Authenticated Users

Original principal ID:

Original granted permissions: Read

Affected functionality: Access import Twitter data.

Target object URI: /webDataAccess_capabilities/googleanalyticsImport
Original principal type: Authenticated Users
Original principal ID:
Original granted permissions: Read
Affected functionality: Access import Google Analytics data.

Target object URI: /webDataAccess_capabilities/youtubeImport
Original principal type: Authenticated Users
Original principal ID:
Original granted permissions: Read
Affected functionality: Access import YouTube data.

Target object URI: /SASDataExplorer/**
Original principal type: Authenticated Users
Original principal ID:
Original granted permissions: Update,Delete,Create,Read,Add,Remove
Affected functionality: Access SAS Data Explorer.

CAS Server Roles

Superusers have unrestricted access to CAS and are exempt from all CAS authorization requirements.

In SAS Environment Manager, the Superuser role is never initially or automatically assumed. If you are a member of a CAS server's Superuser role, you can become a Superuser by explicitly [assuming](#) the role for that server. For example, you might assume the role to troubleshoot and resolve an access issue. After the issue is resolved, you relinquish the role.

The account that starts a CAS server is automatically assigned to that server's Superuser role.

Note: The following built-ins actions for SAS Cloud Analytics Services require a user ID that can assume the Superuser role:

- addNode
- installActionSet
- refreshLicense
- removeNode
- shutdown

For more information about the built-ins actions for SAS Cloud Analytics Services, see [Builtins Action Set: Details](#).

Role	Description	Is the Role Assumable?	Initial Members
Superuser	<p>Provides unrestricted access to a CAS server. Only a Superuser can perform the following tasks:</p> <ul style="list-style-type: none"> ■ Stop the server. ■ Add and remove nodes. ■ Manage role membership. ■ See and manage the paths list. <p>The account under which a CAS server runs is an implicit member of that server's Superuser role. Make sure each CAS server has at least one other designated Superuser.</p> <p>Note:</p> <p>By default, the users that are assigned this role have unrestricted access to metadata. However, they do not have unrestricted access to data (CAS libraries). To give users with this role unrestricted access to data, you must modify access controls to explicitly grant them access.</p>	Yes	<p>SAS Administrators (in a full deployment)</p> <p>Process owner for the server</p> <p>Analytics gateway account (sas.analyticsGateway)</p>
Data	<p>Provides unrestricted access to caslibs, tables, and columns in a CAS server. Assign members to this role only if you have users who should have unrestricted access to data but should not be able to perform all administrative tasks. Not all interfaces support the Data role.</p> <p>Note:</p> <p>By default, the users that are assigned this role have unrestricted access to metadata. However, they do not have unrestricted access to data (CAS libraries). To give users with this role unrestricted access to data, you must modify access controls to explicitly grant them access.</p>	Yes	None
Action	Do not use this role. Not all interfaces support the Action role.	Yes	None

Note: The Data role provides a subset of the abilities of the Superuser role. You cannot be a member of both the Superuser role and the Data role in the same session.

See Also

Using graphical user interfaces to manage CAS server roles:

- [“Identity Management How To” on page 476](#)

Using the Access Control action set to manage CAS server roles:

- SAS Viya: System Programming Guide

Identity Management Reference

Initial Rules for Access to Functionality



SAS provides an initial set of rules to control your users' [access to functionality](#). In most cases, the initial rules provide a sufficient level of control. If necessary, you can [adjust the rules](#).

Initial Rules for All Authenticated Users

All authenticated users can initially do the following:

- access selected functions within applications. For example, they can do the following:
 - access the **Dashboard**, **Data**, and **Content** pages in SAS Environment Manager
 - access functionality in SAS Visual Analytics
- perform operations on folders and on the objects that the folders contain




To see the rules that provide this functionality, follow these steps:

- 1 In the applications menu () , select **Administration** ⇒ **Manage Environment**. In the navigation bar, select .
- 2 From the **Principal** drop-down box, choose **Authenticated Users**. Click **Apply**.

Initial Rules for Other Predefined Custom Groups

Users in other predefined custom groups can initially access selected functions within applications. For more information, see [“Predefined Custom Groups” on page 485](#).





To see the rules that provide a group's functionality, follow these steps:

- 1 In the applications menu () , select **Administration** ⇒ **Manage Environment**. In the navigation bar, select .
- 2 Uncheck any previously selected principals in the **Principal** drop-down box, and select **Add Identities**.
- 3 Choose **Custom Groups** from the drop-down box on the Select Identities page. Enter the ID of the predefined custom group (for example, `SASAdministrators`) in the filter field.
- 4 Click .
- 5 Click **OK**.
- 6 From the Rules page, select the custom group that you just added from the **Principal** drop-down box. Click **Apply**.

User Management: Interfaces

In the following table, the shaded part of each circle is an approximation of the amount of user management functionality that a particular interface exposes. The shading indicates relative coverage. The shading does not indicate alignment of functional coverage across interfaces.

Table A.42 Interfaces for User Management

	SAS Environment Manager	A graphical enterprise web application. See “Identity Management How To” .
	CAS Server Monitor	A graphical web application that is embedded in the CAS server. See “Add or Remove CAS Role Members (in CAS Server Monitor)” on page 479.
	Access Control action set	A programmatic interface for SAS (the CAS procedure), Python, R, and Lua. See Access Control Action Set .
	Command-line interface	A simple scriptable interface that provides commands for managing identities. See “CLI Examples: Identities” on page 207.

Identity Management: Troubleshooting

Cannot Sign In to SAS Studio

- Make sure the user’s account is known to the host of the SAS Studio web application. See [SAS Viya Administration: Authentication](#).
- Examine the object spawner log. See [Logging](#) on page 505.
- If users cannot make a secure connection, see [Encryption in SAS Viya](#) on page 361.

Cannot Access Cloud Analytic Services

- If the user cannot start a CAS session, make sure the user’s account meets all applicable requirements. See [SAS Viya Administration: Authentication](#).
- If an error message in the CAS log states that the user “failed mid-tier authentication”, the user’s credentials are not valid for your direct LDAP provider. See the discussion of dual authentication in [SAS Viya Administration: Authentication](#).
- Ensure that users have a host account before adding them to the CASHostAccountRequired group. A member of the CASHostAccountRequired group without a host account cannot start the necessary CAS session.

Cannot Sign In to CAS Server Monitor

- Make sure the user’s account meets all applicable requirements. See [SAS Viya Administration: Authentication](#).

- If an error message in the CAS log states that the user "failed mid-tier authentication", the user's credentials are not valid for your direct LDAP provider. See the discussion of dual authentication in [SAS Viya Administration: Authentication](#).
- If users cannot make a secure connection, see [Encryption in SAS Viya on page 361](#).

Cannot Administer SAS Home

Make sure the user is a member of the Application Administrators group (or the SAS Administrators group). See ["Predefined Custom Groups" on page 485](#).

Cannot View Users and Group Members

If you receive the following error while viewing users, groups, or their memberships from SAS Environment Manager or any other client, then a referral might have been encountered. SAS Viya does not process LDAP referrals.

Here is an example of this error message:

```
Load Users
An error occurred loading the list of users.
exception:
org.springframework.ldap.PartialResultException
Caused by: javax.naming.PartialResultException: Unprocessed Continuation
Reference(s); remaining name 'DC=COMPANY,DC=COM'
```

This occurs because LDAP is initialized based only on what the Identities service itself configures. Therefore, any environment variables that are set will not be processed. Connecting to the global catalog might be a viable solution.

Cannot Access Esri Geographic Mapping Resources

Make sure that the user is a member of the Esri Users group. Users that are members of the Esri Users group have access to tokens for which there is a fee. See ["Predefined Custom Groups" on page 485](#).

SAS Licensing

Licensing: Overview

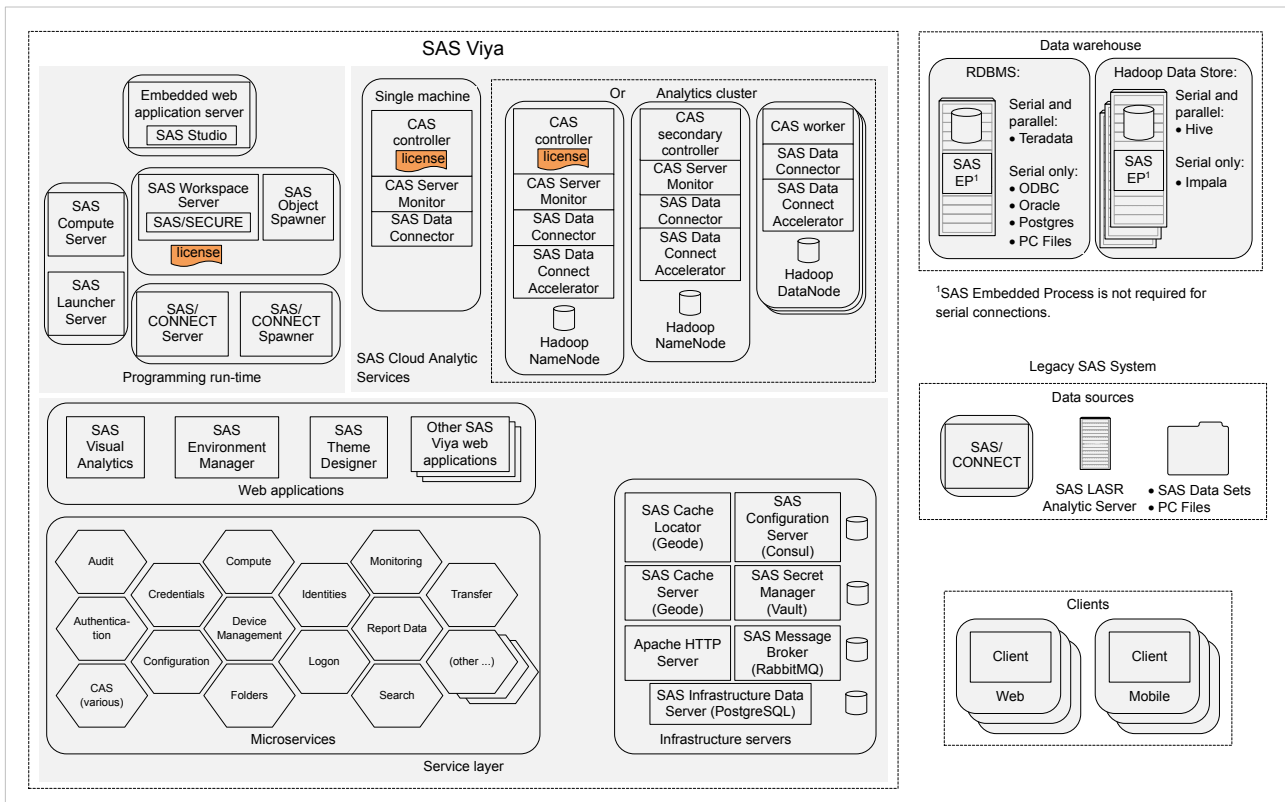
SAS Viya uses a single licensing file. Both SAS Cloud Analytic Services (CAS) and SAS Foundation use the same license.

During installation, a license is applied to both the CAS in-memory compute engine and the SAS Foundation compute engine. You apply a new license to enable new products or to extend expiration dates on existing products.

The following diagram identifies where the license file resides in SAS Viya.

For more information, see [How To on page 500](#).

Figure A.1 Where the SAS License File Resides




Licensing: How to (SAS Environment Manager)

Introduction

These instructions explain how to view product license information using [SAS Environment Manager](#).

Navigation

In the applications menu (☰), under **Administration**, select **Manage Environment**. In the navigation bar, click .

The Licensed Products page is an advanced interface. It is available to only SAS Administrators.

Licensed Products Page

Use the Licensed Products page to view and filter license status for one or more SAS products.

For each product, the following icons depict the effective license status:



The SAS license is current.



The SAS license is due for renewal (grace period).

The grace period is a predetermined range of days immediately after the license expiration date.

For example, if the expiration date is 30 June, the grace period might extend 45 days: from 1 July - 14 August.



The SAS license is about to expire (warning period).

The warning period is a predetermined range of days that follows the grace period.

For example, if the expiration date is 30 June, the warning period might extend 56 days: from 15 August - 09 October.



The SAS license has expired.

License expiration occurs immediately after the warning period ends. An expired license means that SAS does not run.

For example, if the warning period ends on 09 October, SAS stops running at 12:00 a.m. on 10 October.

Licensing: How To

View SAS Foundation License Information

- 1 Open a web browser and sign in to SAS Studio with administrator privileges.

Here is an example:

`https://mysasserver.example.com/SASStudio`

- 2 In the **Code** tab, enter the following command:


```
proc setinit; run;
```

- 3 Click .

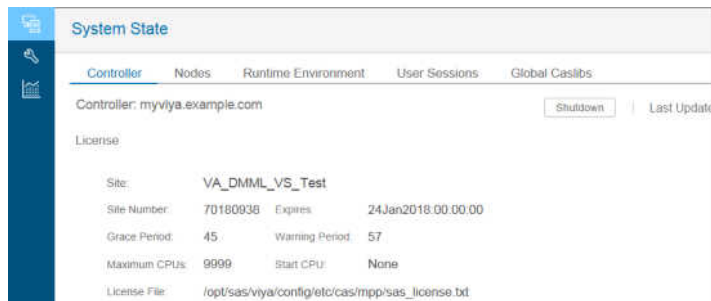
You should see output similar to the following:

```
56      proc setinit;
57
58
59      OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
Original site validation data
Current version: V.03.03M0P040416
Site name:      'smp statistics, ml, data connectors pkg chg 3.30'.
Site number:    70068118.
Expiration:     22MAY2018.
Grace Period:   45 days (ending 06JUL2018).
Warning Period: 56 days (ending 31AUG2018).
System birthday: 24MAR2016.
Operating System:  LIN X64 .
Product expiration dates:
---Base SAS Software          22MAY2018
---SAS/CONNECT                22MAY2018
.
.
.
```

View SAS Cloud Analytic Services License Information

- 1 Open a web browser and sign in to CAS Server Monitor with CAS Administrator privileges:
<https://http-proxy-machine-name/cas-tenant-name-deployment-instance-name-http>
 Here is an example:
<https://myproxy.example.com/cas-shared-default-http>
- 2 [Sign in on page 625](#) to CAS Server Monitor with a user ID that has CAS Administrator [privileges on page 493](#).
- 3 In CAS Server Monitor, beneath the **Cloud Analytic Services** banner, click .
- 4 On the System State page, select **Controller**.

You should see output similar to the following:



System State			
Controller	Nodes	Runtime Environment	User Sessions
Controller: myviya.example.com Shutdown Last Update:			
License			
Site:	VA_DMML_VS_Test		
Site Number:	70180938	Expires:	24Jan2018:00:00:00
Grace Period:	45	Warning Period:	57
Maximum CPUs:	9999	Start CPU:	None
License File:	/opt/sas/viya/config/etc/cas/mpp/sas_license.txt		

Note: The license file path depicted in the figure above is for SAS Viya running on Linux.

Locate My New License File

Your new license file resides in the same directory where you have saved and uncompressed the .tgz file, sent to you by SAS.

The license file is named: SASViyaV0300_order-number_Linux_x86-64.txt.

Here is an example: SASViyaV0300_09JNF2_Linux_x86-64.txt.

Apply New Licenses Using Ansible

You apply a new SAS license when your current license has expired, or when you are adding new SAS products to your deployment. If your deployment was performed using Ansible, you can use Ansible to apply a new license. Ansible applies your new license to the CAS controller and also to SAS Foundation.

Note: To add a new license without using Ansible, see [“Apply New Licenses Manually” on page 502](#).

- 1 Move the current license file into a backup location.
- 2 Copy the new license file into your Ansible playbook directory.

For information about where to locate your new license file and how to identify it, see [“Locate My New License File” on page 502](#).

- 3 Open `sas_viya_playbook/vars.yml`, locate `LICENSE_FILENAME`, and replace the current license filename with your new license filename.

Here is an example:

```
# The name of the license file on the Ansible machine.
LICENSE_FILENAME: "SASViyaV0300_9BLWC8_Linux_x86-64.txt"
```

- 4 Run the following Ansible command:

```
ansible-playbook apply-license.yml
```

CAS sessions created after you apply the new license automatically update with license information from the new license.

- 5 Verify that your SAS Foundation license has been renewed by following the steps in [“View SAS Foundation License Information” on page 500](#).
- 6 Verify that your SAS Cloud Analytic Services license has been renewed by following the steps in [“View SAS Cloud Analytic Services License Information” on page 501](#).

Apply New Licenses Manually

You apply a new SAS license when your current license is about to expire, or when you are adding new SAS products to your deployment. You must apply your license to the CAS controller and also to SAS Foundation.

Note: To add a new license using Ansible, see [“Apply New Licenses Using Ansible” on page 502](#).

- 1 Sign in as the SAS Installer user to the machine where SAS Foundation is deployed.
- 2 Move the current license file into a backup location.

For information about where to locate your current license file and how to identify it, see [“Locate My New License File” on page 502](#).
- 3 We recommend that you perform the following steps in case your site decides to use Ansible in the future:

- a Copy the new license file into your Ansible playbook directory.
- b Modify your Ansible playbook to point to the new license file.

Open `sas_viya_playbook/vars.yml`, locate `LICENSE_FILENAME`, and replace the current license filename with your new license filename.

Here is an example:

```
# The name of the license file on the Ansible machine.
LICENSE_FILENAME: "SASViyaV0300_09JNF2_Linux_x86-64.txt"
```

- 4 Run the following command to apply the license to your SAS Foundation software:

```
sudo su -s "/bin/sh" -c
"/opt/sas/spre/home/SASFoundation/utilities/bin/apply_license
/path/SASViyaVrelease-number_order-number_Linux_x86-64.txt" sas
```

where *path* is the location where the new license file resides.

Here is an example:

```
sudo su -s "/bin/sh" -c
"/opt/sas/spre/home/SASFoundation/utilities/bin/apply_license
/opt/sas/installfiles/SASViyaV0300_09JNF2_Linux_x86-64.txt" sas
```

You receive a message that your license has been applied.

- 5 Verify that your SAS Foundation license has been renewed by following the steps in [“View SAS Foundation License Information”](#) on page 500.
- 6 Log on as the SAS Installer user to your CAS controller machine.
- 7 Copy the new license file to your CAS controller machine.

Here is an example:

```
sudo cp /opt/sas/installfiles/SASViyaV0303_09JNF2_Linux_x86-64.txt
/opt/sas/viya/config/etc/cas/default/
```

- 8 Update the symbolic link for `/opt/sas/viya/config/etc/cas/default/sas_license.txt` to point to the new CAS license file.

Here is an example:

```
cd /opt/sas/viya/config/etc/cas/default
ln -sf SASViyaV0303_09JNF2_Linux_x86-64.txt sas_license.txt
```

CAS sessions created after you apply the new license automatically update with information from the new license file.

- 9 If you have deployed a CAS backup controller (also referred to as a secondary controller) then perform [Step 7 – Step 8](#) on your backup controller machine.
- 10 Verify that your SAS Cloud Analytic Services license has been renewed by following the steps in [“View SAS Cloud Analytic Services License Information”](#) on page 501.

Licensing: Troubleshooting

Licensed Products page cannot be viewed.

Explanation:

Users without SAS administrator privileges and intra-tenant administrators do not have access to the Licensed Products page.





Resolution:

Contact your SAS administrator.

Licensing: Interfaces

There are several interfaces that you can use to manage and to view SAS license information. The following table lists these interfaces, and the shading indicates the relative amount of SAS license administration that each covers:

Table A.43 Interfaces to SAS Viya Licensing

	Ansible	A software orchestration tool that provides the only interface for renewing a license.
	Command-line interface	(Read-Only) A command-line interface that enables you to query SAS license information.
	SAS Environment Manager	(Read-Only) A graphical enterprise web application used to view SAS license information.
	CAS Server Monitor	(Read-Only) A graphical web application that is embedded in the CAS server. Used to view CAS license information.

Logging

Logging: Overview


In SAS Viya, logs are produced not just by applications and servers, but also by each SAS Viya service. In order to manage the large number of logs and to enable you to locate messages of interest, the operations infrastructure provides functions to collect and store log messages. The `sas-watchlog` command continuously collects and sends log messages to the RabbitMQ exchange. The `sas-stream` command then pulls the messages from RabbitMQ and writes them to disk as a tab-separated value (TSV) file. Every five minutes, the `etl-driver.sas` job extracts the log messages from the TSV file and loads them into the VIYALOGS CAS-indexed search table. SAS Environment Manager uses the information in the VIYALOGS table and the VIYALOGS_SOURCES tables to display log messages and graphs that contain the frequency and trends of messages. By default, log messages that are more than three days old are removed from the VIYALOGS table. Messages are removed from the table once a day.


The SAS Environment Manager Dashboard displays a graph of the number of ERROR-level and FATAL-level messages from the current top five sources of log messages. The graph displays messages from the past 30 minutes. The graph displays a separate line for each SAS Viya application, so that you can see at a glance which applications are causing the most problems.

The Logs page in SAS Environment Manager displays detailed information about the logged messages. In addition to a chart of log messages over the past 30 minutes, the page displays the content of each message. You can display a graph of messages that are grouped by level or source, or that display a time series graph of all ERROR-level and FATAL-level messages.

Logging: How To

View Log Activity and Messages

In SAS Environment Manager, select  **Logs** from the left navigation menu to display the Logs page.

The **Messages** table displays log messages from SAS Viya components, subject to the specified filters and time constraints. By default, messages from the past 30 minutes are displayed. The table displays the first three lines of each log message. To view the full message, select the message and click .

The chart on the Logs page displays a graph of the log messages over the selected time range. The default time range is 30 minutes. You can choose from these graph types:

By Level

Number of messages grouped by logging level. Place your pointer over a bar to view the logging level and count. The information that is displayed in this chart is changed only by filtering based on a time range or message text. It does not change if you filter by **Source** or **Level**.

By Source

Number of messages grouped by source (the component or service that generated the message). Place your pointer over a bar to view the source and message count. The information that is displayed in this chart changes based on any filters that you select.

Time Series


Number of ERROR-level or FATAL-level messages, if any, for the current time period (the default is 30 minutes). Counts are displayed for the top five sources of these messages. If there were no ERROR-level or FATAL-level messages during the selected time period, the chart is replaced with the message **No information is available**. The information that is displayed in this chart can be changed by filtering based on a time range, message text, or sources.

The left side of the Logs page contains options that enable you to find specific log messages. You can filter the messages using these conditions:

- display messages from a specified time period
- display messages that contain specific messages or text in the message
- display messages that are at specified levels
- display messages from specified sources

The filters or searches that you apply affect the messages that are displayed in the **Messages** table.

Filter Log Messages

- 1 In SAS Environment Manager, select  from the left navigation menu to display the Logs page.
By default, the graph and the **Messages** table display information from the past 30 minutes of log activity.
- 2 To specify a different relative time period to use, click the **Recent log entries** radio button and then select a value (such as **Last hour** or **Last day**). By default, any log entries that are older than three days are removed from the CAS table that supplies information for this page, so selecting a value of **Last week** or **All** displays messages only from the past three days. For information about changing the length of time for which log messages are retained, see [“Modify Property Values” on page 553](#).
- 3 To specify a different time period, click the **Custom time range** radio button and select a start and end date and time. Specifying a time range that is longer than three days displays messages only from the past three days.
- 4 To display messages that contain specified text anywhere in the message, specify the text in the **Message** field.
- 5 To display messages only of a certain level, select the levels that you want to display from the choices in the **Level** area. Selecting logging levels in this area changes only the messages that are displayed, not the messages that are logged. To change the level of messages that are logged, you must change the logging level. See [“Set the Threshold Level for Logs” on page 507](#).
The numbers that are next to the entries in the **Level** list indicate the number of messages at each level.
- 6 To display messages only from a particular SAS Viya component, select the component in the **Source** area.
The numbers that are next to the entries in the **Source** list indicate the number of messages from each source, subject to any filters you have specified.
Note: The names that are used in the **Source** area are not the same as the names that are used when setting logging levels. See [“Set the Threshold Level for Logs” on page 507](#).
- 7 If you make any selections in the **Logs Filter** area (including **Time**, **Message**, **Level**, or **Source**), you must click **Apply** to apply the filters and to update the table.

- 8 The **Messages** table and the message counts that are beside the entries in the **Level** and **Sources** area change to reflect the filters that you have applied. This list provides details about what causes them to change. The **By Level** and **Time Series** charts change only if you specified a time range as a filter. The **By Source** chart changes based on any filter that you select.

Level area counts and counts on **By Level** graph

The counts change based on the selected time range and the **Message** filter.

Source area counts and counts on **By Source** graph

The counts change based on the selected time range, and the **Message** and **Level** filters.



Time Series graph

The top five sources of ERROR-level and FATAL-level messages are displayed. The messages are subject to the selected time range and **Message** filter.

Messages table



Displays the changes in the content of the table and the counts in the table header based on the selected time range, and the **Message**, **Level**, and **Source** filters.

Save the Log Messages Table

- 1 In SAS Environment Manager, select  from the left navigation menu to display the Logs page.
- 2 By default, the **Messages** table displays log messages from the previous 30 minutes from all sources and all threshold levels. Use the **Logs Filter** options to filter the messages that are displayed. See [“Filter Log Messages” on page 506](#) for details.
- 3 Click  at the top of the **Messages** table.
- 4 In the Save Table dialog box, select a library and specify a table name to which the log message table should be saved. Because of potentially sensitive data that might be contained in the log messages, you can save the log message table only to the CAS server and only to your personal library or the SystemData library. By default, the table is also loaded into memory when it is saved.

Because messages that are more than three days old are removed from the VIYALOGS table once a day, you can use this function to keep a record of log messages over time. To conserve disk space, a best practice is to filter the table for information that you are particularly interested in before saving the contents. For example, you might filter the table to display only ERROR-level messages or messages from a particular source.

Set the Threshold Level for Logs

- 1 In SAS Environment Manager, select  from the left navigation menu.
- 2 In the selection menu in the upper left, select **Definitions**.
- 3 In the list of definitions, select **logging.level**.
- 4 Select .
- 5 In the New Configuration - logging.level window, specify the following parameters:

Services

Select the services to which the logging level applies. Some services (such as SAS Environment Manager and SAS Message Broker) correspond to SAS Viya web applications. CAS Servers are listed by server name. If you leave this field blank, the logging level applies to all services.

Level

Specify the lowest level of messages that you want to be included in the service's logs. Possible values are OFF, INFO, FATAL, WARN, ERROR, DEBUG, TRACE, and ALL. See [“Logging Thresholds” on page 511](#) for detailed information.

Note: Because a logging level of TRACE generates an extremely large number of messages and can fill up available disk space, an additional configuration step is required in order to use this level of logging. Use TRACE-level logging only with the assistance of SAS Technical Support.

Name

Specify the logger name. Some microservices are associated with more than one logger. See [“Microservice and Web Application Loggers” on page 512](#) for a list of valid loggers.

6 Click **Save**.

Note: You cannot delete the configuration after you create it. You can only edit the configuration.

Logging: Troubleshooting

Why Can I Not See Logs in SAS Environment Manager?

If you are a tenant administrator, you do not have the permissions to look at logs. Your provider-level administrator can access the log messages.

Why Is the Logged Issues Chart on the Dashboard Blank?

The **Logged Issues** chart on the Dashboard and the **Time Series** graph on the Logs page display only ERROR-level and FATAL-level messages from the top five sources of these messages over a selected time range (30 minutes by default). If no ERROR-level or FATAL-level messages are received during this time period, the chart is blank and the message **No information is available** appears.

Why Do Log Messages Not Appear in SAS Environment Manager?

The charts and tables on the Logs page and the Dashboard in SAS Environment Manager use information from the VIYALOGS and VIYALOGS_SOURCES tables. If the Logs page indicates that no log messages are present (for example, the **Messages** table is blank), perform these checks:

- Verify that the selected time range is valid. By default, log messages that are more than three days old are removed from the VIYALOGS table once a day. If you specify a time range that is greater than three days ago, there will not be any messages that match the time filter.
- Verify that the CAS tables that are used for logging exist and that they are being updated. Use the Data page in SAS Environment Manager to verify that the VIYALOGS and VIYALOGS_SOURCES tables exist and that they contain data. The tables are in the SystemData library. If the tables exist and contain data, verify that the number of rows in the VIYALOGS table changes over time as new messages are added. If the tables do not exist or are not being updated, verify that the CAS server is running.

Note: The VIYALOGS and VIYALOGS_SOURCES tables are readable only by the SAS Environment Manager logging function. They cannot be read directly.

- If there is a problem with one of the logging CAS tables, use the **Availability** tile on the SAS Environment Manager Dashboard to check the status of the stream-evdm and watch-log services. Restart the services if needed.

- Verify that your SAS license is valid.

Why Are Some Messages in the Message Table Blank?

The process that is used to extract the log message information from the TSV file and to prepare the information for display in SAS Environment Manager parses the message and attempts to identify the most important part of the message. In only a few cases, none of the text of the original message remains after this parsing process has completed, so the **Message** column in the **Messages** table is blank. A blank message does not indicate a problem with SAS Viya logging.

Logging: Reference

Overview and Terminology

The SAS logging facility as used by SAS Viya is a flexible, configurable framework that you can use to collect, categorize, and filter events and to write them to a variety of output devices. The logging facility supports problem diagnosis and resolution, performance and capacity management, and auditing and regulatory compliance. The logging facility has the following features:

- Log events are categorized using a hierarchical naming system that enables you to configure logging at a broad level or a fine-grained level.
- Log events can be directed to multiple output destinations. For each output destination, you can specify the following logging facility components:
 - the categories and levels of log events to report
 - the message layout, including the types of data to be included, the order of the data, and the format of the data
 - filters based on criteria such as diagnostic levels and message content
- Logging levels can be adjusted dynamically without starting and stopping processes.

Here are the common terms that this document uses:

appender

a named entity that represents a specific output destination for messages. Destinations include fixed files, rolling files, operating system facilities, client applications, database tables, message queues, and custom Java classes. You can configure appenders by specifying thresholds, filters, log directories and filenames, pattern layouts, and other parameters that control how messages are written to the destination.

filter

a set of character strings or thresholds, or a combination of strings and thresholds that you specify. Log events are compared to the filter to determine whether they should be processed.

level

the diagnostic level that is associated with a log event. The levels, from lowest to highest, are TRACE, DEBUG, INFO, WARN, ERROR, and FATAL.

log event

an occurrence that is reported by a program for possible inclusion in a log.

logger

a named entity that identifies a message category. Loggers are named using a hierarchical system that enables you to configure logging at a broad level or a fine-grained level.

Loggers inherit settings from their higher-level (ancestor) loggers.

logging configuration

an XML file that determines how log events are processed. You use the logging configuration to assign thresholds to loggers, to configure appenders, and to specify which categories and levels of log events are to be written to each appender.

message category

a classification for messages that are produced by a SAS subsystem.

pattern layout

a template that you create to format messages. The pattern layout identifies the types of data, the order of the data, and the format of the data that is generated in a log event. It is delivered as output.

threshold

the lowest event level that is processed. Log events whose levels are below the threshold are ignored.

Understanding Loggers

A logger is a named entity that identifies a message category. A logger's attributes consist of a level and one or more appenders that process the log events for the message category. The level indicates the threshold, or lowest event level, that is processed for this message category.

Loggers are specified in log events to associate the log event with a message category. By categorizing log events, the logger can write messages of the same category to the same destinations. When a log event occurs, the log event message is processed by the appender that is associated with the logger that is named in the event log. However, the log event level must be the same or higher than the level that is specified for the logger.

Loggers are organized hierarchically and inherit the attributes of their ancestor logger. Hierarchical logger names are separated by a period (.) (for example, Admin.Meta.Security). The root logger is the highest level logger. All loggers inherit the root logger's attributes. The logging configuration file defines several message categories that are immediate descendants of the root logger. These high-level categories — Admin, App, Audit, IOM, and Perf — are used for SAS server logging and can be referenced by log events in SAS programs.

Understanding Appendors

An appender is a named entity that is referenced by a logger. An appender specifies the destination for the message and how the message is formatted. It also specifies the attributes for the appender class and provides additional filtering capabilities.

When a log event occurs, the logging facility processes the message by using the appender that is named. The appender is named in the logger's <appender-ref> element in a logging facility configuration file.

CAS Server Loggers

A logger identifies how different categories of messages are processed, including the level of messages that are processed and the appender to which messages are sent. For example, an Admin logger specifies how administration-related log messages are processed.

In the logging configuration file, a logger has this structure.

```
<logger name="logger-name">
  <level value=threshold/>
  <appender-ref ref="appender-name"?>
</logger>
```

The `level value=threshold` parameter for a logger specifies the lowest level of messages that are processed by the logger. For example, a level of WARN specifies that only messages with a level of WARN, ERROR, and FATAL are included in the log.

The `appender-ref ref=appender-name` parameter for a logger specifies the appender, or the destination, for log messages.

These loggers are defined with a default threshold of INFO in the default logging configuration file for the CAS server, located at `/opt/sas/deployment_name/config/etc/cas/default/logconfig.xml`

Admin

processes administration events. The log messages are sent to the UNIX system log.

App

processes events from applications.

App.cas.actions

processes events from CAS actions.

Audit

processes events used for auditing. These events include user authentication requests and administration of access controls.

Logging

processes events from the logging system. Log messages are sent to the UNIX system log.

These loggers are not defined by default.

App.cas

processes events from the CAS server.

App.cas.actions.actionsetname.actionname

processes events from a specified CAS action set and CAS action.

App.cas.driver

processes events from start-up of the CAS server.

App.cas.tkcasa

processes events from internal processes

App.cas.datastep

processes the output and events from DATA step PUT statements, as well as messages that are sent to the SAS log.

CAS Server Appenders

The logging configuration file for the CAS server at `/opt/sas/deployment_name/config/etc/cas/default/logconfig.xml` defines these two appenders:

RollingFileAppender

writes log messages to a time-based rolling log file. By default, the file rolls over at midnight. By default, messages from the App, App.cas.actions, and Audit loggers are sent to this appender.

UNXFacilityAppender

writes log messages to the syslogd logging facility in the UNIX operating system. It discards messages that have already been logged by the appender. By default, messages from the Admin and Logging loggers are sent to this appender.

Logging Thresholds

The logging configuration file sets a threshold for each logger. Messages at the threshold level or higher are processed by the logger, although messages lower than the threshold are ignored. The following threshold levels are available (ordered lowest to highest):

Trace

produces the most detailed log messages. This level might be useful when isolating the cause of a problem, but it produces too many messages for normal use.

Debug

produces detailed log messages, although less detailed than the Trace threshold. This level might be useful when isolating the cause of a problem, but it produces too many messages for normal use.

Info

produces messages that show an application's progress.

Warn

produces messages that identify areas of potential problems.

Error

produces messages when errors occur, although the application might continue to run.

Fatal

produces messages when severe errors occur. The application will probably end.

Microservice and Web Application Loggers

These loggers are associated with SAS Viya microservices and web applications. Specify these loggers when you are creating a logging level definition. See [“Set the Threshold Level for Logs” on page 507](#) for more information.

Service	Loggers	Usage
All	com.sas.authorization	Authorization decisions
All	com.sas.authorization.bootstrap	Authorization rule bootstrapping
All	con.sas.configuration.bootstrap	Configuration bootstrapping
All	com.sas.credentials.bootstrap	Credential domain bootstrapping
All	com.sas.event	Generated and received events
All	com.sas.folders.bootstrap	Folder definition bootstrapping
All	com.sas.security.oauth2	Authentication issues
All	com.sas.security.oauth2.bootstrap	Client token bootstrapping (allows services to talk to other services)
All	com.sas.typeregistry.bootstrap	Type definition bootstrapping
All	org.apache.http.header	Request and response headers
All	org.apache.http.wire	Full requests and responses
All	org.springframework.security	Authentication issues
appregistry	com.sas.appregistry	
appregistry	com.sas.homeshared	

Service	Loggers	Usage
audit	com.sas.audit	
authorization	com.sas.authorization	
authorization	org.springframework.security	
backup-agent	com.sas.backup.worker	
cas-access-management	com.sas.casconnection	The TRACE level logs the Lua-equivalent of all actions that are sent to the CAS server by cas-access-management.
cas-formats	com.sas.casconnection	The TRACE level logs the Lua-equivalent of all actions that are sent to the CAS server by cas-formats.
cas-management	com.sas.casmanagement	
cas-management	com.sas.casconnection	The TRACE level logs the Lua-equivalent of all actions that are sent to the CAS server by cas-management.
cas-management	com.sas.casmanagement.server	The DEBUG level shows CAS server usage by SAS Visual Analytics and SAS Visual Analytics Viewer.
cas-management	com.sas.casmanagement.session.service.CasSessionService	Tracks by user ID the creation of user CAS server sessions. Can be used to track an individual user or the log file that is processed to obtain counts of CAS user sessions.
casproxy	com.sas.casproxy	
casproxy	com.sas.casconnection	The TRACE level logs the Lua-equivalent of all actions that are sent to the CAS server by casproxy.
casrowsets	com.sas.casrowsets	
casrowsets	com.sas.casconnection	The TRACE level logs the Lua-equivalent of all actions that are sent to the CAS server by casrowsets.
collections	com.sas.collections	
collections	com.sas.homeshared	
comments	com.sas.comment	
configuration	com.sas.configuration	
credentials	com.sas.credentials	

Service	Loggers	Usage
data-preparation-plans	com.sas.data.preparation	The DEBUG level displays the contents of job requests and results.
deploymentBackup	com.sas.backup	
device-management	com.sas.devicemgmt	
files	com.sas.svcs.file	
folders	com.sas.folders	
geodelocator	com.sas.locator	
home	com.sas.home	
home	com.sas.homeshared	
identities	com.sas.identities	
identities	com.sas.identities.provider.lldap	SAS LDAP access
identities	org.springframework.security.lldap	Spring LDAP access
monitoring	com.sas.svcs.monitoring	
preferences	com.sas.preferences	
reportdata	com.sas.report.common	
reportdata	com.sas.report.bireport	
reportdata	com.sas.bicommon.export.office	
reportdata	com.sas.bidata	CAS data provider, CAS server resources, timing for CAS sessions
reportdata	com.sas.reportdata.utils.DataUtils	Cache management, CAS queries
reportdata	com.sas.reportdata	Reportdata service classes. Because it produces verbose logging, you should narrow the scope.
reportdata	com.sas.reportdata.DataServicesBase	SAS Report XML. It shows total elapsed time for report generation tasks. This logger is good for performance analysis of reports in SAS Visual Analytics Viewer. You can search the log statements by user ID.
reportdata	com.sas.reportcache	Caching of report content
reportdata	com.sas.reportcommon.utils.debug.Report	

Service	Loggers	Usage
reportdata	com.sas.cas	CAS resource utilization
reportdata	com.sas.reportcommon.utils.debug.Timer	It provides detailed, step-by-step timing data for report generation: XML parsing, CAS query, result caching. This is the best logger for detailed performance analysis. Use with <code>com.sas.reportdata.DataServicesBase</code> at the DEBUG level to compare CAS query time to total report generation time. You can search the log statements by a specific report GUID and a user ID. This logger also provides an XML report.
saslogon	com.sas.logon	It produces a high volume of messages at the DEBUG level.
saslogon	com.sas.logon.authentication	SAS authentication events
saslogon	org.cloudfoundry.identity	
saslogon	org.springframework.security	
sasthemedesigner	com.sas.themedesigner	
sasvisualanalytics	com.sas.van	
search	com.sas.svcs.search	
searchindex	com.sas.svcs.search.index	
themes	com.sas.themedesigner.service	
themes	com.sas.themedesigner.service.rest	
themes	com.sas.themedesigner.service.publish	
themes	com.sas.themedesigner.service.servlet	
themes	com.sas.themedesigner.service.persistence.model	
transfer	com.sas.transfer	
types	com.sas.typeregistry	

Monitoring

Monitoring: Overview

SAS Viya provides monitoring functions through several facilities. Use the monitoring system that matches your needs and your environment:

- The SAS Viya operations infrastructure collects metrics from SAS Viya applications and services. See [“Operations Infrastructure: Overview” on page 539](#) for more information. SAS Environment Manager uses the collected data to display metric information and status in these interfaces:
 - To quickly view the health and status of your SAS Viya environment, see [“Use the SAS Environment Manager Dashboard for System Monitoring” on page 525](#).
 - To view metrics, status, and performance charts for the machines in your environment, see [“Monitoring: How to \(SAS Environment Manager\)” on page 518](#).
 - To view detailed reports for the status and activity in your system, see [“Use SAS Environment Manager Reports for System Monitoring” on page 519](#).

If you are using the SAS Viya programming-only interface, SAS Environment Manager is not deployed.

- CAS Server Monitor is a graphical web application that is embedded in the CAS server. It provides system-level monitoring for the machines and processes running under the CAS server.

To view detailed information about the load and performance for the machines and processes running under a CAS server, see [“Monitoring: How to \(CAS Server Monitor\)” on page 526](#).

- Grid Monitor provides histograms to view CPU load, memory usage, disk usage, and network performance for each CAS node. Grid Monitor provides a greater level of detail than SAS Environment Manager or CAS Server Monitor. The information that is displayed in the application enables you to quickly identify the nodes that are overloaded compared to the other nodes in the CAS cluster. The application enables you to view detailed information about memory and disk usage and to monitor processes that run on the CAS cluster. See [“Monitoring: How to \(Grid Monitor\)” on page 529](#).
- CAS start-up or session options can enable returning of performance metric information each time a CAS action runs. The data provided by the metrics enables you to monitor the CPU load on the CAS grid and to determine how efficiently the CAS grid is processing the actions. See [“CAS Action Metrics” on page 536](#) for a list of the metrics that are returned.

The CAS options are available in the SAS Viya programming-only environment.

Monitoring: Concepts

A metric is a measurement that describes the performance of a component or a subsystem of SAS Viya. Because metrics are valuable only when they are regularly collected and evaluated, the operations infrastructure is dedicated to collecting data about the state of SAS Viya resources and services. A set of collector


components from the infrastructure then publishes the data as a message to a RabbitMQ exchange, where a publisher sends it to ETL processes and a data mart. SAS Environment Manager uses the collected data from the data mart to display in various interfaces such as reports, tables, and availability indicators. See [Operations Infrastructure on page 539](#) for more information.

In a SAS Viya environment, CAS uses a controller node to distribute work to worker nodes. In this type of distributed environment, it is important to monitor the performance of each of the nodes in the environment, to ensure that nodes are not becoming overloaded and slowing down. You should also monitor session processes on the CAS nodes to ensure that individual processes are not consuming excessive resources.

Monitoring: How to (SAS Environment Manager)





Monitor Machines

Navigation

In SAS Environment Manager, select  **Machines** from the left navigation menu to display the Machines page.

The Machines page displays a list of machines across the top of the page. An icon next to the machine name indicates the status of the machine (available, unavailable, or partially available). Select a machine from the list to display information about the machine on the charts and tables in the Machines page.

View the Status of a Machine

- 1 In SAS Environment Manager, select  from the left navigation menu to display the Machines page.
- 2 On the Machines page, select a machine name from the list at the top of the page. An icon beside the machine name indicates whether the services on the machine are available , partially available , or completely unavailable .
- 3 By default, the chart on the Machines page displays the percentage of total CPU utilization over the last hour. Click **Last hour** to change the display to the last 6, 12, or 24 hours. Place your pointer on a line on the graph to view detailed information about the CPU utilization, divided into User, System, Wait, and Stolen usage. Place your pointer in the chart and use the control wheel on your mouse to zoom in to the chart.

Note: The chart is updated every two minutes. The data that is displayed on the chart is updated every five minutes.

- 4 Click **Memory** above the chart to display the percentage of memory that is used over the selected time period. Place your pointer on a line on the graph to view detailed information about memory usage.
- 5 The **Machine Checks** table displays the results of these predefined system checks that are performed on the machine:

Disk utilization of SAS Config filesystem

The check passes if disk usage does not exceed 95%.

Memory percent free




The check passes if memory usage does not exceed 95%.

Serf Health Status

The check passes if the SAS Configuration Server is running.

The table is refreshed every 10 seconds.

- 6 The **Service Instances** table displays a list of the service instances that are running on the selected machine and the status, address, and port for each service instance. The data is refreshed every 10 seconds.


- 7 By default, the area on the right side of the Machines page displays the properties of the server. The **Properties** area displays information such as the machine address, operating system, uptime, and total memory.
- 8 To display the collected metrics for the server, click  in the toolbar on the right side of the page. The **System Metrics** area displays detailed information about memory usage and availability.
- 9 To display the SAS packages that are installed on the machine, click  in the toolbar on the right side of the page. The **SAS Packages** area displays the name and version number of the packages that are installed on the machine.
- 10 To display the system limits for the machine, click  in the toolbar on the right side of the page. The **System Limits** area displays the resource limits for users on the machine.


Use SAS Environment Manager Reports for System Monitoring


Working with System Reports

SAS Environment Manager provides a set of predefined reports that provide a view of the most important metrics for monitoring a SAS Viya deployment. The Dashboard displays a thumbnail of each report, which you can use to access the full report in SAS Report Viewer. You must be an administrator in order to view system reports.

To display the report thumbnails, on the SAS Environment Manager Dashboard, select **Show Reports**.

The report thumbnails are not live views of the full reports, but are snapshots of the report from the last time the thumbnail was generated. You must refresh the thumbnail in order to view the current state of the report. To refresh a report thumbnail, in the title bar for the report, select  and then select **Refresh**.

To open a report, in the title bar for the thumbnail report, select  and then select **Open**.

To return to SAS Environment Manager from the full view of a report, click your browser's back button or select  and select **Manage Environment**.

Monitor Application Activity

The Application Activity report provides detailed information about SAS applications and services running on your system. See [“Working with System Reports” on page 519](#) for information about accessing and opening reports.

When you open the report, the machines in your environment are listed along the top of the report. Select a machine for which to display the report.

Select the report page to view. The report pages are organized into these tabs:

Main

Displays a chart of memory usage of the 10 applications or services that are consuming the most memory. The report displays the metrics HeapUsedMax and NonHeapUsedMax.

System Session History

Displays a graph of the top 10 applications or services that have had the most active HTTP sessions over the previous eight hours.

Application History



Displays the thumbnails of detailed reports for a selected service or application. Use the menu in the upper left corner of the page to select the service or application whose reports you want to view. If you do not select a service or application, the thumbnails display aggregate data for all services and applications. Use the slider control at the top of the page to select the time range for the reports. Click  in the upper right of any

chart to view a full-size version of the chart, including legends and labels. Click  in the upper right of the full-size chart to return to the thumbnail view.

Here are the available charts:

Heap usage

Displays the amount of heap memory that is used. The chart displays the metrics HeapCommitted, HeapUsed, NonHeapCommitted, and NonHeapUsed.

HTTP sessions

Displays the number of HTTP sessions that are used. The chart displays the metrics HTTPSessionsActive and HTTPSessionsMax.

Class Usage

Displays the number of classes that are used by the application or service. The chart displays the metrics Classes, ClassesLoaded, and ClassesUnloaded.

DataSource Activity

Displays the number of data sources that are used by the application or service. The chart displays the metrics DatasourcePrimaryActive and DatasourcePrimaryUsage.

Garbage Collection Time

Displays the amount of time that is used for garbage collection. The chart displays the metrics GcPsMarkswEEPTime and GcPsScavengeTime.

Threads

Displays the number of application threads that is used. The chart displays the metrics Threads, ThreadsDaemon, and ThreadsPeak.

Uptime

Displays the amount of time that the application or service has been running.

Garbage Collection Count

Displays the number of items that are collected during garbage collection. The chart displays the metrics GcPsMarkswEEPCount and GcPsScavengeCount.

Data collection status

Displays a chart of metric data points that are collected for each application.

Monitor CAS Activity

The CAS Activity report provides detailed information about CAS. See [“Working with System Reports” on page 519](#) for information about accessing and opening reports.

When you open the report, the machines in your environment are listed along the top of the report. Select a machine for which to display the report.

Select the report page that you want to view. The report pages are organized into these tabs:



Main

Displays the **Memory Used**, **I/O**, and **Threads** charts.

CPU Load

Displays the **CPU Load** and **CPU Usage** charts. The **CPU Usage** chart displays the metrics SystemCPU and UserCPU.

System Info

Displays the thumbnails of detailed reports for the CAS servers. Use the slider control at the top of the page to select the time range for the reports. Click  in the upper right of any chart to view a full-size version of the chart, including legends and labels. Click  in the upper right of the full-size chart to return to the thumbnail view.

Here are the available charts:

- I/O Wait Time
- IRQ Time
- Open Files
- Free Memory

System Details

Displays a table of detailed metric information for the CAS servers, which are captured at one-minute intervals. The table includes data for load averages, free memory, idle time, and IRQ time.

Node Details

Displays a table of detailed information about the CPU load on the CAS server nodes, which are captured at one-minute intervals.

CAS Details

Displays a table of detailed metrics for the CAS servers, which are captured at one-minute intervals. The table includes metrics for memory used, CPU usage, and uptime.

Monitor Disk Space

The Disk Space report provides detailed information about disk space and usage. See [“Working with System Reports” on page 519](#) for information about accessing and opening reports.

When you open the report, the machines in your environment are listed along the top of the report. Select a machine for which to display the report.

Select the report page to view. The report pages are organized into these tabs:

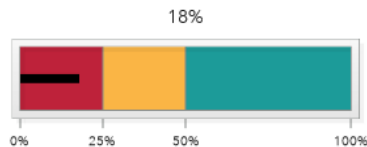
Main

Displays a chart of the top 10 filesystems on SAS Viya machines that have the least amount of free space.

Storage Dashboard

Displays a chart of the total percentage of free disk space on each machine in the system. It also displays a series of charts of the top 10 file storage locations that have the least amount of available space.

In the **Bottom 10 paths by Percent available** charts, the black line represents the available space. The background of the chart is color-coded to indicate whether the available space is in the acceptable zone (green), the warning zone (yellow), or the danger zone (red). For example, the disk corresponding to this graph has 18% free space, which is in the danger zone.



All disk usage over time

Displays a chart of the percentage of free space on all paths for each machine over the previous 48 hours.

Machine disk usage over time

Displays a chart of the total percentage of free disk space on each machine in the system. It also displays a chart of the percentage of free space on all paths for each machine over the previous 48 hours.

Disk usage forecast

Displays a chart of the percentage of free space for a selected machine and the path over the previous 24 hours. It also includes a projection of the free space that will be available over the next 48 hours. Select a machine from the list above the chart, and then select a path from the list below the machine list.

Storage Map

Displays a visual representation of the size and available free space of all disks in all machines. Each disk is represented by a color-coded block. The size of the block represents the size of the disk. The color of the

block represents the amount of free space. The color shifts from blue to red as the disk space decreases. Place your pointer on a block to view the size and percentage of free space for the disk.

Details

Displays a table of the size and free space for a selected machine and path, and that is recorded at one-minute intervals. Select a machine from the list above the table, and then select a path from the list below the machine list.

When monitoring of CAS disk usage, understand that owned disk space is the space used by files that are created in SAS_DISK_CACHE directories from in-memory blocks. These files cannot be shared with other server processes or session processes. Shared disk space is the space that is used by existing SASHDAT files from a co-located data source (PATH, HDFS, or DNFS). These files can be shared with other server processes or session processes.

Monitor SAS Infrastructure Data Server Tables

The Infrastructure Data Server Tables report provides detailed information about the table size and usage on the SAS Infrastructure Data Server. See [“Working with System Reports” on page 519](#) for information about accessing and opening reports.

Select the report page to view. The report pages are organized into these tabs:

Main

Displays a chart of the five largest tables in the SAS Infrastructure Data Server. The chart displays the metrics TableSize Max, IndexSize Max, and ToastSize Max for each table.

Table Usage Trend

Displays a graph of the total size of all SAS Infrastructure Data Server tables over the past 36 hours. The chart separately displays the metrics TableSize, IndexSize, and ToastSize for all tables.

Application Usage History

Displays an animated chart of the size of the largest SAS Infrastructure Data Server tables over the previous 36 hours. Click ► below the chart to start the animation. The chart displays the size of the tables at the time indicated on the slider control below the chart. You can use the slider control to view the size of the tables at a selected time. The chart separately displays the metrics TableSize, IndexSize, and ToastSize for each table.

Table Size History

Displays an animated chart of the size of the largest SAS Infrastructure Data Server tables over the previous five hours. Click ► below the chart to start the animation. The chart displays the size of the tables at the time indicated on the slider control below the chart. You can use the slider control to view the size of the tables at a selected time. The chart separately displays the metrics TableSize, IndexSize, and ToastSize for each table.

Monitor Message Queue Activity

The Message Queue Activity report provides detailed information about traffic and activity on the RabbitMQ message queues that used by the operations infrastructure to provide log messages, metric data, notifications, and alerts to consumers such as SAS Environment Manager. See [“Working with System Reports” on page 519](#) for information about accessing and opening reports.

Select the report page to view. The report pages are organized into these tabs:

Main

Displays a chart of the total amount of data that is published to and from each message queue. The chart displays the PublishIn Max and PublishOut Max metrics for each message queue.

Cumulative traffic

Displays a graph of the amount of data that is published to a selected message queue over the previous 48 hours. Select the queue name from the list at the top of the chart. The chart displays the PublishIn and PublishOut metrics.

Activity over time

Displays an animated chart of the amount of data that is published to message queues over the previous 36 hours. Click ► below the chart to start the animation. The chart displays the amount of data that is published to the queue at the time indicated on the slider control below the chart. You can use the slider control to view the amount of data that is published at a selected time.

System info

Displays charts illustrating the number of RunQueue instructions, the amount of data written to queues, and the amount of memory that is used over the previous 48 hours. The charts display the RunQueue, IoWriteBytes, and MemUsed metrics.

Monitor System Activity

The System Activity report provides detailed information about CPU usage, memory usage, and network activity. See [“Working with System Reports” on page 519](#) for information about accessing and opening reports.

When you open the report, the machines in your environment are listed along the top of the report. Select a machine to display the report for that machine.

Select the report page to view. The report pages are organized into these tabs:

Main

Displays charts of the load average and memory usage for a selected machine over a selected time range. Select the machine from the list at the top of the chart. Select the time range using the slider control at the top of the chart.

CPU history

Displays a chart of the CPU usage for a selected machine over a selected time range. Select the machine from the list at the top of the chart. Select the time range using the slider control at the top of the chart. The chart displays separate lines for the metrics System CPU% and User CPU%.

Memory Usage history

Displays a chart of the free memory and the used memory for a selected machine over a selected time range. The orange area at the top of the chart represents the free memory, and the green area at the bottom of the chart represents the used memory. The two values together always add up to the total memory. Select the machine from the list at the top of the chart. Select the time range using the slider control at the top of the chart.

Network Activity history

Displays charts of the network activity and the cumulative network I/O for a selected machine and an interface over a selected time range. Select the machine and the interface from the lists at the top of the chart. Select the time range using the slider control at the top of the chart. The Network Activity over time chart displays the TransmitBytes and ReceiveBytes metrics. The Cumulative Network I/O chart displays the TransmitBytes_cnt and ReceiveBytes_cnt metrics.

Memory Animation

Displays an animated chart of the used memory and the free memory for all machines over the previous 36 hours. Click ► below the chart to start the animation. The chart displays the memory usage at the time indicated on the slider control below the chart. You can use the slider control to view the memory usage at a selected time. The chart separately displays the metrics Used Memory and Free Memory for each machine. The orange area at the top of the chart represents the free memory, and the green area at the bottom of the chart represents the used memory.

CPU Details Animation

Displays an animated chart of the CPU usage for all machines over the previous 36 hours. Click ► below the chart to start the animation. The chart displays the CPU usage at the time indicated on the slider control below the chart. You can use the slider control to view the CPU usage at a selected time. The chart separately displays the metrics UserCPU, WaitCPU, SystemCPU, and StolenCPU for each machine.

Network Activity Animation

Displays an animated chart of the network activity for all machines over the previous 36 hours. Click ► below the chart to start the animation. The chart displays the network activity at the time indicated on the slider control below the chart. You can use the slider control to view the activity at a selected time. The chart separately displays the metrics `TransmitBytes_cnt` and `ReceiveBytes_cnt` for each machine.

System Details

Displays a table of detailed system metrics for selected machines over a selected time period, which is captured at one-minute intervals. The table includes information about memory usage, CPU usage, and system load. Select a machine from the list at the top of the table. Select a time period by using the slider control at the top of the table.

Network Details

Displays a table of detailed network metrics for the selected machines and the interfaces over a selected time period, which is captured at one-minute intervals. The table includes information about received data, transmitted data, and transmit errors. Select a machine and an interface from the lists at the top of the table. Select a time period by using the slider control at the top of the table.

Monitor User Activity

The User Activity report provides a view of audit information. See [“Working with System Reports” on page 519](#) for information about accessing and opening reports.

Select the report page to view. The report pages are organized into these tabs:

Main

Contains thumbnail graphs for the charts **Most active users**, **Activity counts**, **Most active data**, and **User activity**.

Most Active Users

Displays the **Most Active Users** and **Activity Over Time** charts, and a table of the audit records that are ordered by level of user activity. The table does not display audit records from SAS internal users. Select a bar in the **Most Active Users** chart to display the **Activity Over Time** chart for the selected user, and to list the audit records only for the selected user.

Application Usage

Displays the **Most used Applications** and **Application Activity** charts, and a table of the audit records that are ordered by level of application activity. Select a bar in the **Most used Applications** chart to display the **Application Activity** chart for the selected application, and to list the audit records only for the selected application.

Report Activity

Displays the **Top Report Usage** chart and a table of the audit records for report access. By default, the chart and the table display activity for all users. To view report usage and the audit records only for a specific user, select the user in the **Users** menu.

Data Plan Activity

Displays the **Top Report Usage** chart and a table of the audit records for data plan access. By default, the chart and the table display activity for all users. To view data plan usage and the audit records only for a specific user, select the user in the **Users** menu.

Data Activity

Displays the **Top Report Usage** chart and a table of the audit records for data table access. By default, the chart and the table display data table activity for all users. To view data table activity usage and the audit records only for a specific user, select the user in the **Users** menu.

Failures

Displays the **Failed Requests per Application** chart and the **Failed Activities** chart, and a table of the audit records only for failed requests. By default, the **Failed Activities** chart and the audit records table display failures for all applications. To view the **Failed Activities** chart and the audit records for a specific application, select the application's bar in the **Failed Requests per Application** chart.

Details

Displays a table of audit records. By default, the table displays all audit records. To filter the table, use the menus at the top of the table to display only those records that match your selected criteria. You can filter by user, application, action, and state. You can also filter using multiple criteria.

Note: Note: If the User Activity report is blank or displays the message `Cannot find the requested data source`, you must verify that the command-line interface (CLI) was deployed properly in your SAS Viya environment. See [“Edit the Inventory File” in SAS Viya for Linux: Deployment Guide](#) for more information.

Use the SAS Environment Manager Dashboard for System Monitoring

Monitor Availability of Machines and Services

The **Availability** tile displays grids of color-coded boxes, and each box displays the status of each machine, service, and service instance. A green box indicates that the item is available, a yellow box indicates that it is partially available, and a red box indicates that it is unavailable. The tile is updated every 10 seconds.

Selecting a box on one of the grids highlights the corresponding boxes on the other two grids. The box that you select is outlined with a solid line, and the associated boxes are outlined with a dashed line. Here are the associations between the selected boxes:

- When you click a box on the **Machines** grid, the services and the service instances that are running on that machine are highlighted on the **Services** grid and on the **Service Instance** grid.
- When you click a box on the **Services** grid, the machines on which that service is running are highlighted on the **Machines** grid, and the instances of the service are highlighted on the **Service instances** grid.
- When you click a box on the **Service instances** grid, the machines on which the service instance is running are highlighted on the **Machines** grid, and the service is highlighted on the **Services** grid.

Note: To deselect a box, hold down the Ctrl key and click the box. You can also hold down the Ctrl key and press the spacebar.

Place your cursor over a box to view the name of the machine, the service, or the service instance.

Double-click a box on the **Machine** grid to open the Services dialog box, which lists the services that are running on that machine and their availability. You can open the Machines page for the selected machine from this dialog box.

Click a box on the **Service instances** grid to view the machine address and the port where the instance is running.

Use the **Filter** field to display only certain machines, services, and service instances. As you enter characters in the **Filter** field, the boxes that are displayed in the Availability area dynamically change. The boxes that are displayed either match the characters that you type, or are associated with the boxes that are displayed. For example, entering `laun` in the **Filter** field might cause two **Services** boxes to be displayed (for the Launcher service and the Launcher server), only the **Service instance** boxes that are associated with the displayed services, and only the **Machines** boxes that are associated with the displayed services.

Evaluate CAS Nodes

The **CAS System Health** tile is used to display a pair of graphs that provide a quick view of the nodes that are either registered as a controller or are a worker node for the selected CAS server. Use the buttons at the top of the tile to select the graph that you want to view.

If your environment contains more than one CAS server, a menu above the graph enables you to select the server to view. When you display the dashboard, this functionality behind the tile attempts to connect to the default CAS server. If the default server cannot be found, the tile displays information for the first server to which it can connect. If it can connect to the default server, but the server does not respond within five seconds, the tile

displays a message. You can then retry the server or choose another server. You define the default server in the **default** ⇒ **casServer** property. This property is one of the `sas.casmanagement.global` properties for the CAS Management service. See [“Introduction” on page 215](#) for information about setting this property.

Here are the graphs displayed in the **CAS System Health** tile:

CPU Load

Displays a graph of the 1-minute CPU load average over the past five minutes for each node in your CAS cluster. The chart is updated every 10 seconds. Each node is represented by a separate line on the graph. The vertical scale of the graph changes, depending on the largest value that is displayed in the chart. Position your cursor over a line in the chart to identify both the node and the specific CPU load average value.

Node Memory Usage

Displays a bar chart, which displays the memory usage for each node in your CAS cluster. Each bar represents a separate node. Place your pointer over a bar on the graph to view the name of the node and its memory usage. The vertical scale of the graph changes to match the memory usage of the most heavily used node. The chart is updated every 10 seconds.

Monitoring: How to (CAS Server Monitor)

Access CAS Server Monitor

To log on to CAS Server Monitor, open a web browser and enter the following URL in the address field:

`https://http-proxy-machine-name/cas-tenant-name-deployment-instance-name-http`

You must have an active CAS Server session in order to access CAS Server Monitor.

For more information, see [“Using CAS Server Monitor” on page 625](#).

Monitor CAS Process Performance

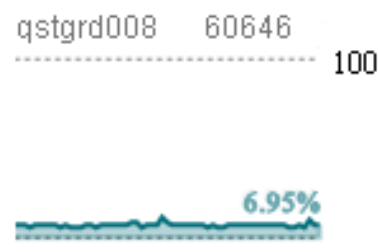
The CAS processes you can monitor with these steps correspond to SAS server processes. You can separately monitor each session that is started from the CAS server.

- 1 In CAS Server Monitor, beneath the Cloud Analytic Services banner, click .

- 2 Select **Add View** ⇒ **CAS Process CPU Usage**.

The **Process CPU Usage** panel on the window displays a set of histograms. There is one histogram for each machine and the corresponding CAS server process. The histogram in the upper left is the CAS controller node. If you are not an administrator, only the histogram for the CAS controller node is displayed.

Each histogram displays the percentage of CPU usage, from 0 to 100%.

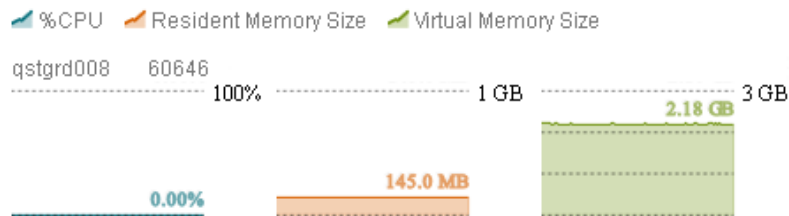




Use these histograms to note patterns of CPU usage among the CAS nodes.

3 Select **Add View** ⇒ **CAS Process Metrics**

The **CAS Process Metrics** panel on the window displays a set of histograms. There is one set of three histograms for each machine and the corresponding CAS server process. If you are not an administrator, only the set of histograms for the CAS controller node is displayed.

Each set of histograms displays the percentage of CPU used, amount of resident memory used, and amount of virtual memory used for the CAS process.



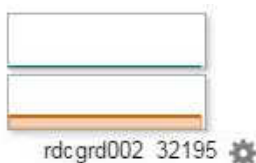
4 Click  if you want to stop metric collection. Click  to resume collection.

Monitor CPU Usage for a Session

1 In CAS Server Monitor, select  on the left side of the window.

2 Select **Add Session View** and select a session.

The panel for the session displays a set of histograms, with one histogram for each machine in the grid. If you are not an administrator, only the histogram for the CAS controller node is displayed. The top half of the histogram displays the percentage of CPU load used by the session, and the bottom displays the amount of resident memory used for the session.



Monitor Host Performance

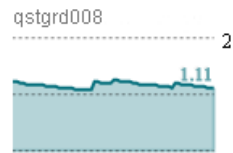
CAS Server Monitor displays histograms that enable you to view the CPU load and memory usage for all machines in your CAS server. Follow these steps:

1 In CAS Server Monitor, select  on the left side of the window.

2 To view the CPU load, select **Add View** ⇒ **Host CPU Load Average**.

The **Host CPU Load Average** panel on the window displays a set of histograms. There is one histogram for each machine in the CAS grid. If you are not an administrator, only the histogram for the CAS controller node is displayed.

Each histogram displays the CPU load on the machine, using the same format as the Linux `xload` command. Each division on the histograms represents one load average point. The highest point on each histogram is displayed to the right of the histogram.



Use these histograms to note usage patterns among the CAS nodes. For example, if you notice that the load on a worker node machine is significantly and consistently higher than the load on other machines, you can use the **Show Processes** function to check for other running processes or defunct processes. See [“Monitor Process Information” on page 528](#) for instructions on this function.



- 3 To view the memory usage, select **Add View** ⇒ **Host Memory Usage**.

The **Host Memory Usage** panel on the window displays a set of histograms. There is one histogram for each machine in the CAS grid. If you are not an administrator, only the histogram for the CAS controller node is displayed.





Each histogram displays the percentage of memory used on the machine, from 0 to 100%. The percentage of memory used is displayed in green, at the top of the histogram. The percentage of virtual memory used is displayed in orange, at the bottom of the histogram.



Use these histograms to note patterns of memory usage among the CAS nodes. For example, if the memory usage is consistently high on a machine, its memory might need to be increased.

- 4 Click  if you want to stop metric collection. Click  to resume collection.





Monitor Process Information

- 1 Perform one of these actions in CAS Server Monitor:
 - Select  on the left side of the window and open one of the views from the **Add View** or **Add Session View** menus. Click  to the right of a histogram. Select **Show Processes**. This option is available only if you are an administrator.
 - Click  and select the **Nodes** tab. Click  on the right side of a node's row and select **Show Processes**.
- 2 The Processes window appears. The window displays this information:
 - Metrics for the selected node, including uptime, number of processes, memory usage, CPU load, and file usage
 - A histogram of the CPU load for the node
 - A table containing the output from the `top` command for the selected node. The output includes metrics such as CPU usage, time, and threads for each process. If you are the process owner, the window displays information about all processes. If you are not the process owner, you can view information about your own processes.

If you are the process owner, you can open a terminal window to terminate processes that are causing problems. See [“Open a Terminal Window on a Node” on page 529](#) for information.



Open a Terminal Window on a Node

After using the monitoring functions of CAS Server Monitor to identify problems with CAS nodes, you might want to issue commands to end processes on a node. If you are the process owner, you can launch a terminal window to manage processes on the node. Follow these steps.

- 1 Perform one of these actions in CAS Server Monitor:
 - Select  on the left side of the window and use the **Add View** menu to display the **Host CPU Load Average**, **Host Memory Usage**, **CAS Process CPU Usage**, or **CAS Process Metrics** views. Click  on the right side of the histogram for a node. Select **Launch Terminal**. This option is available only if you are an administrator.
 - Click  and select the **Nodes** tab. Click  on the right side of a node's row and select **Launch Terminal**.
- 2 A terminal window appears on the selected machine. Use the window to manage processes on the machine.
- 3 Type `exit` to close the terminal window.

Change the Monitoring Display Options

When you are viewing the histograms in the **Grid Monitor** view in CAS Server Monitor, you can control how the histograms are displayed.

- To change how quickly the graph data is refreshed, move the slider next to the **Speed** label.
- To change the size of the histograms, move the slider next to the **Size** label.
- The default layout for a histogram view is a grid. To change to a single column, click the **column icon**  in the banner for a view. To return to a grid layout, click the **grid icon** .

To change the default view for the **Grid Monitor** view, select *userid* ⇒ **Settings** in the upper right of the CAS Server Monitor window. You can select a default monitor view and layout.

Monitoring: How to (Grid Monitor)

Start Grid Monitor

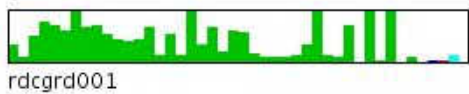
To start the Grid Monitor application, on the controller machine, run the script `/opt/sas/viya/home/SASFoundation/utilities/bin/gridmon.sh`. You must have authorization to log on to the controller machine, and you must have passwordless SSH for the host account that you use to log on.

Monitor Host Performance

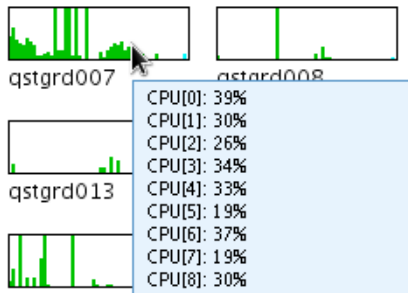
The stand-alone Grid Monitor application displays histograms that enable you to view the CPU load, memory usage, and network performance for all machines on your CAS grid. You can also view the last 60 seconds of metric data that was collected for all machines or for a single machine. Follow these steps:

- 1 Start the Grid Monitor application. See [“Monitoring: Overview” on page 517](#) for information.

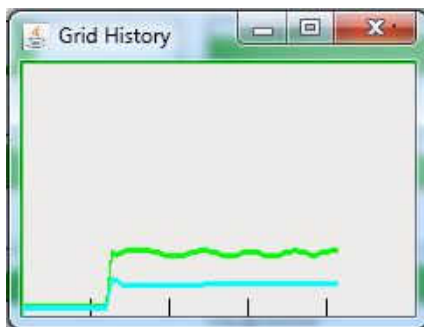
- 2 The Grid Monitor window displays a set of histograms. There is one histogram for each machine on the grid. The histogram displays values for CPU usage (green bars, one for each CPU on the machine), network read speed (dark blue bar), network write speed (red bar), and memory usage (light blue bar).



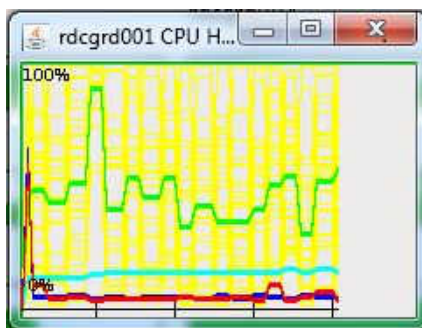
Place your pointer over a node name to view the metric data that is represented by the bars in the chart. The metric data includes CPU usage values for each core in the machine.



- 3 To view average CPU and memory use for all nodes on the grid, select **Menu** ⇒ **Show Grid History**. A chart appears that shows the average CPU usage (marked by a green line) and the memory usage (marked by a blue line) for the last 60 seconds across all nodes on the grid.



- 4 To view metric data for a particular machine, right-click the histogram in the main Grid Monitor window and select **Show History Graph**. The chart that appears displays the average CPU usage for all the cores in the machine (green line), the CPU usage for each core (yellow lines), the percentage of memory used (light blue line), the percentage of maximum network read speed (dark blue line), and the percentage of maximum network write speed (red line). The histogram displays data for the last 60 seconds.



Monitor Process Information

- 1 In the Grid Monitor window, select **Menu** ⇒ **Show Jobs on Grid**.

CPU	Memory
5%	1.9 Tb / 6.8 Tb
0.2%	3.3 Gb
0.2%	3.3 Gb
0.2%	3.3 Gb
0.0%	41.3 Gb

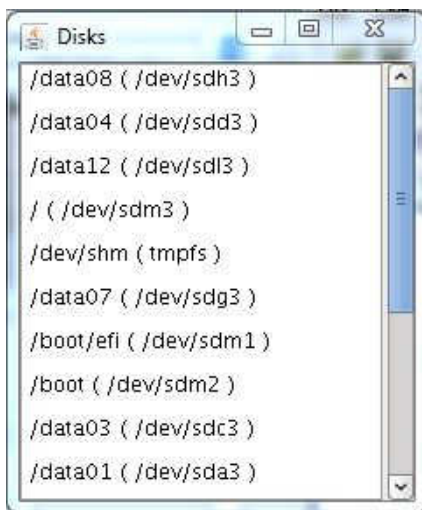
Use these values to evaluate the total load on your system and the need for additional memory or disk capacity.

- To evaluate memory usage for a particular session, locate a server process. Server processes contain a value in the **Port** column.
- Note the value of the **Shared Disk** column. This value represents the space used by existing SASHDAT files from a co-located data source (PATH, HDFS, or DNFS). These files can be shared with other server processes or session processes. As other processes compete for memory, these tables are paged from disk to memory and then back from memory to disk. A high rate of paging can degrade performance.
- Compare the values in the **Memory** and **Shared Disk** columns. If the **Shared Disk** value is lower than the **Memory** value, it indicates that sufficient memory is available for both the processes and the shared tables. In this case, performance problems are not caused by paging.

CPU	Memory	Time	Ranks	Port	Active	Pending	Completed	Owned Disk	Shared Disk
5%	2.0 Tb / 6.8 Tb								
0.0%	35.1 Gb	10:30	28	7330	1	0	9290	23.1 Gb	8.6 Gb
0.0%	18.5 Gb	0:00	28	taskToFs(10000)	0	0	42	0.0 Mb	0.0 Mb

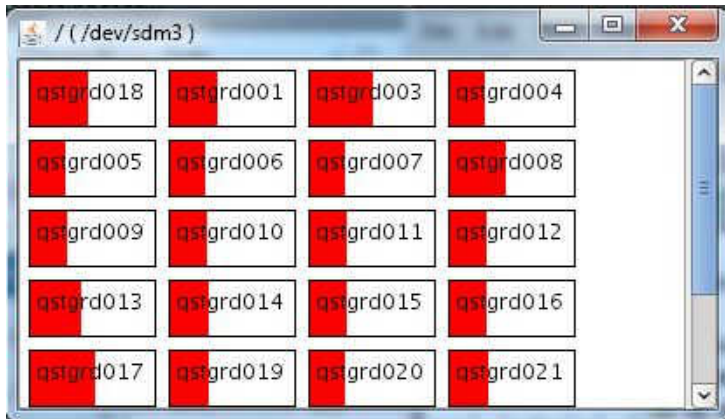
Monitor Disk Usage

- In the Grid Monitor window, select **Menu** ⇒ **Show Disks**.
- The Disks window appears. This window lists the disks used by your CAS environment. It is important that you know which file systems (and devices) are used for the CAS_DISK_CACHE directories. You should monitor these CAS_DISK_CACHE directories to ensure that there is enough room for the in-memory blocks that are written to them.

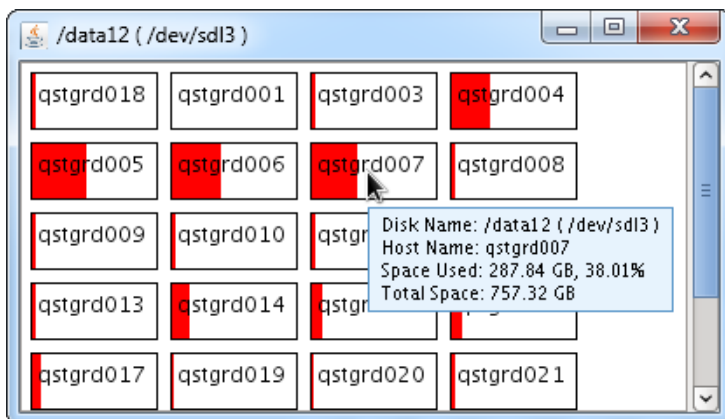


- To view usage information for a disk on each machine in your CAS cluster, click the disk name in the Disks window.

- 4 A window with the name of the disk is displayed. This window displays a histogram for the disk usage on each node in the CAS cluster.



- 5 To view detailed metrics for disk use on a node, position your cursor on a histogram for a CAS node. The information includes the total space available on the disk and the space used by the selected node.

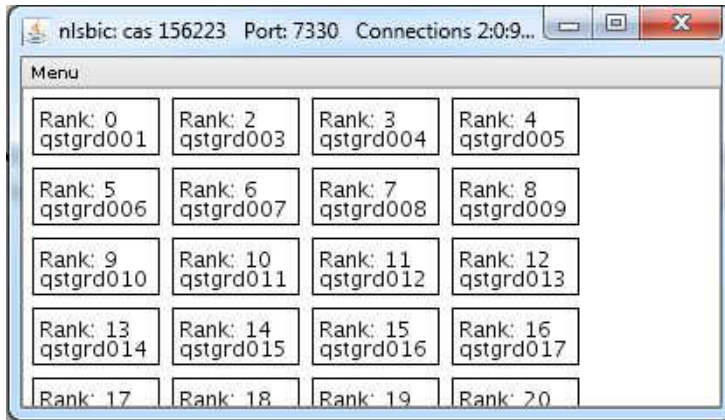


Monitor Ranks

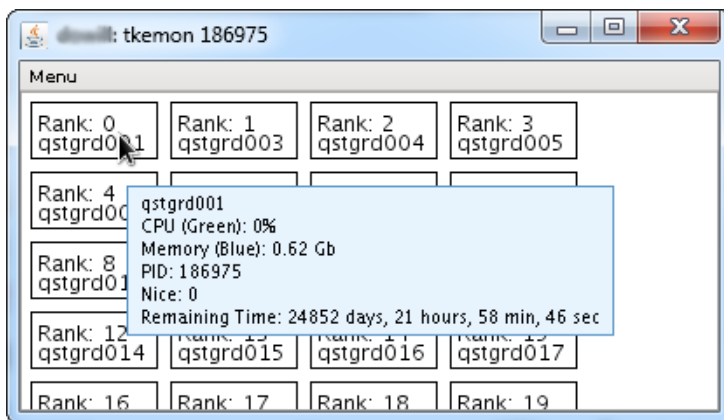
When a server starts, a software process starts on each machine in the cluster. Each process is assigned a rank. You can monitor the processes for a server across all machines in the cluster, or for all the processes that are running on a specific machine.

To monitor processes for a server across all machines, follow these steps:

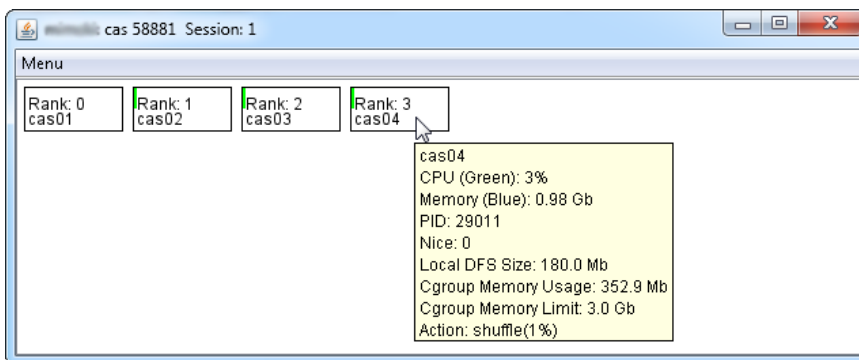
- 1 From the Grid Monitor window, select **Menu** ⇒ **Show Jobs on Grid** to display the Jobs window.
- 2 From the Jobs window, right-click a process and select **Show Ranks** from the pop-up menu to display the ranks for the selected session.



- Place your pointer over a rank to display the CPU usage, memory usage, PID, and nice value for the rank on the machine.



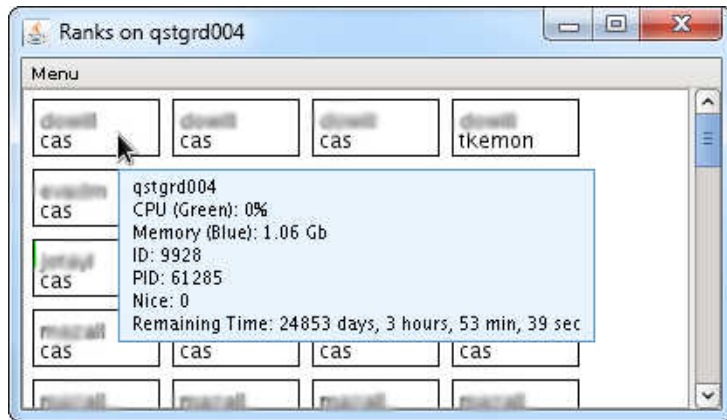
- If you are using cgroups, the information displayed also includes the memory usage and memory limit for the cgroup.



The **Cgroup Memory Limit** value specifies the physical memory limit on a host. For distributed servers, the limit applies to each host. Although more than the limit can be used through memory mapping, only physical memory up to the limit is used. The limit applies cumulatively to all sessions that are started on the server instance.

To monitor the processes on a single machine, follow these steps:

- From the Grid Monitor window, right-click the histogram for a machine and select **Show Ranks on Node** from the pop-up menu. The Ranks on machine_name window appears.
- Place your pointer over a rank to display the CPU usage, memory usage, ID, PID, and nice value for the rank.



Monitoring: How to (CAS Options)

View Performance Metrics for a CAS Action

To view metric performance data when you execute a CAS action, start the CAS server with the `-metrics` option, or set the `cas.metrics` configuration option to `true`.

To start displaying performance metrics for a running server, set the `metrics` session option to `true`.

If you enable metric collection, a standard set of metric data is returned to the log each time that a CAS action completes. The same data is displayed by both the server and the client, although the names of the metrics are different. See “CAS Action Metrics” on page 536 for a list of the metrics that are displayed.

Here is an example of the metrics that are displayed for a CAS action:

```
NOTE: Executing action 'tkimstat.summary'
NOTE: Action 'tkimstat.summary' used (Total process time):
NOTE:     real time           0.024989 seconds
NOTE:     cpu time            0.165974 seconds (664.19%)
NOTE:     total nodes        4 (96 cores)
NOTE:     total memory       377.85G
NOTE:     memory             22.53M (0.01%)
{
  disposition = { ... },
  messages = { ... },
  results = { ... },
  performance = {
    elapsedTime = 0.024989,
    cpuUserTime = 0.132979,
    systemCores = 96,
    systemTotalMemory = 405711519744,
    cpuSystemTime = 0.032995,
    memoryOS = 45793280,
    memory = 23621664,
    memoryQuota = 47366144,
    systemNodes = 4,
```


Evaluate CPU Utilization for an Action

If you specify that performance metrics are collected when CAS actions are executed, you can use these metrics to evaluate the utilization of your CAS environment.

The server metric CPU time is displayed in both the number of seconds and a percentage. Here is an example:

```
cpu time    0.165974 seconds (664.19%)
```

The percentage is calculated as $(\text{cpuUserTime} + \text{cpuSystemTime}) / \text{elapsedTime}$. On a single-threaded system, the maximum value for this metric is 100%. However, for a multi-core system, the maximum value is $100\% * \text{number of cores}$. In this example, the system has 96 cores, so the maximum value is 9600%.

Monitoring: Troubleshooting

Why Can I Not See Machine Information in SAS Environment Manager?

If you are a tenant administrator, you do not have the permissions to look at machine health information or metric data. Your provider-level administrator can access this information.

Monitoring: Reference

CAS Action Metrics

If you enable metric collection for CAS actions, a standard set of metric data is returned each time that a CAS action completes. The same data is displayed by both the server and the client. Here is the data that is displayed:

Server Metric Name	Client Metric Name	Description
real time	elapsedTime	The number of seconds in actual time required to run the action.
	cpuUserTime	The total number of seconds taken by the action in user mode across all cores that were used to run the action.
	cpuSystemTime	The total number of seconds taken by the action in system mode across all cores that were used to run the action.

Server Metric Name	Client Metric Name	Description
cpu time		<p>CPU time is measured and displayed in these formats:</p> <ul style="list-style-type: none"> ■ <code>cpuUserTime + cpuSystemTime</code>, displayed in seconds. ■ $(\text{cpuUserTime} + \text{cpuSystemTime}) / \text{elapsedTime}$, displayed as a percentage.
total nodes	<code>systemNodes</code>	The number of nodes in the cluster (total nodes display both <code>systemNodes</code> and <code>systemCores</code>).
total nodes	<code>systemCores</code>	The number of cores in the cluster (total nodes display both <code>systemNodes</code> and <code>systemCores</code>).
total memory	<code>systemTotalMemory</code>	The total memory available to the system. Total memory is displayed in GB, and <code>systemTotalMemory</code> is displayed in bytes.
memory	<code>memory</code>	Memory used to execute the action.
	<code>memoryOS</code>	Operating system used by the action.
	<code>contextVoluntary</code>	The number of times a context switch occurred because a process relinquished its processor before its time slice had been completely used.
	<code>contextInvoluntary</code>	The number of times a context switch occurred because a higher priority process was present or because the current process exceeded its time slice.
	<code>memoryQuota</code>	The memory quota used by the action.
	<code>dataMovementTime</code>	The amount of time, in seconds, taken by the data that moved between the memory and the processors.
	<code>dataMovementBytes</code>	The number of bytes of data that moved between the memory and the processors.

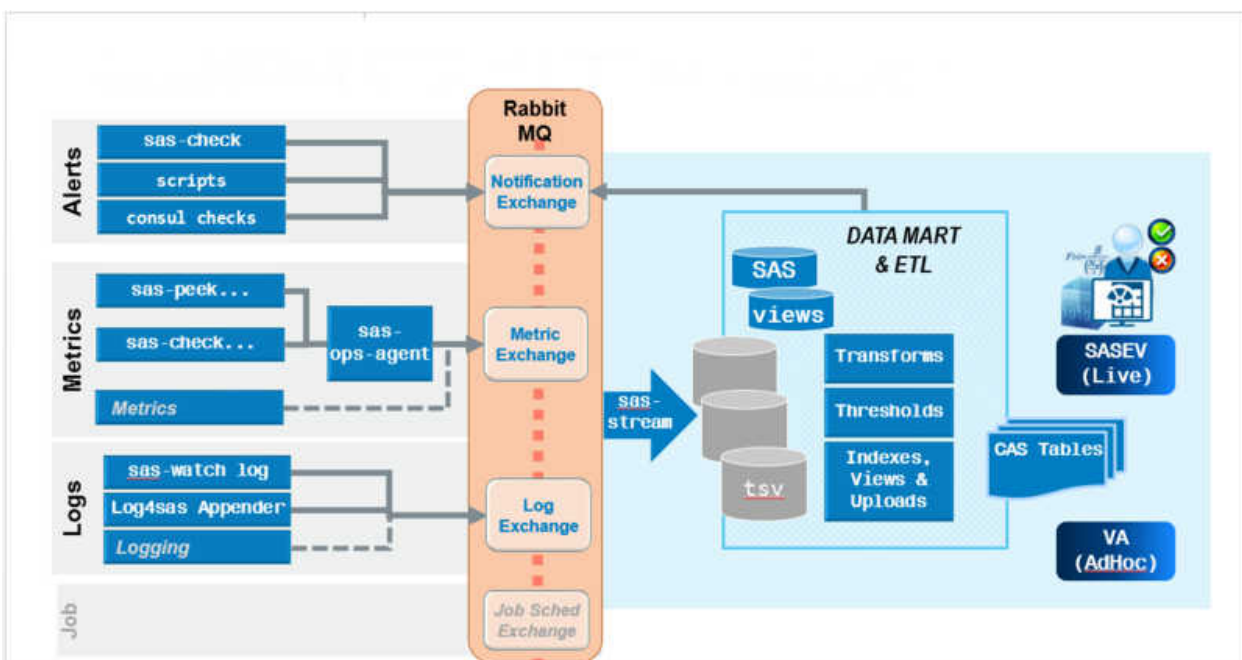
See [“View Performance Metrics for a CAS Action” on page 535](#) for information about displaying these metrics.

Operations Infrastructure

Operations Infrastructure: Overview

The operations infrastructure implements an event-driven architecture that underlies several areas of SAS Viya administration, most notably monitoring and logging. An event in this architecture represents some unit of information, such as a metric reading (for example, CPU usage at a particular time) or a log message (for example, a microservice has failed). The architecture is flexible and adaptable, because it keeps the producer of the events and the consumers of the events separate. The producer of an event collects information such as a system metric or a log message that publishes that information to a message exchange without knowledge about the consumer of the information. The information consumer looks for specified types of information and retrieves the information when it is found. Types of consumers include extract, transform, and load (ETL) processes or a data mart. Likewise, the consumer has no knowledge about the source of the information.

The operations infrastructure implements the producer portion of this architecture by using components such as `sas-peek`, `sas-check`, and `sas-watch` to collect system performance data from resources and log files. These producers then publish events to the appropriate RabbitMQ message exchange. Another component, `sas-stream`, performs the consumer role by reading events from the message exchange and writing them to the data mart. An ETL process runs periodically and loads data from the data mart into the CAS server, where it is then surfaced in SAS Environment Manager. For example, both the Logging view and the Machines view of SAS Environment Manager surface data that is collected through the operations infrastructure.



The infrastructure manages these event types:

Metrics events

measurements describing the performance of a resource, component, or subsystem

Check events

simple threshold checks of a metric, with return codes to indicate whether the metric passed the threshold

Log message events

messages written to a log that is monitored by the infrastructure

Notification events

important events that require attention

The `sas-peek` component is a collector that gathers performance metric data for a specific resource or service. For example, `sas-peek cpu` collects CPU metrics, and `sas-peek cas` collects metrics for the CAS service.

The `sas-watch` component monitors an object such as a log. For example, `sas-watch log` monitors the state of the log files for SAS Viya resources and services.

The `sas-check` component captures metric data and compares it to a defined threshold value. Then the component reports whether the metric value passed or exceeded the warning or critical threshold.

You do not have to run these components (`sas-peek`, `sas-watch`, and `sas-check`) manually. They are configured and run automatically.

How To: Operations Infrastructure Command Line

Overview

Operations infrastructure tasks are performed through the `sas-ops` command. Although you can run these commands manually, they are configured and run automatically as part of the operations infrastructure processes. The functions that are provided by this command follow:

- streaming of operations information, including notifications, alerts, metric data, and log messages
- validation of the SAS Viya environment and the operations infrastructure
- information about the SAS Viya environment, including the services, the machines, and the environment

Here is the format of the command:

```
sas-ops command --option
```

You must be the SAS install user in order to run the command.

Stream the Operations Information

Stream the Alert Messages

Use the `sas-ops alerts` command to stream the alert messages from SAS applications and components.

The default behavior is to stream alert messages until you stop the command. You can use the option `--timeout duration` to stream messages only for a specified time. The duration is specified using the format `0h0m0s0ms`, although subsets of this format are also allowed. See [“Time Format” on page 554](#) for details and examples. For example, `--timeout 5m30s` specifies that alert messages are streamed for 5 minutes and 30 seconds.

Use the `--last n` option to display only a specified number (specified as *n*) of the most recent messages.

Use the `--format` option to specify the format for the data.

- `--format json`
Streams the alerts in JSON format, with the data and parameters on one line.
- `--format pretty`
Streams the alerts in JSON format, with the data and parameters on separate lines.
- `--format line`
Streams the alerts on a single line. The timestamp is listed first.
- `--format block`
- `--format message`

Stream the Log Events

Use the `sas-ops logs` command to stream the log messages that are generated by SAS Viya applications and services.

The default behavior is to stream log messages until you stop the command. You can use the option `--timeout duration` to stream messages only for a specified time. The duration is specified using the format `0h0m0s0ms`, although subsets of this format are also allowed. See [“Time Format” on page 554](#) for details and examples. For example, `--timeout 5m30s` specifies that messages are streamed for 5 minutes and 30 seconds.

The default behavior is to stream messages to the terminal on which the command was issued. However, you can use the `--format` option to specify a different format for the messages.

- `--format json`
Streams the log messages in JSON format, with the message text and parameters on one line.
- `--format pretty`
Streams the log messages in JSON format, with the message text and parameters on separate lines.
- `--format line`
Streams the log message text and parameters on a single line. The timestamp is listed first.
- `--format file`
Streams the log message text and parameters on a single line.
- `--format term`
Streams the log message text and parameters in terminal format. The message level is listed first.
- `--format plain`
Streams the log message text and parameters with no tagging to indicate different parts of the message. The abbreviated message level is listed first.
- `--format logfmt`
Identifies the parts of the message using the format `message-part=string`. An example is `timestamp=2017-10-20T11:14:56.000000-04:00`. The messages are color-coded, depending on their level.
- `--format template`
Streams the log message text and parameters with no tagging to indicate different parts of the message.
- `--format event`
Streams the log messages in JSON format as used by the event service.

Because many log messages are produced in a typical environment, you can use these options to filter the message stream to include only those messages of interest.

- `--match regular-expression`
Streams the messages that match the specified regular expression.

- `--match-file file`
Streams the messages that match the regular expressions that are contained in the specified file.
- `--min-level level`
Streams the messages that are at the specified level or a higher level.
Valid values for *level* are trace, debug, info, warn, error, fatal, and none.
- `--source source`
Streams the messages only from the specified source.

Stream the Metric Data

Use the `sas-ops metrics` command to stream the metric data that is generated by SAS Viya applications and services.

The default behavior is to stream data until you stop the command. You can use the option `--timeout duration` to stream data only for a specified time. The duration is specified using the format `0h0m0s0ms`, although subsets of this format are also allowed. See [“Time Format” on page 554](#) for details and examples. For example, `--timeout 5m30s` specifies that data is streamed for 5 minutes and 30 seconds.

The default behavior is to stream metric data to the terminal on which the command was issued. However, you can use the `--format` option to specify a different format for the data.

- `--format json`
Streams the metric data in JSON format, with the data and parameters on one line.
- `--format pretty`
Streams the metric data in JSON format, with the data and parameters on separate lines.
- `--format line`
Streams the metric data on a single line. The timestamp is listed first.
- `--format property`
Streams the metric data on a single line.
- `--format event`
Streams the metric data in JSON format as used by the event service.

Stream the Notification Messages

Use the `sas-ops notifications` command to stream notification messages from SAS applications and components.

The default behavior is to stream notifications until you stop the command. You can use the option `--timeout duration` to stream notifications only for a specified time. The duration is specified using the format `0h0m0s0ms`, although subsets of this format are also allowed. See [“Time Format” on page 554](#) for details and examples. For example, `--timeout 5m30s` specifies that notifications are streamed for 5 minutes and 30 seconds.

Publish a Notification Message

Use the `sas-ops notify message` command to publish a notification message. You can use the `--level` option to specify the level of the message. Supported level values are info, warn, or alert.

An example command is `sas-ops notify The server will be rebooted. --level alert`.

Verify the Status of Your SAS Viya Environment

The Operations infrastructure provides the `validate` command to enable you to perform checks on your Viya environment in order to locate problems. To use the command, SAS Viya does not have to be running. However, the SAS Configuration Server and RabbitMQ must be running.

The syntax of this command is

```
sas-ops validate --level level --json --tags string --verbose.
```

The value for *level* specifies the complexity of the validation checks. Each level performs the checks both at its own level and the previous level. For example, specifying `--level 2` causes both the level 1 and level 2 checks to be performed.

Three levels of validation are available, in increasing order of complexity.

- 1
 - Verifies a connection to the SAS Configuration Server (Consul) and to the RabbitMQ exchanges `sas.application`, `sas.log`, `sas.metric`, and `sas.notification`. This level of validation ensures that you can perform validation checks at levels 2 and 3.
- 2
 - Verifies that the operation infrastructure is operating properly. These checks are performed:
 - Verifies the SAS Configuration Server (Consul) services on each machine. The checks verify the following: the disk space used is not greater than 95%, the memory used is not greater than 95%, and the SAS Configuration Server is running.
 - Verifies that the operations data mart ETL is running properly by performing checks on the standard regularly scheduled ETL jobs. The information displayed is the same information that is generated by the `sas-ops datamarts` command.
 - Verifies the status of the operations services, including all instances of the `ops-agent`, `alert-track`, `watch-log`, `ops-agentsrv`, and `stream-evdm` services.
- 3
 - Verifies the status of CAS, the HTTP service, and the authorization service. These checks are performed:
 - Verifies the status of the CAS servers by locating the defined servers and verifying that they are running.
 - Verifies HTTP connectivity by attempting to connect to the base HTTP address.
 - Verifies that the authorization service is working by attempting to obtain an OAuth token for the `sas-ops` command.

Display the Environment Information

Display Information about the Data Mart

Use the `sas-ops datamart` command to display metric and status information about the data mart. Here is typical information that is returned:

```
evdm
  status
    etl_driver
      casLogLoad_SYSCCRC : 0
      casLogSearchLoad_SYSCCRC : 4
      casLogconnect_SYSCCRC : 0
      casMetricConnect_SYSCCRC : 0
      casMetricLoad_SYSCCRC : 0
      endtime : 2017-10-18T14:11:29.72-04:00
```

```

    jobExitRC : 0
    osLogRC : 1
    osMetricRC : 0
    readMetricTransform_SYSCCRC : 0
    starttime : 2017-10-18T14:11:05.881639-04:00
    status : ok
    statusRC : 4
  rolloff
    endtime : 2017-10-18T02:00:10.69-04:00
    jobExitRC : 42
    starttime : 2017-10-18T02:00:00.034636-04:00
    status : error
  ziptsv
    deleteOldZips_SYSCCRC : 0
    endtime : 2017-10-18T03:01:27.89-04:00
    jobExitRC : 0
    osZipTSVRC : 0
    starttime : 2017-10-18T03:00:00.023883-04:00
    status : ok
    statusRC : 0
    updateInventory_SYSCCRC : 0
    zipTSV_SYSCCRC : 0

```

The results include information about the following three jobs that are used by the data mart:

- The `etl_driver` job, which processes the metric data and the log data, and loads the data into the data mart

`casLogLoad_SYSCCRC`

Return code from SAS for loading the log data into CAS for the log phase of the ETL job

`casLogSearchLoad_SYSCCRC`

Return code from SAS for updating the CAS search index for the log phase of the ETL job

`casLogconnect_SYSCCRC`

Return code from SAS for connecting to CAS for the log phase of the ETL job

`casMetricConnect_SYSCCRC`

Return code from SAS for connecting to CAS for the metric phase of the ETL job

`casMetricLoad_SYSCCRC`

Return code from SAS for loading the metric data into CAS for the metric phase of the ETL job

`endtime`

End time of the ETL job

`jobExitRC`

Return code from the operating system that is written by `emi-runsasjob`

`osLogRC`

Maximum return code (either 0, 1, or 2) from SAS for the log phase

`osMetricRC`

Maximum return code (either 0, 1, or 2) from SAS for the metric phase

`readMetricTransform_SYSCCRC`

Maximum return code from SAS for reading the raw TSV file for the metric phase

`starttime`

Start time of the ETL job

`status`

Status message (ok, error, or warning) that is written by `emi-runsasjob` at the end of the job

statusRC

Maximum return code from SAS

- Nightly rolloff job, which removes old data mart data from CAS

endtime

End time of the rolloff job

jobExitRC

Return code from the operating system that is written by emi-runsasjob

starttime

Start time of the rolloff job

status

Status message (ok, error, or warning) that is written by emi-runsasjob at the end of the job

- Nightly ZIPTSVC job, which archives old TSV files into ZIP format, removes old archive files, and updates the data mart inventory

deleteOldZips_SYSCCRC

Return code from SAS for deleting old zipped TSV files

endtime

End time of the ZIPTSVC job

jobExitRC

Return code from the operating system, written by emi-runsasjob

osZipTSVRC

Maximum return code (either 0,1, or 2) from SAS for archiving the TSV files

starttime

Start time of the ZIPTSVC job

status

Status message (ok, error, or warning) that is written by emi-runsasjob at the end of the job

statusRC

Maximum return code from SAS

updateInventory_SYSCCRC

Maximum return code from SAS for updating all inventory files (one per resource type)

zipTSV_SYSCCRC

Maximum return code from SAS to Zip TSVs and delete old ZIP files, written by SAS

Access Information about Your Environment

Use the `sas-ops env` command to display information about the machine (on which you run the command), SAS environment variables, and the SAS Viya deployment.

Here is typical information that is returned:

Host Information:

```
Full hostname           : full_hostname
Short hostname          : short_hostname
Consul node name        : full_hostname
```

SAS environment variables:

```
CONSUL_HTTP_ADDR = https://localhost:8501
```

SAS Viya Deployment:

```
Install user           : sas
```

```

Deployment ID      : viya
SAS root          : /opt/sas
Deployment root    : /opt/sas/viya
Home directory     : /opt/sas/viya/home
Config directory   : /opt/sas/viya/config
Log directory      : /opt/sas/viya/config/var/log
SPRE directory     : /opt/sas/spre

```

View Machine Information

Use the `sas-ops info` command to obtain information about each machine in your SAS Viya environment. For each machine in your environment, the command returns this information:

- machine identity
- packages installed on the machine
- system metrics
- system limits

The information returned by this command is the same information that is displayed on the Machines page in SAS Environment Manager.

View Information about Services

Use the `sas-ops services` command to view information about the services in your environment.

Run the command `sas-ops services` with no options to display a list of all SAS Viya services that are currently active in your environment.

Run the command `sas-ops services --detail service-name` to view detailed information about a specified service. Here is typical information that is returned:

```

{
  "ID": "e77ab2dc-b6c6-4a4d-af4e-bf3712de3c98",
  "Node": "vdmml-tue-17w47-ud.uda.sashq-r.openstack.sas.com",
  "Address": "10.104.29.192",
  "Datacenter": "",
  "TaggedAddresses": {
    "lan": "10.104.29.192",
    "wan": "10.104.29.192"
  },
  "NodeMeta": {},
  "ServiceID": "compute-10-104-29-192",
  "ServiceName": "compute",
  "ServiceAddress": "vdmml-tue-17w47-ud.uda.sashq-r.openstack.sas.com",
  "ServiceTags": [
    "proxy",
    "rest-commons",
    "https",
    "jobExecution-provider",
    "jobExecution-provider-Compute",
    "dataSources-provider",
    "dataSources-provider-Compute",
    "dataTables-provider",
    "dataTables-provider-Compute",
    "rowSets-provider",
    "rowSets-provider-Compute",
    "contextPath=/compute"
  ]
}

```

```

],
"ServicePort": 39504,
"ServiceEnableTagOverride": false,
"CreateIndex": 16797,
"ModifyIndex": 16797
}

```

Run the command `sas-ops services --health service-name` to perform the health checks on each instance of the specified service:

Disk utilization of SAS Config filesystem

The check passes if disk usage does not exceed 95%.

Memory percent free

The check passes if memory usage does not exceed 95%.

Serf Health Status

The check passes if the SAS Configuration service is running.

Service '*service-name*' check

The check passes if the service is running,

View Information about Metric Tasks

The operations agent (`sas-ops-agent`) runs a defined set of tasks to collect system metrics and to publish the metric data to RabbitMQ. Use the `sas-ops tasks` command to view a list of the tasks that are performed by the agent and the frequency of the task that is run. For more information about the agent and the tasks performed by the agent, see [“How To: Operations Infrastructure Agent Command Line” on page 548](#).

Here is an example of the information that is returned by the `sas-ops tasks` command:

Task Name	Description	Frequency
-----	-----	-----
CASMetrics	CAS performance metrics (level=2)	1m0s
CheckCpu	Check CPU activity less than 95% busy	1m0s
CheckFileSystem	Check file system space less than 90% used	1m0s
CheckMemory	Check memory less than 95% used	1m0s
Emisweeper	Retry publishing any payloads that failed to publish earlier	1h0m0s
FileSystemMetrics	Host file system metrics (level=2)	1m0s
HostEnvSnapshot	Host environment snapshot	02:25
LogFileArchive	Archive daily	04:00
NetworkInterfaceMetrics	Host network interface metrics (level=2)	1m0s
OpsAgentActivity	Internal sas-ops-agent activity monitor	2m0s
OpsAgentTaskStatistics	Internal sas-ops-agent task statistics activity monitor	4m0s
PostgresMetrics	Postgres metrics (level=2)	1m0s
RabbitmqMetrics	RabbitMQ performance metrics (level=2)	1m0s
SpringBootMetrics	Spring Boot performance metrics (level=2)	1m0s
SpringBootMetricsLevel3	Spring Boot performance metrics (level=3)	4h0m0s
SystemMetrics	Host system metrics (level=2)	1m0s
TopProcessMetrics	Top CPU process consumers (level=2)	1m0s
genAudit	Extract audit records. Generate a CSV files for given applications	2h0m0s
registerOpsAgentServiceTask	Register Ops-Agent service task	5m0s
registerOpsServiceTask	Register Ops service task	5m0s

How To: Operations Infrastructure Agent Command Line

Overview

The operations infrastructure agent runs a defined set of tasks. Each task is defined as a combination of a command to execute and information about how to publish the output of the command. Most tasks invoke the `sas-peek` or `sas-check` components and publish the output as an event to RabbitMQ.

The list of tasks for the agent to run are provided from the SAS Infrastructure Data Server or from a file to be read. The task definition also includes other attributes such as how often the task should be run.

Summary of the Default Task List

The following tasks are provided by default on the agent server:

Task name	Command	Description
CASMetrics	<code>sas-peek cas -level 2</code>	Collects the CAS performance metrics (at level 2) every minute.
CheckCPU	<code>sas-check cpu -warning 95 -metric percentCpuBusy</code>	Checks the CPU activity and verifies that it is lower than 95% busy. It runs every minute.
CheckFileSystem	<code>sas-check filesystem -warning 90 -metric percentUsedBytes -inctype xfs</code>	Checks the file system and verifies that it is less than 90% used. It runs every minute.
CheckMemory	<code>sas-check memory -warning 95 -metric percentUsed</code>	Checks the system memory and verifies that it is less than 95% used. It runs every minute.
EmiSweeper (server)	<code>emi-event-sweep run -delete -verbose</code>	Attempts to publish any component outputs that were not published. It runs every hour.
FileSystemMetrics	<code>sas-peek filesystem -level 2</code>	Collects the host file system metrics (level 2). It runs every minute.
EvdmDatamartEtl (server)	<code>emi-runsasjob -pgm etl_driver.sas -datamart evdm</code>	Specifies the ETL driver for the data mart. It runs every 5 minutes.
EvdmDatamartRollOff (server)	<code>emi-runsasjob -pgm rolloff.sas -datamart evdm</code>	Rolls off old data from the data mart. It runs at 2 AM every day.
evdmDatamartzipTSV (server)	<code>emi-runsasjob -pgm ziptsv.sas -datamart evdm</code>	Zips data from the data mart to a TSV file. It runs at 3 AM every day.

Task name	Command	Description
HostEnvSnapshot	<code>emi-sysinfo</code>	Takes a snapshot of the host environment at 2:25 AM every day.
LogFileArchive	<code>sas-archive</code>	Archives the logs from the previous day. It runs at 4 AM every day.
NetworkInterfaceMetrics	<code>sas-peek network -level 2</code>	Collects the host network interface metrics. It runs every minute.
OpsAgentActivity (server)	<code>sas-event-pub-exchange sas.metric</code>	Monitors the activity of the operations infrastructure agent. It runs every 2 minutes.
OpsAgentTaskStatistics (server)	<code>sas-event-pub-exchange sas.metric</code>	Monitors the activity of the operations infrastructure agent task statistics. It runs every 4 minutes.
PostgresMetrics	<code>sas-peek postgres -level 2</code>	Collects the metrics from the Infrastructure Data Server (level 2). It runs every minute.
RabbitmqMetrics	<code>sas-peek rabbitmq -level 2</code>	Collects the metrics from RabbitMQ (level 2), It runs every minute.
SpringBootMetrics	<code>sas-peek springboot -level 2</code>	Collects the metrics from Spring Boot (level 2). It runs every minute.
SpringBootMetricsLevel3	<code>sas-peek springboot -level 3</code>	Collects the metrics from Spring Boot (level 3). It runs every minute.
SystemMetrics	<code>sas-peek system -level 2</code>	Collects the host system metrics (level 2). It runs every minute.
TopProcessMetrics	<code>sas-peek top -level 2</code>	Collects the top consumers of CPU processes (level 2). It runs every minute.
registerOpsAgentSvrServiceTask (server)	<code>emi-util register -id sas.ops-agentsrv</code>	Registers the ops-agentSvr service task. It runs every 5 minutes.
genAudit	<code>genAudit.sh -a reports,folders,dataPlans,casManagement,casAccessManagement,--user-id -l 1000 -d 7</code>	Extracts the audit records for reports, folders, data plans, CAS management, and CAS access management, and generates a CSV file. It runs every 2 hours.

Controlling theTasks

List the Tasks

To list the tasks that are in the current task list and that are loaded to the SAS Configuration Server, run the command `sas-ops-agent list`. The command returns a list of the tasks that are in the current list. The command also displays the frequency and a brief description for each task. Here is typical output:

```
Task name: CASMetrics
```

```

    Freq.....: 1m0s
    Description..: CAS performance metrics (level=2)
Task name: CheckCpu
    Freq.....: 1m0s
    Description..: Check CPU activity less than 95% busy
Task name: CheckFileSystem
    Freq.....: 1m0s
    Description..: Check file system space less than 90% used
Task name: CheckMemory
    Freq.....: 1m0s
    Description..: Check memory less than 95% used
Task name: EmiSweeper
    Freq.....: 1h0m0s
    Description..: Retry publishing any payloads that failed to publish earlier
Task name: FileSystemMetrics
    Freq.....: 1m0s
    Description..: Host file system metrics (level=2)
Task name: HostEnvSnapshot
    Freq.....: 02:25
    Description..: Host environment snapshot

```

Add a Task

You can define a new task to include in a task list. Each task is specified by a task attribute list, which is a part of a task list. Here is the format of a task attribute list:

```

{
  "version": 1,
  "taskName": "TaskExample",
  "description": "Non-persistent task",
  "hostType": "any",
  "runType": "once",
  "frequency": "0s",
  "maxRunTime": "2m0s",
  "timeOutAction": "cancel",
  "errorAction": "cancel",
  "command": "emi-stress",
  "commandArgs": "-version",
  "commandType": "sas",
  "publisherType": "amqp",
  "publisherCommand": "sas-event-pub",
  "publisherArgs": "-exchange sas.metric"
}

```

Here are the attributes to specify when defining a task:

version

The version of the task attribute list.

taskName

A name that you assign to the task. The name must be a single string, and it must start with an alphabetic character or an underline (), followed by one or more alphanumeric characters.

description

(Optional) A description of the task.

hostType

The operating system on which the task can run. Here are the valid values:

linux

Specifies that the task can run only on a 64-bit Linux system.

windows

Specifies that the task can run only on a 64-bit Windows system.

any

Specifies that the task can run on either a Linux or Windows system. If you specify "hostType": "any", you must use a command name that does not have an extension.

Note: If you do not specify a value for this attribute, the value is the current operating system.

runType

How often the task runs. Here are the valid values:

once

Specifies that the command for the task is run only one time, after the agent server starts.

time

Specifies that the command runs at the time that is provided in the frequency attribute.

start_time

Specifies that the command starts when the agent server starts, and also runs at the time that is provided in the frequency attribute.

periodic

Specifies that the command runs according to the time interval that is specified in the frequency attribute. This is the default value.

periodic_aligned

Specifies that the command runs according to the time interval that is specified in the frequency attribute. The interval starts at midnight.

frequency

When the command in the task should run. See [“Time Format” on page 554](#) for details about specifying times. Here are the valid values:

0h0m0s0ms

Specifies a time interval.

YYYY-MM-DDTHH:MM:SS.ssssssZhh:mm

Specifies a specific time.

maxRunTime

How long to wait after the task starts before taking the action that is specified in the timeOutAction attribute.

timeOutAction

The action to take if the task times out (after the interval that is specified in the maxRunTime attribute). Here are the valid values:

cancel

Specifies that the task processes are killed and the task is not rescheduled. This is the default.

quiesce

Specifies that the task is completed, but it is not scheduled to run again until the agent server is restarted.

restart

Specifies that the task processes are killed and then the task is restarted.

errorAction

The action to take if the task completes with a nonzero return code. Here are the valid values:

cancel

Specifies that the task is not rescheduled. This is the default.

restart

Specifies that the task restarts at the next scheduled time.

command

The command to run. Do not include a path or command arguments. If you specify `windows` for the `hostType` attribute, the command should include the extension `.exe`. If it does not include the extension, it will automatically be added.

commandArgs

The arguments for the command. If you specify an invalid argument and the command end with a nonzero return code, the task is disabled and will not be rescheduled until the agent server restarts.

commandType

The type of command. Here are the valid values:

ext

Specifies that the command is an external, user-supplied command. The output for this type of command must be written to `stdout`. If the output is to be published, it must use the JSON metric event format.

sas

Specifies that the command is provided by SAS. These commands include `sas-peek`, `sas-check`, `sas-logwatch`, and so on.

int

Specifies an internal task.

Note: Do not use this value.

publisherType

The type of publisher program for the task. Here are the valid values:

amqp

Specifies that the collector data is published using `sas-event-pub` to the AMQP server. This the default value if the `publishParm` attribute is set to `-exchange`.

http

Specifies that the collector data is published using `sas-event-pub` to an HTTP end point. This the default value if the `publishParm` attribute is set to `-url`.

file

Specifies that the collector data is published using `sas-event-pub` to a file. This the default value if the `publishParm` attribute is set to `-file`.

olog

Specifies that the collector data is published using `emi-outlog` to a file. This the default value if the `publishPgm` attribute is set to `emi-outlog`.

ext

Specifies that the collector data is published to an external program other than SAS.

none

Specifies that the collector data is not published. This is the default.

publisherCommand

Specifies the name of the publishing program that receives the output from the program that is specified in the task definition. The publishing program receives input on its `stdin` from the `stdout` of the task command. Do not include a path or command arguments. If the host system for the publisher is Windows, the command should include the extension `.exe`. If it does not include the extension, it will automatically be added.

publisherArgs

Specifies the arguments for the publisher program.

Change the Frequency of a Task

You can change how often a defined and recurring task runs by modifying the task definition. Some tasks are defined to run at a specified interval (for example, every two minutes), and some are defined to run at a specified time (for example, 2 AM). To make the change, you export the current task list, make the modifications, and import the task list back to the server. Follow these steps.

- 1 Export the current task list. Use this command:

```
sas-ops-agent export -tasks yourExportedTasksFileName.json
```

- 2 Modify the task file.

The definition for a task that runs at a specified time interval contains the line `"runType": "periodic",`. To change how often this type of task runs, change the line

```
"frequency": "Ns"
```

Substitute your time specification for *Ns*. Use the format

```
MhNmNsNms
```

For example, `2m30s` specifies that the task runs every 2 minutes and 30 seconds.

The definition for a task that runs at a specified time contains the line `"runType": "time",`. To change how often this type of task runs, change the line `"frequency": "YYYY-MM-DDTHH:MM:SS",` to use your time specification. For example, `03:30` specifies that the task runs at 3:30 AM.

- 3 Import the modified task list, overwriting the existing list. Use this command:

```
sas-ops-agent init -tasks yourNewTasksFileName.json -f
```

How To: ETL and Data Mart Operations

Modify Property Values

You can manually change the property values that are used by the ETL and data mart processes. Use the command `dm-admin --datamart evdm set property-name=property-value`

Here are the properties that you can specify:

EMI_CAS_LOAD

Specifies that the collected alert and notification messages are loaded into CAS. The default value is Y.

EMI_CAS_LOAD_LOGS

Specifies that the collected log messages are loaded into CAS. The default value is Y.

EMI_CAS_LOAD_METRICS

Specifies that the collected metric data is loaded into CAS. The default value is Y.

EMI_CAS_RETAIN_DAYS

Specifies the number of days that the metric and log data are retained in CAS. The default value is 3.

EMI_DELETE_TSVZIP_DAYS

Specifies the number of days to keep the metric and log data on disk in the data mart. The default value is 30.

EMI_ZIP_TSV_DAYS

Specifies the number of days before raw TSV files are compressed into a ZIP file. It also applies to the number of days to keep SAS log files that are generated by standard data mart batch jobs. The default value is 1.

Summary of ETL Return Codes

Here are possible return codes for ETL jobs:

Return code	Meaning
93	A required CAS table could not be initialized because the specified initialization job does not exist.
94	A required CAS table does not exist after the specified initialization job has been run.
95	A required SAS table could not be initialized because the specified initialization job does not exist.
96	A required SAS table does not exist after the specified initialization job has been run.
156	A lock on a data set could not be obtained within the specified time limit.
195	CAS configuration information has not been provided or located.
196	A CAS connection could not be established.
197	The contents of the data mart lock file could not be released when releasing the data mart lock.
198	The data mart lock could not be released.
199	The data mart lock could not be released because it is not locked by this process.
298	A data mart lock could not be obtained because the data mart is locked by another process.
299	A data mart lock could not be obtained because the data mart lock file cannot be renamed.

Operations Infrastructure Reference

Time Format

Time can be specified in various formats. Here are the general forms of time specifications:

- An amount of time, specified in the format 0h0m0s0ms
- A specific time and date, specified in the format YYYY-MM-DDTHH:MM:SS.ssssssZhh:mm

Subsets and variants of each form are allowed. Here are some examples:

Specification	Time specified
"5m20s"	5 minutes and 20 seconds
"2h"	2 hours
"426s"	426 seconds
"3:55:05.29754"	3:55 and 5.29754 seconds, in the current time zone, on any day Example: 0000-01-01 03:55:05.29754 -0500 EST
"14:18:34"	14:18 and 34 seconds, in the current time zone, on any day Example: 0000-01-01 14:18:34 -0500 EST
"14:18:34.38"	14:18 and 34.38 seconds, in the current time zone, on any day Example: 0000-01-01 14:18:34.38 -0500 EST
"14:28"	14:28, in the current time zone, on any day Example: 0000-01-01 14:28:00 -0500 EST
"5T14:25:04.54731"	14:25 and 4.54731 seconds, in the current time zone, on the 5th day of any month. Example: 0000-01-05 14:25:04.54731 -0500 EST
"05T09:37"	9:37, in the current time zone, on the 5th day of any month. Example: 0000-01-05 09:37:00 -0500 EST
"05T09"	9:00, in the current time zone, on the 5th day of any month. Example: 0000-01-05 09:00:00 -0500 EST
"05-23T17:26:09.237"	17:26 and 9.237 seconds, in the current time zone, on 23 May of any year. Example: 0000-05-23 17:26:09.237 -0500 EST
"04-05T09:37"	9:37, in the current time zone, on 5 April of any year. Example: 0000-04-05 09:37:00 -0500 EST
"04-16T21"	21:00, in the current time zone, on 16 April of any year. Example: 0000-04-16 21:00:00 -0500 EST
"03-15"	15 March, in the current time zone, of any year. Example: 0000-03-15 00:00:00 -0500 EST
"2015-04-05T14:18"	14:18, in the current time zone, on 5 April 2015. Example: 2015-04-05 14:18:00 -0400 EDT
"2015-04-05T18"	18:00, in the current time zone, on 5 April 2015. Example: 2015-04-05 18:00:00 -0400 EDT
"2018-01-16"	16 January, 2018, in the current time zone. Example: 2018-01-16 00:00:00 -0500 EST

Specification	Time specified
"05:21:19Z"	5:21 and 19 seconds UTC, on any day. Example: 0000-01-01 05:21:19 +0000 UTC
"08:22:00.21Z"	8:22 and 0.21 seconds UTC, on any day. Example: 0000-01-01 08:22:00.21 +0000 UTC
"14:25:04.54731Z"	14:25 and 4.54731 seconds UTC, on any day Example: 0000-01-01 14:25:04.54731 +0000 UTC
"3:55Z"	3:55 UTC, on any day Example: 0000-01-01 03:55:00 +0000 UTC
"17T14:25:04.54731Z"	14:25 and 4.54731 seconds UTC, on the 17th day of any month Example: 0000-01-17 14:25:04.54731 +0000 UTC
"05T09:37Z"	9:37 UTC, on the 5th day of any month Example: 0000-01-05 09:00:00 +0000 UTC
"05T09Z"	9:00 UTC, on the 5th day of any month Example: 0000-01-05 09:00:00 +0000 UTC
"05-23T17:26:09.23731Z"	17:26 and 9.23731 seconds UTC, on 23 May of any year Example: 0000-05-23 17:26:09.23731 +0000 UTC
"11-10T06:44Z"	6:44 UTC, on 10 November of any year Example: 0000-11-10 06:44:00 +0000 UTC
"09-21T16Z"	16:00 UTC, on 21 September of any year Example: 0000-09-21 16:00:00 +0000 UTC
"2019-11-07T13:21Z"	13:21 UTC, on 7 November, 2019 Example: 2019-11-07 13:21:00 +0000 UTC
"2019-11-07T11Z"	11:00 UTC, on 7 November, 2019 Example: 2019-11-07 11:00:00 +0000 UTC
"05T09+12"	9:00, UTC +12 hours, on any day Example: 0000-01-05 09:00:00 +1200 +1200
"2018-01-16+12:00"	16 January 2018, UTC +12 hours Example: 2018-01-16 00:00:00 +1200 +1200
"01-16+06:30"	16 January of any year, UTC +6 hours and 30 minutes Example: 0000-01-16 00:00:00 +0630 +0630

SAS Mobile BI

Mobile: Overview

The SAS Mobile BI app enables mobile device users to view and interact with reports that can contain a variety of charts, graphs, gauges, tables, and other report objects. Supported mobile devices include iPads, iPhones, Android tablets and smartphones, and Windows 10 tablets. For information about how to use the SAS Mobile BI app, see the [SAS Mobile BI Help](#).

As an administrator, you can control how a mobile device running the SAS Mobile BI app can access reports and data located on a SAS Visual Analytics server. You can use the following features, rules, and properties (alone or in combination) to control access to the server data and reports from the app:

Blacklist and whitelist feature

You can manage whether a device can access servers through the SAS Mobile BI app, either by exclusion or inclusion.

Passcode properties and rule

You can require SAS Mobile BI app users to lock the app with a passcode. You can configure two properties that control the behavior of the passcode.

Offline access time-out property and rule

If a user has not opened the SAS Mobile BI app for a specified number of days, you can require that he or she must enter the user ID and password to access the server. The time-out is specified by a server property. You can use a rule to identify users who are exempt from the time-out.

Remote report data rule

You can specify that when users view a report in the SAS Mobile BI app, the mobile device must maintain a network connection to the server.

Limit functionality in the app

You can limit the functionality of the SAS Mobile BI app by applying one or more rules to a user or group of users. Functionality includes whether a user can subscribe to and view reports; share links to reports (and screen captures) by using email, text messaging, or other functionality; add or view comments; see and use the Favorites or Recent views; and view alerts.

Mobile: How To

Manage Mobile Devices



Who Can Manage Mobile Devices?

Only members of the SAS Administrators group can manage mobile devices.

Navigate to the Mobile Devices Page

Blacklists and whitelists are managed on the **Mobile Devices** page.

Note: This page is available only if you are a member of the SAS Administrators group.


- 1 Click  and select **Manage Environment**.
- 2 In the navigation bar, click .

Add a Device to a List from Last Access

You can add a device that has already connected (or attempted to connect) to the blacklist or whitelist.



TIP This option is disabled if the ID already exists on the respective list.

Complete the following steps on the **Mobile Devices** page:

- 1 Click the **Last Access** tab.
- 2 Select the device and click .
- 3 Select the list to which you want to add the device.
- 4 In the Add Device window, click **Yes**.

Add One or More Devices to the Blacklist or Whitelist

Complete the following steps on the **Mobile Devices** page:


- 1 Click the **Blacklist** or **Whitelist** tab, depending on which list you want to add devices.
- 2 You can add one device or multiple devices to a list:
 - To add one device to a list, click .
Enter the Device ID in the Add to Whitelist window.
 - To add multiple devices to a list, click .
In the Add to Whitelist window, enter each Device ID to create a new line.

Note: Validation is not performed on the device IDs as they are added to the list.
- 3 Click **Save**.

Move One or More Devices between Lists


You can move devices from one list to the other (for example, from the blacklist to the whitelist).

Complete the following steps on the **Mobile Devices** page:

- 1 Click the tab that corresponds to the list from which you want to move a device.
- 2 Select one or more devices that you want to move, and click .
- 3 In the Move Device window, click **Yes**.

Remove One or More Devices from a List

Complete the following steps on the **Mobile Devices** page:

- 1 Click the tab that corresponds to the list from which you want to remove a device.
- 2 Select one or more devices that you want to remove, and click .
- 3 In the Confirm Remove window, click **Yes**.

View Logon Event Information

Complete the following steps on the **Mobile Devices** page:

- 1 Click the **Last Access** tab.
- 2 View the device logon event information, including status. See “[Device Logon Information](#)”.

TIP Use the **Filter by** drop-down list to filter the information about the tab.

View Previous Logon Events

You can view records that were captured from devices on a prior application version or operating system version.




Complete the following steps on the **Mobile Devices** page:

- 1 Click the **Last Access** tab.
- 2 Select the **Include device history** option.

Determine Which List Is Enforced

There are several ways to determine whether the blacklist or whitelist is being enforced.

On the **Mobile Devices** page, look for the following indicators:

- The list that is being enforced has a  next to the list name.
- The list that is being enforced displays the following message above the **Device ID** table:
“ This list is currently being enforced.”
- The list that is not being enforced displays the following message above the **Device ID** table:
“ This list is not currently being enforced.”

Change How Devices Are Managed

CAUTION! These are deployment-level instructions that affect user access. Changing how devices are managed can disrupt existing users by changing which devices are eligible to connect to servers through the SAS Mobile BI app.

- 1 Verify that the list that you intend to enforce is appropriately populated.
 - If you enforce the whitelist, the whitelist should contain all eligible devices. The blacklist is ignored.

- If you enforce the blacklist, the blacklist should contain all excluded devices. The whitelist is ignored.
- 2 On the **Mobile Devices** page, click the tab that corresponds to the list that you want to enforce.
 - 3 To change the list that is enabled, select the **Enable blacklist** or **Enable whitelist** option.
 - 4 In the confirmation window, click **Yes** to enable the new list.

Limit Functionality

Initially, all authenticated users can access all functionality in the SAS Mobile BI app.

- 1 To limit access to functionality, locate the relevant authorization rule.
- 2 Change the principal from its initial value (Authenticated Users) to a different value (for example, the group ID for a custom group).

Note: Users who are within the scope of a revised rule have access to the functionality that the rule provides. Other users do not have access to the functionality that the rule provides.

For details, see [“Rules to Control Access to SAS Mobile BI App Functionality” on page 566](#) and [“Adjust Rules for Access to Functionality” on page 481](#).

Limit Caching of Report Data

Initially, all mobile devices cache report data. To change this behavior:

- 1 Locate and adjust the following predefined authorization rule:

Object URI: /SASMobileBI_capabilities/cacheMobileReportData

Principal: Authenticated Users

- 2 Choose one of the following:
 - To prevent all caching of report data, disable the rule.
 - To selectively prevent caching of report data, change the principal from its initial value (Authenticated Users) to an alternate value (for example, the group ID for a custom group).

Note: For users who are within the scope of the revised rule, report data is cached. For users who are outside the scope of the revised rule, report data is downloaded when a report is open and purged when the report is closed. Offline access to reports is not supported for users who are outside the scope of the revised rule.

For details, see [“Adjust Rules for Access to Functionality” on page 481](#).

Manage Passcode Requirements

Initially, use of a passcode is not required. However, you can require passcodes or adjust passcode constraints.

Require Passcodes

- 1 Locate and adjust the following predefined authorization rule:

Object URI: /SASMobileBI_capabilities/exemptFromPasscodeRequirements

Principal: Authenticated Users

- 2 Choose one of the following:




- To require all users to use a passcode, disable the rule.
- To require some users to use a passcode, change the principal from its initial value (Authenticated Users) to an alternate value (for example, the group ID for a custom group).

Note: Users who are within the scope of the revised rule can choose to use a passcode, but they are not required to do so. Users who are outside the scope of the revised rule must use a passcode.

For details, see [“Adjust Rules for Access to Functionality” on page 481](#).

Adjust Passcode Constraints

To adjust the passcode constraints, use the **passcodeAttempts** and **passcodeTimeoutMinutes** properties in the **sas.devicemanagement** configuration definition.

- 1 Click  and select **Manage Environment**.
- 2 In the navigation bar, click .
- 3 In the **View** list, choose **Definitions**.
- 4 In the **Filter** field, type `device`.
- 5 Select **sas.devicemanagement** from the results. The configuration properties appear in the right pane.
- 6 Click .
- 7 Edit the value in the **passcodeAttempts** field to configure the passcode lock-out behavior.
- 8 Edit the value in the **passcodeTimeoutMinutes** field to configure the passcode time-out behavior.
- 9 Click **Save**.

Manage Offline-Access Time-Outs

Initially, no users are subject to time-outs for offline access. However, you can prevent time-outs and adjust the time-out interval.

Prevent Time-Outs

To implement time-outs for offline access:

- 1 Locate and adjust the following predefined authorization rule:




Object URI:	<code>/SASMobileBI_capabilities/exemptFromOfflineTimeLimit</code>
Principal:	Authenticated Users
- 2 Choose one of the following:
 - To make all users subject to time-outs, disable the rule.
 - To make some users subject to time-outs, change the principal from its initial value (Authenticated Users) to an alternate value (for example, the group ID for a custom group).

Note: Users who are within the scope of the revised rule are exempt from time-outs. Users who are outside the scope of the revised rule are subject to time-outs.

For details, see [“Adjust Rules for Access to Functionality” on page 481](#).

Adjust the Time-Out Interval

To adjust the time limit, set the **offlineLimitDays** property in the **sas.devicemanagement** configuration definition.

- 1 Click  and select **Manage Environment**.
- 2 In the navigation bar, click .
- 3 In the **View** list, choose **Definitions**.
- 4 In the **Filter** field, type `device`.
- 5 Select **sas.devicemanagement** from the results. The configuration properties appear in the right pane.
- 6 Click .
- 7 Edit the value in the **offlineLimitDays** field. Specify, in days, how many days a device can be offline without requiring the user log on to the server when opening SAS Mobile BI app again.
- 8 Click **Save**.

Mobile: Concepts

SAS Viya provides ways to manage mobile devices and the security of reports and data accessed by means of a SAS Viya server. You can manage mobile devices by using a combination of configuration properties for the server and rule authorizations that control the access of mobile device users to the server.

Prerequisites for Managing Mobile Devices

To manage mobile devices, you must be a SAS administrator, and your user ID must have the device management rule authorization for mobile devices. By default, SAS administrators already are granted authorization for the `/deviceManagement_capabilities/manageMobileDevices` rule.

Prerequisites for Mobile Device Access

The SAS Mobile BI app uses the `/SASMobileBI/**` authorization rule to identify those who can log on to the server from the app. By default, all authenticated users are added to this rule. A SAS administrator can choose to update the rule to identify users or groups of users to prohibit from using the app.

Blacklist and Whitelist Features

Overview

The *whitelist* manages the devices that can access servers by using the SAS Mobile BI app. A device must be on the whitelist in order to use SAS Mobile BI on your network. The whitelist affects devices, not users. If a device is lost, a SAS administrator can remove the device from the whitelist and prevent access to the reports and data.

The *blacklist* manages the devices that cannot access servers by using the SAS Mobile BI app. All devices can use SAS Mobile BI on your network except those that are on the blacklist. The blacklist affects devices, not

users. If a device is lost, a SAS administrator can add the device to the blacklist and prevent access to the reports and data.

Considerations

Here are the key points for managing mobile devices:

- You can manage devices either by exclusion or by inclusion.
 - If you manage by exclusion, all devices can access servers through the SAS Mobile BI app, except those that are on the blacklist. A blacklist is a list of mobile devices that are not authorized to use the SAS Mobile BI app.
 - If you manage by inclusion, only devices that are on the whitelist can access servers through the SAS Mobile BI app. A whitelist is a list of mobile devices that are authorized to use the SAS Mobile BI app.
- A deployment enforces only one list (either the blacklist or the whitelist) at a time.
- In a new deployment, the blacklist is enforced and contains no items. Therefore, all devices can access servers through the SAS Mobile BI app.
- You can modify both lists. Making changes to a list that is not currently enforced can help accommodate a future change.
- The blacklist and whitelist affect devices, not users. As an administrator, you authorize what a particular user can see or do. For more information, see [General Authorization on page 107](#).

Add Devices by User ID

The easiest way to add a device to the whitelist or blacklist is to add a device that has already connected (or attempted to connect) to the server. When the attempt is made, the **Last Access** tab logs the device owner's user ID, device ID, device type, and other information. You can sort the **User ID** column to locate the user ID of the person whom you want to add.

Restricting and enabling devices by user ID is a best practice because users can have more than one device. By identifying the user ID, you can be sure to add all devices used by that person.

TIP The only way to add a device running Windows 10 is by user ID.

Passcode Feature

Overview

The passcode feature locks the SAS Mobile BI app. This feature is separate from and in addition to the passcode feature that is provided by mobile devices. There are two types of app passcodes: required and optional.

A *required passcode* is a passcode that is required by the server. When the app first connects to the affected server, the server forces the app to require that the app user create a passcode. Then, whenever the app user opens the app or views a report that is associated with that server, the user must enter the passcode.

Note: By using an additional rule, the SAS administrator can exempt app users from using a passcode. By using a combination of two rules, all mobile devices that access the server must use a passcode except for those separately exempted.

An *optional passcode* is a passcode that the app user can choose to use to lock the app. The passcode is not required to access the server. The app user can disable the passcode at any time.

Considerations

Here are some key points to remember when working with passcodes:

- The passcode should be known only to the app user. If the app user loses the mobile device, no one else should be able to guess the passcode and use it to open the app.
- The passcode has a time-out feature. The SAS administrator can customize the `passcodeTimeoutMinutes` setting to configure this feature. This setting specifies, in minutes, how long a user must wait before re-entering his or her passcode in the SAS Mobile BI app. The default is 15.

If the app user (or another person) provides an incorrect passcode a specific number of times (`passcodeAttempts`), the app locks itself for a length of time (`passcodeTimeoutMinutes`). The app user can enter the passcode again after the time-out expires.

- The passcode has a lock-out feature. The SAS administrator can customize the `passcodeAttempts` setting to configure this feature. The setting limits the number of sequential, failed attempts to enter a passcode for the SAS Mobile BI app. The default is 5.

If a user reaches the specified limit (`passcodeAttempts`), the user is timed out of the app for 15 minutes (or the value set for `passcodeTimeoutMinutes`). After the time-out interval, the user can make one more attempt to enter his or her passcode. If the password fails again, all custom content (data, reports, settings, and connection information) is removed from the mobile device. The app is reset to its default settings.

- If the app user forgets the passcode, the app user must delete and re-install the app on the device. Doing so deletes the reports and data.

For information about how app users set a passcode, see the [SAS Mobile BI Help](#). Be sure to view the Help for the platform (iOS, Android, or Windows 10) and release that you are using.

Remote Report Data Feature

Remote Report Data

When you subscribe to a report, it appears in the **Subscriptions** view of SAS Mobile BI. However, depending on the security assigned to the user ID, the report data might not exist on the mobile device. Report data can be local or remote:

- *Local* data is stored on the mobile device.
- *Remote* data exists on the mobile device only while the report is open and the device is connected to a Wi-Fi or cellular network. If a report uses remote data, the report tile in the **Subscriptions** view displays the cloud icon.

How Remote Data Works

Each time you open a report with remote data, the app connects to the server. The Prepare Data notification is displayed while the data is downloaded. The report opens when the data is available on the mobile device. The data is available only while you view the report.

After you close the report, the data is removed from the device. The thumbnail image on the report tile no longer appears. If you are not connected to a network and you try to open the report, it does not open.

This feature affects the user ID that is used to access the server. When you access the server via SAS Mobile BI using that user ID, all reports on that server use the remote report data feature.

Prevent Mobile Devices from Storing Report Data

The /SASMobileBI_capabilities/cacheMobileReportData rule specifies that a mobile device can store (or cache) report data on the device when it is not connected to a network. By default, all authenticated users' mobile devices can cache report data.

If you want to enforce additional security by preventing mobile devices from storing report data, then you must prohibit the rule authorization that is applied to a user or group of users.

Offline-Access Time-Out Feature

If a user has been offline for a specified number of days, he or she must sign in to the server used by the SAS Mobile BI app. For example, if the user attempts to browse reports on the server or open a report in the report viewer, the app requires the user to enter the password for the requested server connection. If the user fails to sign in, then the app no longer downloads reports, updates subscribed reports, or opens reports for viewing.

This feature is not only useful when the device is missing. It also provides security when the employee leaves the organization but keeps the device. The blacklist and whitelist features require that the device must access the server before the list can look up the device to deny or permit access. The offline access time-out feature denies access by checking the employee's credentials, which the IT organization revokes when the employee leaves the organization.

Mobile: Reference

Device ID Criteria

To add one or more devices to the blacklist or whitelist, you must enter valid device IDs. If an invalid ID is entered, you cannot add devices to the lists. In order for an ID to be valid, the following conditions must be met:

- non-empty string
- length of 36 characters or less
- contains alphanumeric characters and hyphens
- is not a duplicate of an existing device ID

Device Logon Information

A device might appear multiple times in the blacklist or whitelist if a different user ID attempts to log on with a device that has already been captured. The following occurrences are logon events:

- a connection attempt that comes from a new source (a unique combination of device ID and user ID)
- a connection attempt that comes from an existing source (existing device ID and new user ID)
- a connection attempt that is accompanied by a device change (such as a new operating system version or application version)

The following table lists the device status icons that might be displayed:

Table A.44 Status Icons and Descriptions

Icon	Description
●	Indicates that the authentication was a success.
◆	Indicates that the authentication was a failure.
■	Indicates that the device is in the blacklist.
▲	Indicates that the device is not in the whitelist.

Rules to Control Access to SAS Mobile BI App Functionality

You can limit the functionality of the SAS Mobile BI app by applying one or more rules to a user or group of users.

The following table lists the rules that enable you to limit a user's access to functionality in the SAS Mobile BI app:

Table A.45 Rules and Descriptions

Rule	Description
ACCESS_MOBILE_BI	Enables a user to subscribe to and view reports.
ACCESS_RECENTS	Enables a user to see items for a particular server in their Recent view. It also enables the user, in Add Reports, to see the contents of the Recent folder for that server.
ADD_COMMENTS	Enables a user to add comments to a report or its contents.
EMAIL	Enables a user to share links to reports (and screen captures) by using email, text messaging, or other functionality.
EXPORT_DATA	Enables a user to export data for a report object.
EXPORT_DETAILED_DATA	<p>Enables a user to select the Detailed data option (if applicable) in the Export Data window.</p> <p>Note:</p> <p>The EXPORT_DATA rule takes precedence over this rule. If a user ID is not authorized to use the EXPORT_DATA rule, then authorizing the EXPORT_DETAIL_DATA rule to that user ID has no effect.</p>

Rule	Description
MANAGE_FAVORITES	<p>Enables a user to add a report to the Favorites view or to remove a report from the Favorites view.</p> <p>Note:</p> <p>The VIEW_FAVORITES rule takes precedence over this rule. If a user ID is not authorized to view favorites, the user cannot add or remove favorites, even if the user is authorized to use the MANAGE_FAVORITES rule.</p>
SUBSCRIBE_TO_REPORT_ALERTS	Enables a user to view alerts in a report and to subscribe to them.
VIEW_COMMENTS	Enables a user to view the comments that are associated with a report or its content.
VIEW_FAVORITES	Enables a user to see items for a particular server in the Favorites view. It also enables the user, in Add Reports, to see the contents of the Favorites folder for that server.

Mobile: Troubleshooting

TIP For more troubleshooting information about the SAS Mobile BI app, see the [SAS Mobile BI Help](#). Be sure to view the Help for the platform (iOS, Android, or Windows 10) that you are using.

A user cannot open reports on an offline device.

Explanation:

The user ID might be required to use remote report data.

The user ID might be affected by the offline-access time out.

Resolution:

If the user ID is subject to the remote report data authorization rule, make sure the user understands that he or she must be connected to a network while viewing the report. See [“Remote Report Data Feature”](#).

If the user ID is subject to the offline-access time out authorization rule, make sure the user can log on to the server connection in the SAS Mobile BI app. See [“Offline-Access Time-Out Feature”](#).

A user is prompted for an application passcode.

Explanation:

The user is required to secure the SAS Mobile BI app with a passcode. See [“Passcode Feature”](#).

Resolution:

To learn how to create a required passcode in the SAS Mobile BI app, see the [SAS Mobile BI Help](#).

Note: Be sure to view the Help for the platform (iOS, Android, or Windows 10) that you are using.

In the Mobile Devices window, a message indicates that a list is not currently in use.

Explanation:

By design, only one list (either the blacklist or the whitelist) is in use.

Blacklist and whitelist status displays ‘status unknown’.

Explanation:

If you are working in a multi-tenant environment, then device configuration properties are not available to display the status of the blacklist and whitelist. These properties are available only in the provider tenant configuration. Other locations show a status of 'unknown' unless the administrator specifically sets configuration properties for the individual tenant locations.

Multi-Tenancy

Multi-tenancy: Overview

This document is applicable to only a [multi-tenant](#) deployment.

Multi-tenancy gives the *provider* the ability to manage isolated, independent *tenants* within a single deployment. These tenants have access to all licensed software, but have no visibility into each other's data and workflows. Tenants are generally administered from within, but certain administrative functions are limited to the provider.

The process of creating and configuring tenants is known as *onboarding*. Decommissioning and removing tenants is known as *offboarding*.

A multi-tenant deployment has two scopes for administration: provider-level administration and intra-tenant administration.

- *Provider-level administration* includes administration of tenants (management of tenants by the provider from the provider's own default tenant). For example, creating and onboarding tenants are provider-level tasks. Provider-level system administration (for example, monitoring machine performance, viewing system-level logs, and so on) is handled by the provider, in the provider's tenant.
- *Intra-tenant administration* consists of administration by each tenant of its own internal resources. For example, assigning users to custom groups and managing access to SAS Viya content and CAS data are intra-tenant tasks.

For more information, see [“Distribution of Responsibilities” on page 582](#).

Multi-tenancy: Initial Tasks

Configuring LDAP for Onboarding Tenants

- 1 Verify that your initial deployment included multi-tenancy.

TIP A simple test is to look at the SAS Environment Manager user interface, when you are signed in as an administrator. If the deployment is multi-tenancy-enabled, the **Tenants** option is visible in the left-navigation bar. If not, then you must re-install with multi-tenancy enabled before you can configure your tenants.

See [“Enable Multi-tenancy” in SAS Viya for Linux: Deployment Guide](#) for more details.

Note: When you run the playbook, a multi-tenancy enabled deployment leads to a single tenant (with tenant ID: “provider”) configured for the provider.

- 2 Configure your LDAP server for multi-tenancy. See [“Identity Providers” on page 484](#) for more details. The LDAP environment should be structured in the following way:

```

dc=example,dc=com
  ou=tenant1
    ou=groups
    ou=users
  ou=tenant2
    ou=groups
    ou=users
  ...
  ou=provider
    ou=groups
    ou=users

```

All tenants must share a single LDAP server. Follow these steps to configure the LDAP server for multi-tenancy:

- a Configure the base distinguished name (baseDN) for both groups and users via the standard `sas.identities.providers.ldap.group.baseDn` and `sas.identities.providers.ldap.user.baseDn` properties. SAS recommends that both properties are set to the root distinguished name (rootDN) that all tenants are created under, such as `dc=example,dc=com`.

Note: Doing so is recommended because when the system sends a request to the LDAP server, the appropriate searchDN for the search request is dynamically created by appending the rootDN, the tenant OU, and the tenant-specific user or group relative distinguished names (RDNs). This strategy ensures that the search request is looking only at users and groups in a specific tenant.

- b Within that LDAP server, ensure that a unique organizational unit (OU) is defined for each tenant, where the name of the OU matches the tenant ID.
- c Create the tenant OUs under a common base distinguished name (DN) because the tenant OUs should be peers of one another. Here is an example: `ou=tenant1,dc=example,dc=com`.
- d Within a tenant OU, groups and users are created by default under the following relative distinguished name (RDN): `ou=groups,ou=users`

Modify these values by editing the two properties:

`sas.identities.providers.ldap.tenancy.groupRdn` and `sas.identities.providers.ldap.tenancy.userRdn` in SAS Environment Manager.

Note: The same structure must be used for all tenants, including the provider.

- 3 Once the LDAP configuration is in place, you must also follow these guidelines:
 - a Users are assigned to a specific OU, so they can access only the associated tenant. Users do not have access to other tenants. This includes provider-level administrators such as `sasboot`. Provider-level administrators are expected to access the tenant using the special `sasprovider` account, which is created internally (not part of LDAP) and is separate and specific to each given tenant.
 - b For each user defined in LDAP, follow these steps:
 - i Specify the `uidNumber` and `gidNumber` attributes.
 - ii Set the `homeDirectory` attribute to the default home directory location for users on the host. For `user1` on Linux, this is `/home/user1`.
A system administrator must create this directory before the user can access SAS Studio.
 - iii Set the `loginShell` attribute to `/bin/bash`.

A combination of the tenant ID and the value in the `zones.internal.hostnames` property determines to which tenant a user authenticates. For example, if you name your tenant `acme`, and `zones.internal.hostnames`

contains `sas.com`, then you can authenticate to the `acme` tenant using the address `acme.sas.com`. Note that this domain (for example: `acme`) is relevant only for authentication. Once a user has authenticated, the domain is no longer used to resolve tenancy. Tenancy travels with the user for that session.

Onboard Tenants

The process of creating a tenant and making it ready for use is known as *onboarding*. Follow these steps to onboard tenants.

Note: You do not have to onboard all tenants at the same time. More tenants can be added later.

TIP Take a full binary backup before and particularly after onboarding tenants. See “How to Use the Binary Backup and the Default Backup” on page 138 for more details.

- 1 Ensure that you have sufficient resources to onboard tenants. For more details, see “Add Resources” on page 575.
- 2 Create the CAS controller that the tenant will use. See “SAS Cloud Analytic Services: How To (CAS Server Monitor)” on page 614 for more information.
- 3 If you intend to have a backup controller for your tenant, then both the primary and backup (secondary) controller must use a shared file system. Configure the shared file system with the following steps:
 - Create and configure the directory `/opt/sas/tenant/config/data/cas` on both controllers. Where *tenant* is the tenant ID of the tenant that you plan to onboard.


```
sudo mkdir -p /opt/sas/tenant/config/data/cas
sudo chown -R sas:sas /opt/sas/tenant/config/data/cas
sudo chmod 0755 /opt/sas/tenant/config/data/cas
```
 - Mount the shared location on both controllers:


```
sudo mount source-machine:path_to_shared_location /opt/sas/tenant/config/data/cas
```

Where *source-machine* is the host name of the machine where the files will be stored and *path_to_shared_location* is the absolute path to the shared location.
- 4 Complete the [start-up instructions](#) for using the command-line interface.
- 5 Follow this structure for the `tenant_vars.yml` file.
 - a Replace values in single brackets (including the brackets themselves) with your values. For example, to onboard a tenant with tenant ID of `acme`, `sas_tenant: {put tenantID here}` becomes `sas_tenant: acme`.
 - b Do not edit any values in double brackets (including the double brackets).

```
DEPLOYMENT_ID: {put tenantID here}
sas_tenant: {put the same tenantID here}

tenant_instance: default
tenant_admin: {put tenant admin user here -
must match a member of the tenant_admin_group for this tenant in LDAP}
tenant_admin_group: {put the LDAP group that contains tenant admins here -
note that this value cannot contain spaces}
tenant_users_group: {put the LDAP group that contains tenant users here}
tenant_provider_pwd: {create a password for the sasprovider user in this tenant here}

provider_admin: {put a specific user that has the SASAdministrators group permissions here}
```

```

provider_admin_pwd: {put the existing password of the provider admin here.
Needed for authentication of the CLI onboarding commands.}
provider_endpoint_scheme: {put https or http here}
provider_endpoint_port: {put 80 or some other available port number here}

INSTALL_USER: "{{ tenant_admin }}"
INSTALL_GROUP: "{{ tenant_admin_group }}"

SPAWNER_CONFIGURATION:
  sasPort: {put a unique port number for the Object Spawner here}

sasenv_connect_port: {put a unique port number for the SAS/CONNECT Spawner here}
sasenv_connect_mgmt_port: {put a unique port number for the SAS/Connect Management here}

CLUSTER_DEFINITIONS:
  cas:
    default:
      primary_host: {value must be in the sas-casserver-primary host group in inventory.ini}
      secondary_host: {value must be in the sas-casserver-primary host group in inventory.ini}
      worker_hosts: {values use hosts in the sas-casserver-primary
host group in inventory.ini. List is comma delimited}
      tenant: "{{ sas_tenant }}"
      casenv_user: "{{ tenant_admin }}"
      casenv_group: "{{ tenant_admin_group }}"
    cas:
      port: {put a unique port number for the CAS Server Starting Port here}
      httpport: {put a unique port number for the CAS Server Monitor here}
      gcport: 0

# Creates a autoexec_deployment.sas
#WORKSPACESERVER_CONFIGURATION:
#1: '/* Comment about key */'
#2: key=value;

# Creates a sasenv_deployment file
#FOUNDATION_CONFIGURATION:
#1: '# Comment about KEY'
#2: KEY=value

# Creates a sasv9_deployment.cfg file
#SASV9_CONFIGURATION:
#1: '/* Comment about OPTION */'
#2: 'OPTION value'

```

- c Ensure that host names must match those in the inventory.ini file.
- d Ensure that user names are those specified in the LDAP server. The identities must be in the correct OU.
- e Port values are configurable, so ensure that ports are unique for the tenant and available.

TIP This file contains unencrypted passwords for both the sasprovider user, and a provider-level administrator. The first is the user account that you, the provider, use to sign in to the tenant, where you can assume administrative privileges. The second gives you provider-level administrative privileges. Restrict access to this file.

- 6 Create a *tenantID_vars.yml* file, where *tenantID* is the tenant ID that you want to give to the new tenant. This value must also match the OU defined in the LDAP environment. Create this file in the same directory as your Ansible playbook.


Note: The tenant ID must be unique, 1–16 characters, contain only lowercase letters (only a-z in English alphabet) and numbers, and begin with a letter. The following identities are reserved: *default*, *provider*, *shared*, *sharedservices*, *spre*, *uaa*, *viya*, and any identity starting with “sas”.

- 7 Run the multi-tenancy play to onboard the tenant.

```
cd path_to_playbook
ansible-playbook -i some_inventory.ini utility/multi-tenancy.yml -e "@tenantID_vars.yml" -vv
```

- 8 Replace *tenantID* with your chosen tenant ID, and *some_inventory.ini* with the name of your inventory file.

Validate the Tenant

- 1 Sign in to SAS Environment Manager for the provider-tenant and opt in to the SAS Administrator group.
In the navigation bar, click . Select the new tenant and confirm that the **Access Policy** is set to **Open**. This allows members of the associated LDAP OU to access the tenant.
- 2 In SAS Environment Manager for the provider, check that the tenant is onboarded.
On the Tenants page, confirm that the new tenant has a status of onboarded.
- 3 When the play is complete, validate the tenant environment:
 - a Sign in to the newly created tenant at *tenant.hostname/SASEnvironmentManager* using the *sasprovider* user that you configured in the playbook. For example, if you normally access the software at *viya.example.com/SASEnvironmentManager*, then for a tenant called *acme* the URL becomes: *acme.viya.example.com/SASEnvironmentManager*.
 - b Ensure that you can access SAS Environment Manager regardless of whether you opt in to your assumable groups.
- 4 Validate that the tenant administrator was created.
While in SAS Environment Manager as the *sasprovider* user, assume the SAS Administrator group. Confirm that your intended tenant administrators have been added to the SAS Administrators group. See [“View User and Group Information” on page 476](#) for more information.
- 5 Confirm that you can start a SAS Cloud Analytics Services (CAS) session. See [“SAS Workspace Servers and SAS Cloud Analytic Services” on page 667](#) for more details.
- 6 Validate that the home directories exist for the tenant, and are accessible from all machines.
- 7 Share the tenant URL (*tenant.hostname/SASEnvironmentManager*) with your intra-tenant administrators.

Initial Tasks for the Intra-Tenant Administrator

As the tenant administrator, you have SAS Administrator privileges within your tenant.

- 1 Access *tenant.hostname/SASEnvironmentManager*. Sign in with your credentials. Opt in to SAS Administrators assumable group. If you are not prompted to opt in, check with your provider-level administrator that you have been added to the SAS Administrators group.
- 2 Assign users to custom groups, as needed. For example, you might want to add more users to the SAS Administrators group. Note that such changes apply only within your tenant. See *SAS Viya Administration: Identity Management* for more details.

- 3 Provide your users with the tenant url: *tenant.hostname/SASStudio* or, more generally, *tenant.hostname/SASHome*.

Multi-Tenancy: How To Manage Tenants

Introduction

These instructions explain how a provider can use [SAS Environment Manager](#) and the sas-admin tenant command-line interface to manage tenants in a [multi-tenant](#) deployment.



In addition, instructions are provided to add resources to existing tenants, offboard tenants, and onboard a previously offboarded tenant.

- To onboard a tenant, see [Add the tenants on page 571](#) in the topic “[Configuring LDAP for Onboarding Tenants](#)” on page 569.
- To offboard a tenant, see “[Offboard a Tenant](#)” on page 576.


Navigation

- 1 Sign in to SASHome on the provider tenant.


TIP Tenant management is administration of tenants. Such administration is a provider-level activity. It is performed in the provider by members of the provider’s SAS Administrators group. See “[Distribution of Responsibilities](#)” on page 582.

- 2 In the applications menu () , select **Administration** ⇒ **Manage Environment**.
In the navigation bar, click .

View the Properties of a Tenant

- 1 On the **Tenants** page, select a row.
- 2 In the table toolbar, click .

Edit the Properties of a Tenant


- 1 On the **Tenants** page, select a row.
- 2 In the table toolbar, click .
- 3 Modify the tenant’s name, description, or access policy, and then click **OK**. Note that the tenant ID cannot be changed.

Here are details about the **Access policy** values:

Open causes the tenant to be available to all users in the tenants OU.

Limited causes the tenant to be available to the sasprovider user.

View the Status of the Services for a Tenant

- 1 On the **Tenants** page, select a row.
- 2 In the table toolbar, click .

Note: The Services Status window provides a list of all the services for the selected tenant and his or her status. When all services have a status of “onboarded”, the tenant is fully onboarded.

Add Resources

Each new tenant requires resources. Overall scaling of resources is no different in a multi-tenant environment than a standard deployment. It is recommended that you mirror the repository before deployment and scale using the mirror. See [“Use a Mirror Repository” in SAS Viya for Linux: Deployment Guide](#) for more information.

Add Shared Resources across All Tenants

The service layer, including SAS Configuration Server, SAS Infrastructure Data Server, and the SAS Message Broker, is shared across all tenants. To scale these functions, add resources to the original inventory file and rerun the playbook. Additional services might be needed to support higher load requests.

Add Resources to a Specific Tenant

Each tenant has its own CAS controller. A single machine can host multiple CAS configurations. It is recommended to use a single CAS controller per machine in a production environment.

If additional machines are needed for CAS, the provider appends machines to the CAS host group in the inventory file. The playbook is run with the updated inventory to install software, from the mirror, to the new machines.

The programming run-time is also configured per-tenant. This includes the Object Spawner, SAS/CONNECT Spawner, and SAS Launcher Service. A single machine can host multiple programming run-time configurations. The tenant’s inventory file configures the host groups for the programming run-time. This inventory is used in the onboarding process.

Enable Access to the Tenant

Access SAS Environment Manager on the provider tenant and assume administrative privileges. On the Tenants pane, set the **Access Policy** to **Open** for the tenant.

All users that are members of this tenant’s OU now have authorization to access the tenant.

Disable Access to the Tenant

Access SAS Environment Manager on the provider tenant and assume administrative privileges. On the Tenants pane, set the **Access Policy** to **Limited** for the tenant.

No tenant user or intra-tenant administrator can now sign in to the tenant, but jobs and services continue to run.

Note: Users who have active sessions can continue to work.

TIP Disable access to tenants before [offboarding on page 576](#).

Offboard a Tenant

Overview


The process of removing a tenant from your system is known as *offboarding*. There are various stages to offboarding a tenant.

- 1 Stop tenant users and tenant administrators from accessing the tenant.
- 2 Stop the services that are dedicated to the tenant.
- 3 Delete the tenant and its associated resources.

Prepare for Offboarding

A tenant can be offboarded only from certain states. Ensure that the tenant is onboarded or onboarding.

Disable access to the tenant, see [“Disable Access to the Tenant” on page 575](#) for more details.

- 1 Stop any scheduled jobs.
- 2 Sign in to SAS Environment Manager as sasprovider for the tenant:
 - a Select  from the left navigation.
 - b Schedule jobs have a green check in the **Scheduled** column. Select each scheduled job and click **Unschedule**.
- 3 Validate that there are no scheduled jobs for the tenant.
 - a Obtain an Oauth2 token from SASLogon for sasprovider in the relevant tenant. See [“Obtain an Access Token Using Password Credentials” on page 41](#) for more details.
 - b Use this token to get jobs from the scheduler:


```
curl -s -H "Accept:application/json" -H "Authorization:bearer $TOKEN"
http://tenant.hostname:port/scheduler/jobs
```
 - c Confirm that the above command returns zero items in its **Resource Collection**.

TIP Take a full binary backup before offboarding tenants. See [“How to Use the Binary Backup and the Default Backup” on page 138](#) for more details.

Stop Tenant-Based Services

CAUTION! This process interrupts jobs that are running on the tenant.

On all machines with the following host groups:

- sas-casserver-*
- programming
- computeServer

run the following command:

```
sudo /opt/sas/viya/home/libexec/deployment/tenant-services.sh
--tenant tenantID --action stop
```


Where *tenantID* is the tenant ID of the tenant to be offboarded.

This script does the following:

- stops CAS servers associated with *tenant*
- stops any launchers and compute servers associated with *tenant*
- stops any object spawners and workspace servers associated with *tenant*
- stops any connect spawners associated with *tenant*

Remove the Tenant Configuration

- 1 Initialize the `sas-admin` command-line interface:

```
sas-admin prof init
```

- 2 Enter the tenant's URL as the service endpoint. Here is an example: `http://acme.viya.example.com/`.

- 3 Sign in as the `sasprovider` user for this tenant:

```
sas-admin auth login
```

- 4 List the configuration with the following command:

```
sas-admin configuration configurations list
```

- 5 For each ID value returned by this command, delete the configuration:

```
sas-admin configuration configurations delete --id idfromlist
```

For example, if the list command returns: `"definitionName": "sas.casmanagement.global"` with an ID of `"id": "b37bf71b-6d09-4283-a1da-fada7408158f"`, then delete this configuration:

```
sas-admin configuration configurations delete --id b37bf71b-6d09-4283-a1da-fada7408158f
```

Offboard Services for the Tenant

- 1 Sign in to the host that has the `CommandLine` host group, navigate to `/opt/sas/viya/home/bin`, and initialize the `sas-admin` command-line interface:

```
sas-admin prof init
```

- 2 Enter the provider URL for the service endpoint. Here is an example: `http://viya.example.com/`. Note that this URL does not contain the tenant ID.

- 3 Sign in as a provider-level administrator:

```
sas-admin auth login
```

- 4 To inform the service layer that the tenant is not active, issue the command:

```
sas-admin tenant offboard --id tenant
```

where *tenant* is the identity of the tenant.

- 5 Only after the services have reached the status of "Offboarded" (see the status with the command `sas-admin tenant show --id tenant`), remove the tenant definition from the tenant service:

```
sas-admin tenant delete --id tenant
```

Note: It is important that no services are onboarding or onboarded when you perform this step. One way to be absolutely sure that no services are in the wrong state is to defer this step to the end of the offboarding process.

Delete Tenant Resources

CAUTION! This process cannot be undone.

- 1 Remove any tenant-specific configuration values from the SAS Configuration Server (Consul). Run the following commands on the machine that has the `sas-bootstrap-config` in which `SAS_Viya-Token` is the full path to your client consul token:

```
sudo /opt/sas/viya/home/bin/sas-bootstrap-config --token-file SAS_Viya-Token
kv delete locks/identities-service/tenantID-cacheStatus
```

```
sudo /opt/sas/viya/home/bin/sas-bootstrap-config --token-file SAS_Viya-Token
kv delete --recurse config/application/fact/connect-spawner-tenantID
```

```
sudo /opt/sas/viya/home/bin/sas-bootstrap-config --token-file SAS_Viya-Token
kv delete --recurse config/application/fact/launcher-server-tenantID
```

```
sudo /opt/sas/viya/home/bin/sas-bootstrap-config --token-file SAS_Viya-Token
kv delete --recurse config/cas-tenantID-default
```

```
sudo /opt/sas/viya/home/bin/sas-bootstrap-config --token-file SAS_Viya-Token
kv delete --recurse configuration-service/sitedefaults/config/cas-tenantID-default
```

- 2 To validate the Delete operations, run the following command for each deleted configuration:

```
sudo /opt/sas/viya/home/bin/sas-bootstrap-config --token-file
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/consul/default/client.token
kv read --recurse config_item
```

Where `config_item` is the tenant-specific item at the end of each of the preceding commands. Here is an example: `configuration-service/sitedefaults/config/cas-tenantID-default`.

- 3 Drop the database schemas for the tenant.

Note: A script is provided to remove schemas. The script is named `sds_manage_database_schema.sh` and can be found in the `/opt/sas/viya/home/libexec/sasdatasvc/script/` folder. The script must be run as the `sas` user with `sudo` privileges and is run from the host for `sasdatasvc`.

- 4 Remove the schemas by issuing the command:

```
/opt/sas/viya/home/libexec/sasdatasvc/script/sds_manage_database_schema.sh -t tenantID -d SharedServices
```

- 5 On the same machine, after the schemas are removed, drop the tenant-specific database users. One such user (per tenant) was created for you: `dataminingwarehousetenantID`. Drop this user by issuing the command:

```
/opt/sas/viya/home/libexec/sasdatasvc/scripts/sds_manage_database_user.sh -u dataminingwarehousetenantID
-o drop -d SharedServices -k
config/application/tenants/tenantID/sas.datamining.warehouse
```

- 6 Remove the logon service configuration for this tenant. On the `sasdatasvc` host, do the following:

a `/opt/sas/viya/home/bin/psql -h localhost -p port -U dbmsowner -d SharedServices`

b Enter the password for the `dbmsowner`.

c Run the following SQL commands:

- `delete from logon_provider.group_membership where member_id in (select id from logon_provider.users where identity_zone_id='tenantID');`
- `delete from logon_provider.user_info where user_id in (select id from logon_provider.users where identity_zone_id='tenantID');`

- `delete from logon_provider.users where identity_zone_id='tenantID');`
- `delete from logon_provider.group_membership where group_id in (select id from logon_provider.groups where identity_zone_id='tenantID');`
- `delete from logon_provider.groups where identity_zone_id='tenantID');`
- `delete from logon_provider.identity_provider where name='tenantID');`
- `delete from logon_provider.identity_zone where name='tenantID');`

7 Remove tenant directories from hosts for this tenant.

On the Ansible controller machine where the initial deployment was performed and the inventory file exists, create a script named `delete_tenant_directories.sh` with the following content:

```
#!/bin/bash
#command: <script_name> tenant_name instance_name  playbook_directory inventory_name
_tenant=$1
_instance=$2
_directory=$3
_inventory=$4

cd $_directory

# Remove the $_tenant files from the /etc/init.d/
ansible sas-casserver-primary -i $_inventory -m file -a
"path=/etc/rc.d/init.d/sas-$_tenant-cascontroller-$_instance state=absent" -b
ansible ComputeServer:programming -i $_inventory -m file -a
"path=/etc/rc.d/init.d/sas-$_tenant-connect-$_instance state=absent" -b
ansible ComputeServer:programming -i $_inventory -m file -a
"path=/etc/rc.d/init.d/sas-$_tenant-runlauncher-$_instance state=absent" -b
ansible ComputeServer:programming -i $_inventory -m file -a
"path=/etc/rc.d/init.d/sas-$_tenant-spawner-$_instance state=absent" -b
ansible ComputeServer:sas-casserver-worker -i $_inventory -m file -a
"path=/etc/rc.d/init.d/sas-$_tenant-cas-$_instance state=absent" -b
ansible sas-casserver-worker -i $_inventory -m file -a
"path=/etc/rc.d/init.d/sas-$_tenant-cas-$_instance-deployment state=absent" -b
ansible sas-casserver-worker -i $_inventory -m file -a
"path=/etc/rc.d/init.d/sas-$_tenant-cas-$_instance-usermods state=absent" -b
ansible ComputeServer:sas-casserver-worker:sas-casserver-primary -i
$_inventory -m file -a "path=/etc/rc.d/init.d/sas-$_tenant-all-services state=absent" -b

# Remove the $_tenant files from the /etc/sysconfig/sas/
ansible ComputeServer -i $_inventory -m file -a
"path=/etc/sysconfig/sas/sas-$_tenant-runlauncher-$_instance state=absent" -b
ansible ComputeServer -i $_inventory -m file -a
"path=/etc/sysconfig/sas/sas-$_tenant-launcher-$_instance state=absent" -b
ansible ComputeServer -i $_inventory -m file -a
"path=/etc/sysconfig/sas/sas-$_tenant-workspaceserver-$_instance state=absent" -b
ansible ComputeServer -i $_inventory -m file -a
"path=/etc/sysconfig/sas/sas-$_tenant-compsrv-$_instance state=absent" -b
ansible ComputeServer:programming -i $_inventory -m file -a
"path=/etc/sysconfig/sas/sas-$_tenant-connect-$_instance state=absent" -b
ansible programming -i $_inventory -m file -a
"path=/etc/sysconfig/sas/sas-$_tenant-spawner-$_instance state=absent" -b
ansible sas-casserver-primary -i $_inventory -m file -a
"path=/etc/sysconfig/sas/sas-$_tenant-cascontroller-$_instance state=absent" -b
ansible sas-casserver-worker -i $_inventory -m file -a
"path=/etc/sysconfig/sas/sas-$_tenant-cas-$_instance state=absent" -b
```

```

ansible sas-casserver-worker -i $_inventory -m file -a
"path=/etc/sysconfig/sas/sas-$_tenant-cas-$_instance-deployment state=absent" -b
ansible sas-casserver-worker -i $_inventory -m file -a
"path=/etc/sysconfig/sas/sas-$_tenant-cas-$_instance-usermods state=absent" -b

# Delete the /opt/sas/{tenant} directory
ansible ComputeServer:programming:sas-casserver*:consul:CASServices
-i $_inventory -m file -a "path=/opt/sas/$_tenant state=absent" -b

# Exit with success
exit 0

```

- 8 Set permissions so that you can run the script. Then run the script:

```
./delete_tenant_directories.sh tenantID tenantinstanceID /path_to_playbook inventory.ini
```

Where *tenantinstanceID* is the value of “tenant_instance” in the *tenant_vars.yml* file, and the *inventory.ini* file is the file referenced in the playbook.

- 9 Disable or remove tenant-based users. Either remove these users from your LDAP environment, or block them with a filter for sssd.

Access to SAS Studio is controlled by OS authentication. The system administrator should remove tenant users from the OS.

- 10 Remove the home directories of these users. The relevant directories are referred in the *homeDirectory* attribute in LDAP.

- 11 Remove DNS mapping for the tenant’s subdomain, if applicable.

The tenant is now offboarded. The tenant logon URL might still be reachable, but no account, including *sasprovider*, can sign on. This acts as a validation step.

Here are additional steps to consider:

- Review any backups for the offboarded tenant. You might want to purge these, or store them in case the tenant needs to be re-onboarded at some future time.
- Run binary and default backups. See “[How to Use the Binary Backup and the Default Backup](#)” on page 138 for more details.

Re-Onboarding a Tenant

Before an offboarded tenant can be re-onboarded, all micro-services must be restarted (after the tenant was offboarded). Note that administrators of this re-onboarded tenant, can restore old backups, unless you have archived and removed them.

Multi-tenancy: Concepts

Multi-tenancy

In multi-tenancy, a provider manages one or more tenants within a single deployment. The essential characteristics of multi-tenancy are separation and sharing.

You cannot convert a single-tenant deployment to a multi-tenant deployment, either through an update or an upgrade. See *SAS Viya for Linux: Deployment Guide* for more details.

Here are key points:

- Each tenant is isolated, with no visibility into other tenants or into the provider. The provider, can see all tenants, but cannot access them. See “[Distribution of Responsibilities](#)” on page 582.
- Many components are shared across tenants. For example, applications are shared across tenants.
- Some components have a dedicated instance for each tenant. For example, each tenant has its own dedicated CAS server.
- A multi-tenant environment can be established only during deployment. You cannot retrofit multi-tenancy into an existing environment. You can add tenants to a multi-tenant deployment at any time.
- Multi-tenancy is not applicable to a programming-only deployment.

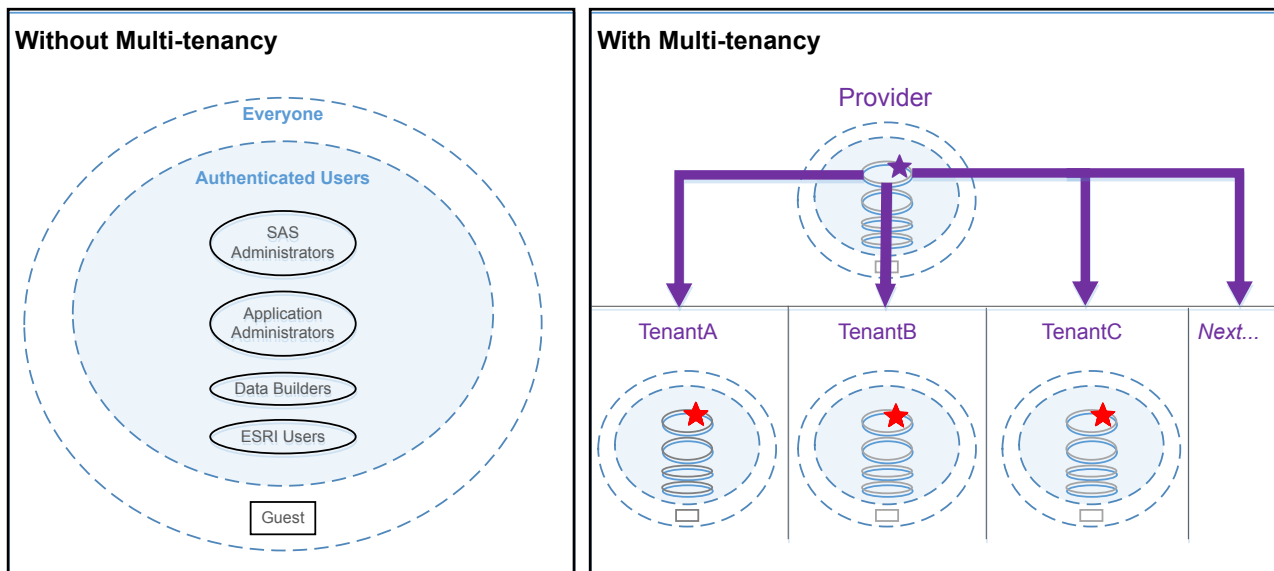
Sub-Domain Navigation

Tenants are accessed via tenant-specific URLs. Provide tenant users and administrators the URL specific to their tenant. The URL has the following format: `tenant.hostname/applicationName` where: *tenant* is the tenant identifier established when you create the tenant, *hostname* is the host name for the deployment, and *applicationName* is the SAS application. Here is an example: `acme.sas.com/SASStudio`.

Parallel Sets of Groups

An identical set of predefined groups and principals exists in the provider and in each tenant. Membership in the provider’s SAS Administrators group enables you to perform provider-level tasks, but does not enable you to sign in to any tenant. Membership in a tenant’s SAS Administrators group enables you to perform intra-tenant tasks for that tenant. The following figure depicts the structure:

Figure A.1 Predefined Groups and Principals



Here are details about the preceding figure:

- Notice that the provider’s groups are the same as the tenants’ groups. The provider is implemented as the initial or default tenant. The provider primarily populates and uses its SAS Administrators group. The provider might also populate its other groups for purposes such as validation and reporting.
- The user who is represented by the purple star is the provider and cannot directly access the tenants. Instead, when a tenant is onboarded, a user, `sasprovider`, is created in the SAS Administrators group for the tenant (represented by the red star).

- A tenant administrator can choose to add its own members to its SAS Administrators group.

Distribution of Responsibilities

Introduction

Certain activities can be performed only from the provider tenant. Such activities are performed by *provider administrators*. Provider-level administration includes tenant management (management of tenants by the provider).

Certain activities can be performed only at the tenant level. Such activities are performed in the tenant, by *tenant administrators* (or by using the sasprovider user). These activities are referred to as intra-tenant administration.

For an overview, see “[Routine Ongoing Tasks](#)” on page 2.

The following sections provide details.

Provider-Level Tasks

The provider-level administrator has additional permissions and responsibilities compared to the intra-tenant administrator. Here are examples of provider-level tasks:

- view or modify deployment resources (Consul key-value pairs, system configuration properties, system component status, and so on)
- modify LDAP connection information for the LDAP identity provider
- manage product licenses
- access application and server logs
- perform machine-level monitoring
- modify contents of LDAP
- perform provider-level backup and recovery
- perform tenant management tasks

Intra-tenant Tasks

Here are tasks that an administrator within a tenant can perform:

[Create and restore tenant-level backups](#)

[Add caslibs](#)

[Manage access to data](#)

[Manage access to content](#)

[Manage access to functionality](#)

[Promote content](#)

[Assign users to custom groups](#)

[Manage mobile devices](#)

Guest Access in Multi-tenancy

This topic supplements the general information and instructions in [“Authentication: Guest Access” on page 57](#).

Here are details for a provider who wants to enable guest access:

- Manage the guest-related configuration property (`sas.logon.provider.guest`) as follows:

Goal	Instructions
Make guest access available to selected tenants.	Instruct each participating intra-tenant administrator to add and enable <code>sas.logon.provider.guest</code> within its environment. Do not set that property in the provider.
Make guest access available to the provider and all tenants.	Add and enable <code>sas.logon.provider.guest</code> in the provider. Clear the Apply configuration only to this tenant (provider) check box. Each tenant can opt out from within its own environment.
Make guest access available to the provider and selected tenants.	Add and enable <code>sas.logon.provider.guest</code> in the provider. Leave the Apply configuration only to this tenant (provider) check box selected. Each tenant can opt in from within its own environment.

- Run the following command for each participating tenant (and for the provider, if applicable):

```
sas-admin authorization facilitate-guest
```

TIP To target a tenant, specify that tenant's URL in your CLI profile. When you sign in, you must supply credentials that are valid for that tenant (for example, the `sasprovider` user created during onboarding).

- Make sure any necessary host-layer access controls are in place. See [“Data Separation in Multi-tenancy” on page 583](#).
- Instruct each participating intra-tenant administrator to use SAS Environment Manager to make any additional adjustments to guest access within its own environment.

Here are details for a provider who wants to disable guest access:

- To disable guest access, make sure that the `sas.logon.provider.guest` property is disabled in each appropriate environment.
- If you want to remove rules and access controls that were added by the `facilitate-guest` commands, follow the general instructions in each appropriate environment.

Data Separation in Multi-tenancy

CAS Access Controls

You do not need to use CAS access controls to separate data across tenants. Caslibs and tables are inherently separated for per-tenant, isolated use. Here are details:

- Each tenant has its own CAS server.
- Each CAS server can be accessed only by users from the associated tenant. This restriction is established by the `tenantid` option.

- Each CAS server has its own list of Superusers.

Host Access Controls

You must use host access controls to separate data across tenants. Host resources are not inherently separated for per-tenant, isolated use. Without host-layer constraints, each CAS server on a machine can access all directories and files on that machine. Here are details:

- For visual interface users who are not members of the `CASHostAccountRequired` group, host access is under the CAS server's account. Each CAS server runs under a distinct, tenant-specific account. Set host layer access controls so that each server account can access only those directories and files that are appropriate for that tenant.
- For programmers and members of the `CASHostAccountRequired` group, host access from CAS is under each user's account. Set host layer access controls so that each user's account can access only those directories and files that are appropriate for that user.

TIP You can use a per-server blacklist or whitelist to limit the available host paths for caslib creation. However, Superusers (and any members of the Data role) are not subject to such constraints. See “[Paths List](#)” on page 620.

Onboarding Tenants

The process of onboarding a tenant consists of creating a tenant and making it available to its intended users. Onboarding is achieved by creating a `tenant_vars.yml` file and then running an Ansible play.



Offboarding Tenants

The process of offboarding a tenant consists of removing access to the tenant, stopping scheduled jobs from running on the tenant, stopping services that run on the tenant, dropping the database schemas for the tenant, removing the tenant configuration, and deleting the tenant resources. Unlike onboarding, offboarding is not an Ansible play, but rather a sequence of commands and scripts.

Tenant Management: Interfaces

In the following table, the shaded part of each circle is an approximation of the amount of tenant administration functionality that a particular interface exposes. The shading indicates relative coverage.

Table A.46 *Interfaces to Tenant Administration*

	Tenants page	The enterprise graphical interface in SAS Environment Manager
	Command-line interface	A simple, scriptable interface.

Multi-tenancy: Troubleshooting

Tenant Administrator not Created during Onboarding

If the tenant administrator is not set up correctly when running the `tenant_vars.yml` play, the provider must manually add the user to the tenant's SASAdministrators group. This is done by accessing SAS Environment Manager for the tenant using the appropriate sasprovider account. Navigate to the **Users** view, and add the tenant users to the SASAdministrators group.

Unable to Access the Tenant URL

If you receive a 404 Not Found message when attempting to log in to the tenant (for example, `tenantID.hostname/SASEnvironmentManager`), and the navigation bar shows `tenantID.hostname/SASLogon`, your `zones.internal.hostnames` property is not configured correctly.

This property must contain the base host name without any tenant prefixes. For example, if tenants are addressed as `tenant1.example.com`, `tenant2.example.com`, and so on, the property must contain `example.com`. Also include any default host names for the provider (`localhost` is already included).

The `saslogon` service must be restarted when making changes to this property.

Tenant Remains in an Onboarding State

If a tenant stays too long in an onboarding state, log in to SAS Environment Manager (as an administrator on the provider's tenant), and navigate to the **Tenants** view. For the specific tenant in question, select the **Services Status** option, and determine which services have a status of **Onboarding**. For these services, view the appropriate log file, and look for any errors or warnings related to the onboarding process.

Scheduling

Scheduling: Overview

The Scheduling page enables you to schedule jobs to run at a particular time or in response to a specific trigger. You can run a job immediately, or you can specify a time interval (from daily to yearly) to control when the job runs. You can also unschedule, delete, and view the properties of jobs.

Jobs that are available for scheduling are from these sources:

SAS Data Explorer

Creates jobs that you can schedule using SAS Environment Manager.

SAS Data Studio

Creates jobs that you can schedule using SAS Environment Manager.

SAS Visual Analytics


Creates jobs that are scheduled in SAS Visual Analytics. You can view and modify the schedules in SAS Environment Manager.

CAS table state management

Three jobs are provided to manage CAS tables.


- Import cas-shared-default Public data
- Load cas-shared-default Public data
- Unload cas-shared-default Public data

You can schedule these jobs, but you cannot delete them, and you can modify the job options only on copies of the jobs. If you schedule one of these jobs and then make a copy of the job, only the job is copied, not any triggers that are associated with the job.

To access the Scheduling page, click  **Scheduling** in the SAS Environment Manager navigation menu.

Scheduling: How to (SAS Environment Manager)

Schedule a Job

- 1 On the Scheduling page, select a job in the **Jobs** table.
- 2 Click  in the toolbar or select **Schedule** from the pop-up menu.
- 3 (Optional) To run the job under credentials other than your own, in the Schedule Job window, specify the user ID under which the job should be run in the **Run as** field. Click to select from defined identities. The user that you select must have previously signed in to SAS since it was installed.

- 4 Activate the **Enabled** control for one or more triggers in the **Available triggers** table. A trigger controls when the job runs. See [“Create a Time Trigger” on page 588](#) to define a new trigger. You can use a trigger only with the job for which it was created.

Note: Currently, **Time Event** is the only supported trigger type.

- 5 Click **Save**.
- 6 Verify that the listing for the job in the **Jobs** table contains a check mark in the **Scheduled** column.

Run a Job

- 1 On the Scheduling page, select a job in the **Jobs** table.
- 2 To run the job under your own credentials, click ► in the toolbar or select **Run** from the pop-up menu.
- 3 To run the job under credentials other than your own, select **Run As** from the pop-up menu.

The Select Identities window appears, and you can select the user ID under which the job should run.

Note: The user ID that you select must have previously signed in to SAS.

You can run a job regardless of whether it has been scheduled.

Create a Time Trigger

- 1 In the Schedule Job window, click + above the **Available triggers** table.
- 2 In the New Trigger window, assign a name to the new trigger.
- 3 Use the **Frequency** field to specify how often the trigger should be repeated (such as a specified number of minutes, hours, or days).
- 4 Depending on your choice for the frequency interval, different fields appear in the window to enable you to completely specify a frequency for the trigger. For example, if you select **Yearly** in the **Frequency** field, you can specify a day of a month (such as the first of January), the last day of a month, or a specific weekday in a month (such as the third Thursday in February). If you specify **Minutes** in the **Frequency** field, you can specify that the job runs every 5, 10, 15, 20, or 30 minutes. Use these fields to specify the criteria for the trigger interval.

Note: If you select **Date List** in the **Frequency** field, you cannot select a date more than once.

- 5 In the **Start time** field, specify when the job schedule should start. Click the entry in the **Start time** field to select a time.

For example, if you use the **Frequency** fields to specify that the job runs every hour, and you specify **10:15** in the **Start time** field, the job runs at 10:15, 11:15, 12:15, and so on. If you use the **Frequency** fields to specify that the job runs every 20 minutes, and you specify **09** in the **Start time** field, the job runs at 9:00, 9:20, 9:40, and so on.

- 6 Specify the time zone to use when evaluating the time for the trigger, and the date on which the trigger starts.


Note: If you choose **Date List** in the **Frequency** field, you must select the same value in the **Time zone** field for every scheduled date.

- 7 Specify when the trigger ends. You can specify that the trigger never ends, that it ends after a certain number of times, or that it ends on a specific date.
- 8 Click **Save**.

- 9 Repeat these steps to create other triggers for the job.

Edit a Scheduled Job

After a job is scheduled, you can edit the schedule or the parameters for the job. Follow these steps:

- 1 Select a scheduled job in the **Jobs** table on the Scheduling page. Scheduled jobs with at least one enabled trigger contain a check mark in the **Scheduled** column. Scheduled jobs with disabled triggers contain a disabled icon **II** in the **Scheduled** column.
- 2 To modify the schedule for the job, click  or select **Edit Schedule** from the pop-up menu. In the Edit Schedule window, you can add, edit, and remove triggers for the job. Click **Save** when you have finished modifying the schedule

Filter the List of Jobs

Use the fields in the **Jobs Filter** area to narrow the list of jobs that are displayed in the **Jobs** table. You can specify a date range for when jobs were created and modified.


You can also use the **Search Name** field to search for a specific job name.

Jobs remain in the list on the Scheduling page unless you delete them.


Manage Jobs

Disable the Schedule for a Job


To prevent a job from running its defined schedule, you can either unschedule the job or disable the triggers. Unscheduling the job prevents the job from running on the defined schedule and also removes the triggers that are defined for the job. Disabling the triggers prevents the job from running the schedule, but preserves the defined triggers.

To unschedule a job, select a scheduled job in the **Jobs** table in the Scheduling window. Click  from the toolbar or select **Unschedule** from the pop-up menu.

CAUTION! When you unschedule a job, any enabled triggers that are associated with the job are deleted. To unschedule a job and keep the triggers, instead of selecting **Unschedule**, edit the schedule and manually disable the triggers.

To disable the triggers, select a scheduled job in the **Jobs** table in the Scheduling window. Click  from the toolbar or select **Edit Schedule** from the pop-up menu. In the Edit Schedule dialog box, disable all slider controls in the **Enabled** column of the **Available triggers** table. If you disable all triggers for a job, the disabled icon **II** appears in the **Scheduled** column of the **Jobs** table. A check mark appears in the column if any of the triggers for the job are enabled.

View Job Properties

To view properties for a job, select a job in the **Jobs** table and click  in the toolbar or select **Properties** from the pop-up menu. The information in the Job properties window is read-only.

Delete a Job

Jobs remain in the list on the Scheduling page unless you delete them. To delete a scheduled job, follow these steps.

- 1 Select a job in the **Jobs** table.

- 2 Click  in the toolbar or select **Delete** from the pop-up menu.

Note: You cannot delete any of the provided CAS table state management jobs (Import cas-shared-default Public data, Load cas-shared-default Public data, and Unload cas-shared-default Public data).

Scheduling: How to (Command Line Interface)

Run a Job

In order to run a job using the command-line interface, follow these steps:

- 1 Create a template file for the job definition. This file contains the fields that are needed for a job definition.

```
sas-admin job definitions generate-template --template-filename
```

Here is an example of the job definition template:

```
template:
{
  "name": "Replace with name of the Job Definition",
  "type": "Replace with type of the Job Definition",
  "code": "Replace with code of the Job Definition"
}
```

- 2 Modify the job definition template file to supply information for the job that you want to run.
- 3 Use the job definition file that you created in the previous step to create the job definition.

```
sas-admin job definitions create --definition-filename
```

This command returns a URI for the job definition.

- 4 Create a template file for the job request. This file contains the fields that are needed for a job request.

```
sas-admin job requests generate-template --template-filename
```

Here is an example of the job request template file:

```
template:
{
  "version": 0,
  "name": "Replace with name of the Job Request",
  "description": "Replace with description of the Job Request",
  "jobDefinitionUri": "(Mutually exclusive with Definition) Replace with uri to the Job Definition",
  "arguments": null,
  "properties": null
}
```

- 5 Modify the generated job request template file to supply information for the job that you want to run.
- 6 Use the job request file that you created in the previous step to create the job request.

```
sas-admin job requests create --request-filename
```

The command returns an ID for the job request.

- 7 Execute the request. The job runs immediately.

```
sas-admin job requests execute --request-ID
```

Schedule a Job

In order to schedule a single job using the command-line interface, follow these steps:

- 1 Create a template file for the job definition. This file contains the fields needed for a job definition.

```
sas-admin job definitions generate-template --template-filename
```

Here is an example of the job definition template:

```
template:
{
  "name": "Replace with name of the Job Definition",
  "type": "Replace with type of the Job Definition",
  "code": "Replace with code of the Job Definition"
}
```

- 2 Modify the job definition template file to supply information for the job that you want to run.
- 3 Use the job definition file that you created in the previous step to create the job definition.

```
sas-admin job definitions create --definition-filename
```

This command returns a URI for the job definition.

- 4 Create a template file for the job request. This file contains the fields needed for a job request.

```
sas-admin job requests generate-template --template_filename
```

Here is an example of the job request template:

```
template:
{
  "version": 0,
  "name": "Replace with name of the Job Request",
  "description": "Replace with description of the Job Request",
  "jobDefinitionUri": "(Mutually exclusive with Definition) Replace with uri to the Job Definition",
  "arguments": null,
  "properties": null
}
```

- 5 Modify the generated job request template file to supply information for the job that you want to run.
- 6 Use the job request file that you created in the previous step to create the job request.

```
sas-admin job requests create --request-filename
```

The command returns an ID of the job request.

- 7 Create a JSON file and include the time triggers for the job. See [“Time-Based Triggers” on page 592](#) for information about defining the triggers.
- 8 Schedule the job, and specify the file that contains the time triggers.

```
sas-admin job requests schedule --triggers-file
```

Scheduling: Command-Line Interface Reference

Time-Based Triggers

Use the following syntax when defining a time-based trigger to schedule a flow or a job.

Here is the general form of the syntax:

```
"triggers": [
  {
    "type" : "timeevent",
    "active":true,
    "event": { "recurrence":{"type":"recurrence-type"}, options,
              "hours":hours,
              "minutes":minutes,
              "duration":duration,
              "timeZone":zone,
              "maxOccurrence":occurrences, }
  }
],
```

This list identifies the options that are used for each type of `recurrence` interval.

Minutes

Type

```
"type": "minutely"
```

Options

- `startDate` (specifies when to start the recurrence)
- `endDate` (specifies when to stop the recurrence)
- `skipCount` (specifies how many minutes pass between executions) For example, `"skipCount": "15"` specifies that the job runs every 15 minutes. Valid values are 1, 2, 3, 4, 5, 6, 10, 12, 15, 20, and 30.
- `minutes` (specifies the offset from the beginning of the hour for when the executions start) The maximum value is `skipCount-1`. For example, `"minutes": "10"` specifies that the timing for `skipCount` starts at 10 minutes past the hour.

Here is an example that specifies a trigger that starts at six minutes past the hour, and then causes a job to run every 15 minutes (06, 21, 36, 51, and so on).

```
"triggers": [
  {
    "type" : "timeevent",
    "active":true,
    "event": { "recurrence":{"type":"minutely"}, "skipCount":"15"}
              "minutes":"6", }
  }
],
```

Hours

Type

```
"type": "hourly"
```

Options

- `startDate` (specifies when to start the recurrence)
- `endDate` (specifies when to stop the recurrence)

- `skipCount` (specifies how many hours pass between executions) For example, `"skipCount": "3"` specifies that the job runs every 3 hours. Valid values are 1, 2, 3, 4, 6, 8, and 12.

Here is an example that specifies a trigger that starts on June 13, 2018, and causes a job to run every 4 hours:

```
"triggers": [
  {
    "type": "timeevent",
    "active": true,
    "event": {
      "recurrence": {
        "type": "hourly",
        "startDate": "2018-06-13",
        "skipCount": "4"
      }
    }
  },
]
```

Days

Type

```
"type": "daily"
```

Options

- `startDate` (specifies when to start the recurrence)
- `endDate` (specifies when to stop the recurrence)
- `skipCount` (specifies how many days pass between executions) For example, a value of 3 specifies that the job runs every 3 days.
- `daysOfWeek` (specifies the days on which the job runs) For example, `"daysOfWeek": "thursday"` specifies that the job runs every Thursday. Valid values are names of days: monday–sunday.

Here is an example of a trigger:

```
"triggers": [
  {
    "type": "timeevent",
    "active": true,
    "event": {
      "recurrence": {
        "type": "daily",
        "hours": "12",
        "minutes": "0"
      }
    }
  },
]
```

Weeks

Type

```
"type": "weekly"
```

Options

- `startDate` (specifies when to start the recurrence)
- `endDate` (specifies when to stop the recurrence)
- `skipCount` (specifies how many weeks pass between executions) For example, `"skipCount": "3"` specifies that the job runs every 3 weeks.
- `daysOfWeek` (specifies the days on which the job runs) For example, `"daysOfWeek": "thursday"` specifies that the job runs every Thursday. Valid values are names of days (monday–sunday).

Example:

```
"triggers": [
  {
    "type": "timeevent",
    "active": true,
    "event": {
      "recurrence": {
        "type": "weekly",
        "startDate": "2017-08-15",
        "skipCount": "4",
        "daysOfWeek": "tuesday"
      }
    }
  },
]
```

Months

Type

"type": "monthly"

Options

- startDate (specifies when to start the recurrence)
- endDate (specifies when to stop the recurrence)
- skipCount (specifies how many months pass between executions) For example, "skipCount": "3" specifies that the job runs every 3 months.
- daysOfWeek (specifies the days on which the job runs) For example, "daysOfWeek": "thursday" specifies that the job runs on Thursday. Valid values are names of days (monday-sunday). daysOfWeek and dayOfMonth are mutually exclusive. If daysOfWeek is specified, then occurrence is required.
- occurrence (used with daysOfWeek to specify the days on which the job runs) Valid values are first, second, third, fourth, and last.
- dayOfMonth (specifies the day of the month to run on. Valid values are 1–31. A value of 32 specifies the last day of the month. dayOfMonth and daysOfWeek are mutually exclusive.

Example 1:

```
"triggers": [
  {
    "type": "timeevent",
    "active": true,
    "event": {
      "recurrence": {
        "type": "monthly",
        "startDate": "2017-05-14",
        "daysOfWeek": "sunday",
        "occurrence": "second"
      }
    }
  },
  ]
```

Example 2:

```
"triggers": [
  {
    "type": "timeevent",
    "active": true,
    "event": {
      "recurrence": {
        "type": "monthly",
        "dayOfMonth": "15"
      }
    }
  },
  ]
```

Years

Type

"type": "yearly"

Options

- startDate (specifies when to start the recurrence)
- endDate (specifies when to stop the recurrence)
- skipCount (specifies how many years pass between executions)
- daysOfWeek (specifies the days on which the job runs) For example, "daysOfWeek": "thursday" specifies that the job runs on Thursday. Valid values are names of days (monday-sunday). daysOfWeek and dayOfMonth are mutually exclusive. If daysOfWeek is specified, then occurrence is required.
- occurrence (used with daysOfWeek to specify the days on which the job runs) Valid values are first, second, third, fourth, and last.
- dayOfMonth (specifies the day of the month to run on) Valid values are 1–31. A value of 32 specifies the last day of the month. dayOfMonth and daysOfWeek are mutually exclusive.
- monthOfYear (specifies the month in which the job run) Valid values are january–december.

Example 1:

```
"triggers": [
  {
    "type": "timeevent",
    "active": true,
    "event": {
      "recurrence": {
        "type": "yearly",
        "daysOfWeek": "friday",
        "occurrence": "last",
        "monthOfYear": "june"
      }
    }
  }
],
```

Example 2:

```
"triggers": [
  {
    "type": "timeevent",
    "active": true,
    "event": {
      "recurrence": {
        "type": "yearly",
        "dayOfMonth": "32",
        "monthOfYear": "may"
      }
    }
  }
],
```

Specified dates

Type

```
"type": "dateList"
```

Options

- `startDate` (specifies when to start the recurrence)
- `endDate` (specifies when to stop the recurrence)
- array of dates in the form `yyyy '-' mm '-' dd`

Example:

```
"triggers": [
  {
    "type": "timeevent",
    "active": true,
    "event": {
      "recurrence": {
        "type": "dateList",
        "2017 '-' 06 '-' 13",
        "2017 '-' 08 '-' 02",
        "2017 '-' 10 '-' 05"
      }
    }
  }
],
```

For the `"hours": hours` parameter, specify a set of hours when the job runs. You can specify a list of hours separated by commas (1,2,3), a range of hours (2–4), a combination of a range and a list (1–3,5,7), or an asterisk to specify all hours. If you specify a recurrence of hourly, only the first value is used, and it must be equal to or less than the value of `skipCount`. If you specify a recurrence of minutely, the hours parameter is ignored.

For the `"minutes": minutes` parameter, specify a set of minutes when the job runs. You can specify a list of minutes separated by commas (0,10,30), a range of minutes (20–25), a combination of a range and a list (0,10–15), or an asterisk to specify every minute. If you specify a recurrence of minutely, only the first value is used, and it must be equal to or less than the value of `skipCount`.

For the `"duration": duration` parameter, specify the number of minutes the event is to remain true.

For the `"timeZone": zone` parameter, specify the time zone to use when evaluating the time trigger. Specify the value using the Olson time zone database, in the form `region/city`. For example, `America/New_York`.

For the `"maxOccurrence": occurrences` parameter, specify the maximum number of times the job can execute.

General Services

General Servers and Services: Overview

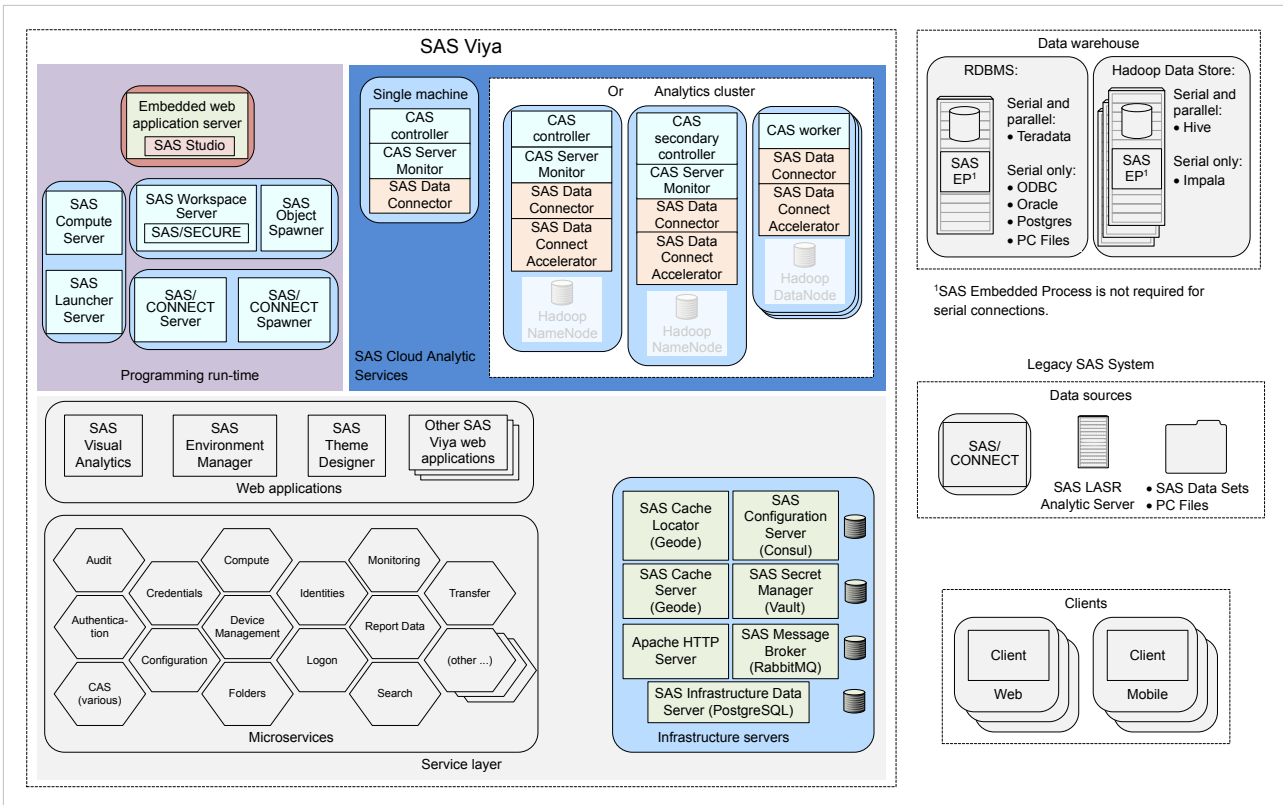
Servers

SAS Viya contains these servers:

- [SAS Cloud Analytic Services](#)
- Programming run-time servers:
 - [SAS Compute Server](#)
 - [SAS Launcher Server](#)
 - [SAS Workspace Server and SAS Object Spawner](#)
 - [SAS/CONNECT Server and SAS/CONNECT Spawner](#)
 - [embedded SAS Web Application Server](#)
- Infrastructure servers:
 - [SAS Cache Locator and SAS Cache Server \(Geode\)](#)
 - [SAS Configuration Server \(Consul\)](#)
 - [SAS Secret Manager \(Vault\)](#)
 - [Apache HTTP Server](#)
 - [SAS Message Broker \(RabbitMQ\)](#)
 - [SAS Infrastructure Data Server \(PostgreSQL\)](#)

Note: A [programming-only deployment on page 1](#) includes only one of the infrastructure servers: Apache HTTP Server.

Figure A.1 SAS Viya Servers



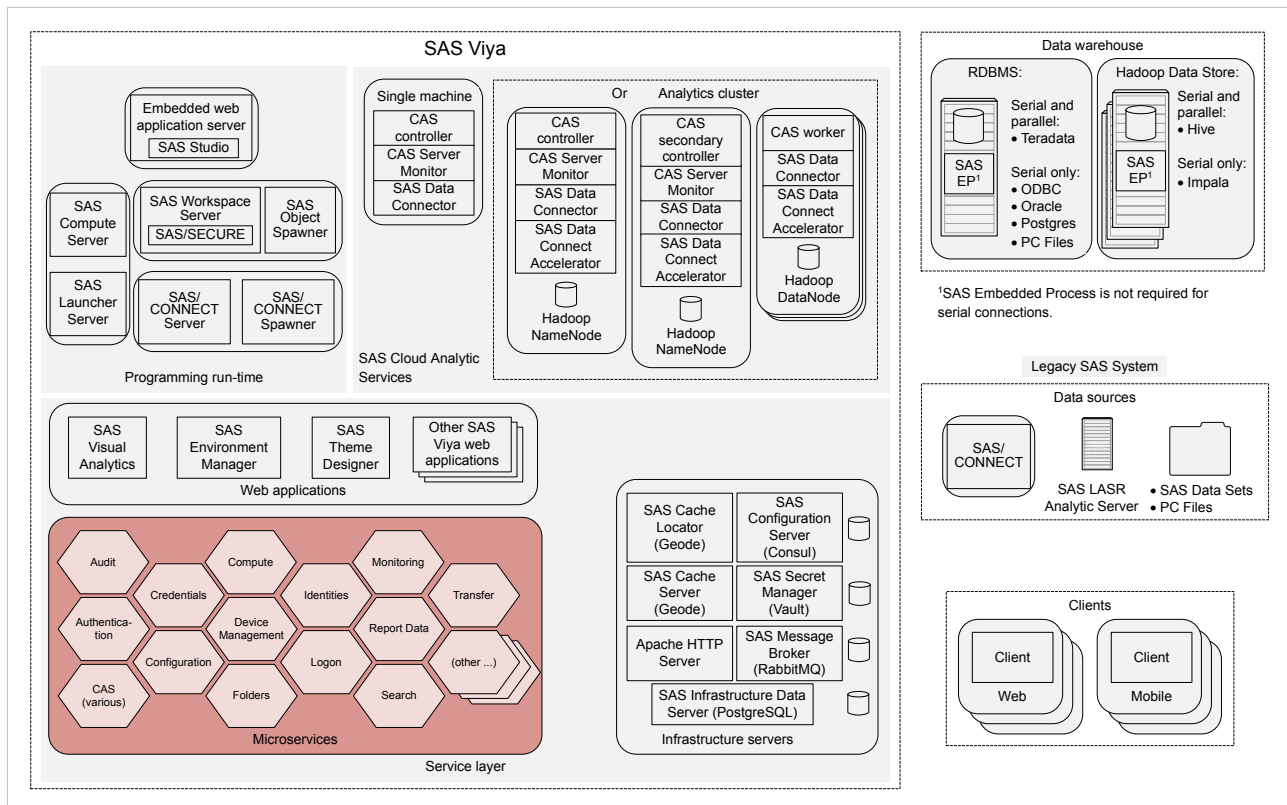
Services

SAS Viya contains several services often referred to as *microservices*. A microservice is a small service that runs in its own process and that communicates with a lightweight mechanism (HTTP).

SAS Viya includes services such as, Audit, Identities, and Monitoring. To see the complete list of SAS Viya services follow the initial steps in [“Edit Configuration Instances”](#) on page 215.

Note: A [programming-only deployment on page 1](#) does not use most SAS Viya services.

Figure A.2 SAS Viya Services



General Servers and Services: Operate

All Servers and Services

SAS Viya uses the operating system’s default init system command to launch a script that can stop, start, and check the status of all SAS Viya servers and services. This script, `sas-viya-all-services`, resides in `/etc/init.d`.

Note: You must be logged on to the machine where the SAS Viya servers and services reside, and you must have sudo privileges to run this script.

To operate all SAS Viya servers and services, run the following command, as appropriate:

```
sas-viya-all-services status | stop | start
```

Note: `sas-viya-all-services` does not control [Apache HTTP Server](#).

TIP On multi-tenant SAS Viya systems, the `sas-viya-all-services` script is named `sas-tenant-ID-all-services`.

Note: When checking status, it is normal for certain servers and services to not display host, port, and PID information. The reason is that these servers and services are not registered with the SAS Configuration Server, including the configuration server itself.

Here are a few examples of how to operate this script:

Note: When running `sas-viya-all-services` Red Hat Linux version 7, users should not use the `systemd` command, but instead use the `init` system command. (When running [individual service scripts](#), Red Hat Linux version 7 users should use the `systemd` command.)

- To check status of all servers and services using a direct call:

```
sudo /etc/init.d/sas-viya-all-services status
```

- To stop all servers and services using the Red Hat Linux version 6 `init` system command:

```
sudo service sas-viya-all-services stop
```

- To start all servers and services using the Red Hat Linux version 6 `init` system command:

```
sudo service sas-viya-all-services start
```

TIP When stopping the entire SAS Viya environment, be sure to run `sas-viya-all-services` on every SAS Viya machine. Running `sas-viya-all-services` ensures that the servers and services are stopped in the correct order.

A Particular Server or Service

SAS Viya uses the operating system's default `init` system or `systemd` command to launch scripts that can stop, start, restart, and check the status of servers and services. These scripts reside in `/etc/init.d`.

Note: You must be logged on to the machine where the compute service resides, and you must have `sudo` privileges to run these scripts.

To operate a particular SAS Viya server or service, run the following command, as appropriate:

```
sas-viya-server-or-service-default status | stop | start | restart
```

TIP To see the complete list of SAS Viya server and service scripts, run the following command: `ls /etc/init.d/sas-viya-*`. To operate Apache HTTP Server, see [“Operate”](#).

Here are a few examples of how to operate these scripts:

Note: On Red Hat Linux version 7 systems, use the `systemd` command when running the individual service and server scripts. The `systemd` command maintains a record of service status that the `init` system command and a direct call does not use.

- To check status of SAS Logon Manager using a direct call:

```
sudo /etc/init.d/sas-viya-saslogon-default status
```

- To stop the Comments service using the Red Hat Linux version 6 `init` system command:

```
sudo service sas-viya-comments-default stop
```

- To start SAS Configuration Server using the Red Hat Linux version 7 `systemd` command:

```
sudo systemctl start sas-viya-consul-default
```

- To restart the Cross Domain Proxy service using a direct call:

```
sudo /etc/init.d/sas-viya-crossdomainproxy-default restart
```


General Servers and Services: Locale and Encoding

SAS Viya 3.3 supports all the SAS session encodings that are available in SAS 9.4. By default, SAS Viya 3.3 uses an encoding of `UTF-8` and a locale of `en_US`. You can change the SAS `LOCALE` option, the `ENCODING` option, or both options.

Change the SAS Locale

- 1 Sign in to the machine on which SAS Foundation resides as the SAS install user or with a user account that has sudo privileges.

Note: SAS Workspace Server automatically uses the locale that matches the locale that is sent by the client. The `LOCALE` option value from `sasv9.cfg` and `sasv9_local.cfg` does not affect a SAS Studio session. If a locale is set in the `sasv9_local.cfg` file, that locale is set for SAS programs that are run in batch mode.

- 2 Modify `/opt/sas/spre/home/SASFoundation/sasv9_local.cfg` with a new line that contains the following:

```
locale=five-character-POSIX-locale-code
```

Here is an example:

```
locale=fr_CA
```

For valid POSIX locale codes, see [five-character POSIX locale codes](#).

Consider these tips about using the `LOCALE` option:

- You can override the `LOCALE` option setting for your session by setting the `LOCALE` option on the command line.
- You can change the `LOCALE` option during your SAS session by setting the `LOCALE` option in the `OPTIONS` statement.

Change the SAS Encoding

- 1 Sign in to the machine on which SAS Foundation resides as the SAS install user or with a user account that has sudo privileges.
- 2 Change to the `/opt/sas/spre/home/SASFoundation` directory, and update the symbolic link for `sas` to point to the new encoding configuration file:

```
cd /opt/sas/spre/home/SASFoundation
```

```
ln -sf bin/sas_nn sas
```

where `nn` is the two- or four-character code that supports the SAS encoding.

Note: Encoding configuration files reside in `/opt/sas/spre/home/SASFoundation/bin`.

Here is an example:

```
cd /opt/sas/spre/home/SASFoundation
```

```
ln -sf bin/sas_en sas
```

- 3 In the shell environment on the SAS Foundation machines and in the `sasv9_usermods.cfg` file for the server, modify the `LANG` environment variable to match the new `LOCALE` and `ENCODING` option values:

```
LANG=five-character-POSIX-Locale-code.Linux-encoding-string; export LANG
```

The `sasv9_usermods.cfg` file resides in `/opt/sas/viya/config/etc/server/deployment-instance` for each server.

Here is an example:

```
/opt/sas/viya/config/etc/workspaceserver/default
```

For valid POSIX locale codes and Linux encoding strings, see [five-character POSIX locale codes and Linux encoding strings](#).

Note: The LANG environment variable setting must match the locale and encoding that you plan to select for SAS Foundation and SAS Workspace Server.

Here is an example:

```
LANG=ja_JP.eucjp; export LANG
```

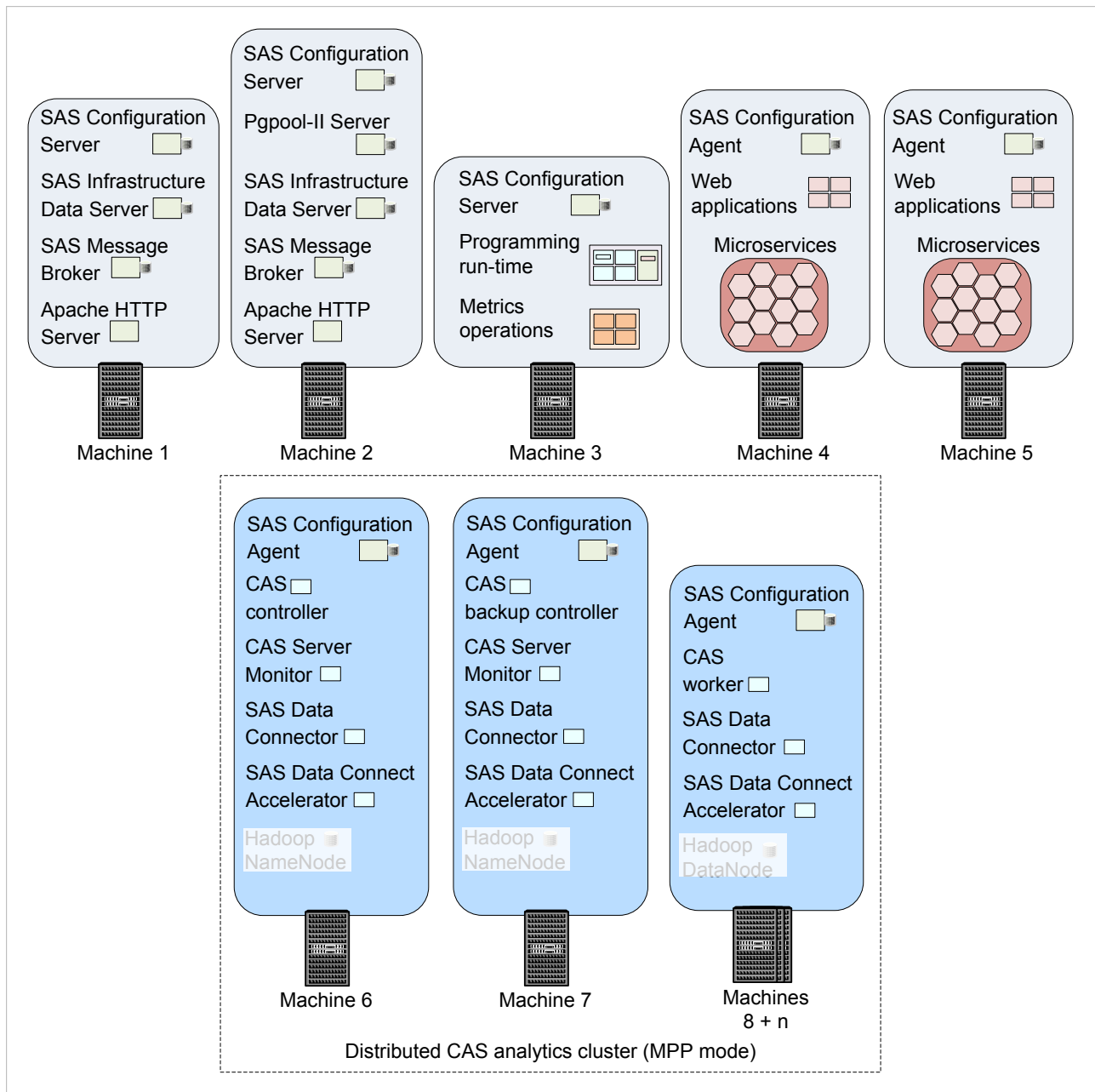
Note: The `export` command (`export LANG`) is not needed when modifying the `sasv9_usermods.cfg` file.

Note: On Linux, there are differences in the spelling and the casing of language-encoding pairs. For this reason, you should run the `locale` command to check the current locale and to verify the spelling of locale values. A misspelling causes the LANG environment variable to be improperly set, and it interferes with propagation to other locale-related environment variables.

Fault Tolerance in SAS Viya

The following figure shows the minimum recommended fault tolerant SAS Viya configuration on eight machines.

Figure A.3 Minimum Requirements for a Fault Tolerant SAS Viya Deployment



SAS Configuration Server (Consul) is unique because it maintains a quorum voting structure among members. Three SAS Configuration Server instances (Machines 1–3) are the minimum number that is required to provide fault tolerance. For more information, see <https://www.consul.io/docs/internals/consensus.html>.

The **programming run-time servers** reside on Machine 3 along with **metrics operations**. Metrics operations use an infrastructure that supports monitoring and logging. This infrastructure can be deployed on one machine because the programming run-time and metrics operations are not fault tolerant.

In this sample deployment, the **infrastructure servers** are deployed on separate machines from the web applications and the microservices (Machine 4 and Machine 5) to enhance performance. (Machine 5 provides fault tolerance for Machine 4.)

For **Cloud Analytic Services (CAS)** that are running in MPP mode and are distributed across an analytics cluster, unique forms of fault tolerance are available for each machine type: CAS worker and CAS controller.

Because there are more CAS worker machines than CAS controller machines, worker machines are more likely to experience failure. Fault tolerance is provided automatically for worker machines that contain CAS tables, which are created with redundant copies of blocks.

The less common CAS controller failure problem is addressed with an optional CAS backup controller (also referred to as the *secondary controller*). For more information, see [“Fault Tolerance” on page 624](#).

In this sample deployment, the CAS controller is deployed on Machine 6, its backup controller is deployed on Machine 7, and CAS workers are on Machines 8 + n.

Each machine runs a SAS Configuration Server agent process that performs health checks on the SAS Viya services that are running and on the machine itself. Each configuration agent provides health information to one or more configuration servers. In fault tolerant deployments, configuration servers choose a leader and store and replicate service information.

SAS Viya microservices send queries to configuration servers or configuration agents in order to discover other services. Every configuration agent has its own copy of service discovery information.

General Servers and Services: Troubleshooting

Starting sas-viya-consul-default

Timed out waiting for the consul service to start Exiting

Explanation:

The SAS Configuration Server (Consul) is not starting. One cause might be that certain configuration files are corrupted.

Resolution:

- 1 Check `/opt/sas/viya/config/data/consul/checks/`. Delete all zero-length files.
- 2 Check `/opt/sas/viya/config/data/consul/services/`. Delete all zero-length files.
- 3 [Restart all services](#).
- 4 If the configuration server still fails to start, delete all files in `/opt/sas/viya/config/data/consul/checks/` and `/opt/sas/viya/config/data/consul/services/` and restart all services again.

One or more SAS Viya microservices fail to start up

UnknownHostException: rabbitmq.service.consul

Explanation:

All microservices use the same API to publish and receive events from SAS Message Broker (RabbitMQ). The microservices are attempting to fetch one or more message broker server hostnames from SAS Configuration Server (Consul) but that information is not registered correctly because of missing information in `/etc/hosts`.

Resolution:

Make sure that `/etc/hosts` contains every machine name in your SAS Viya deployment, and that `/etc/hosts` has been copied to every machine in your SAS Viya system.

sas-viya-all-services status command returns ‘not ready’

Explanation:

Machines in your SAS Viya deployment are defined in `/etc/hostname` with a short host name.

Resolution:

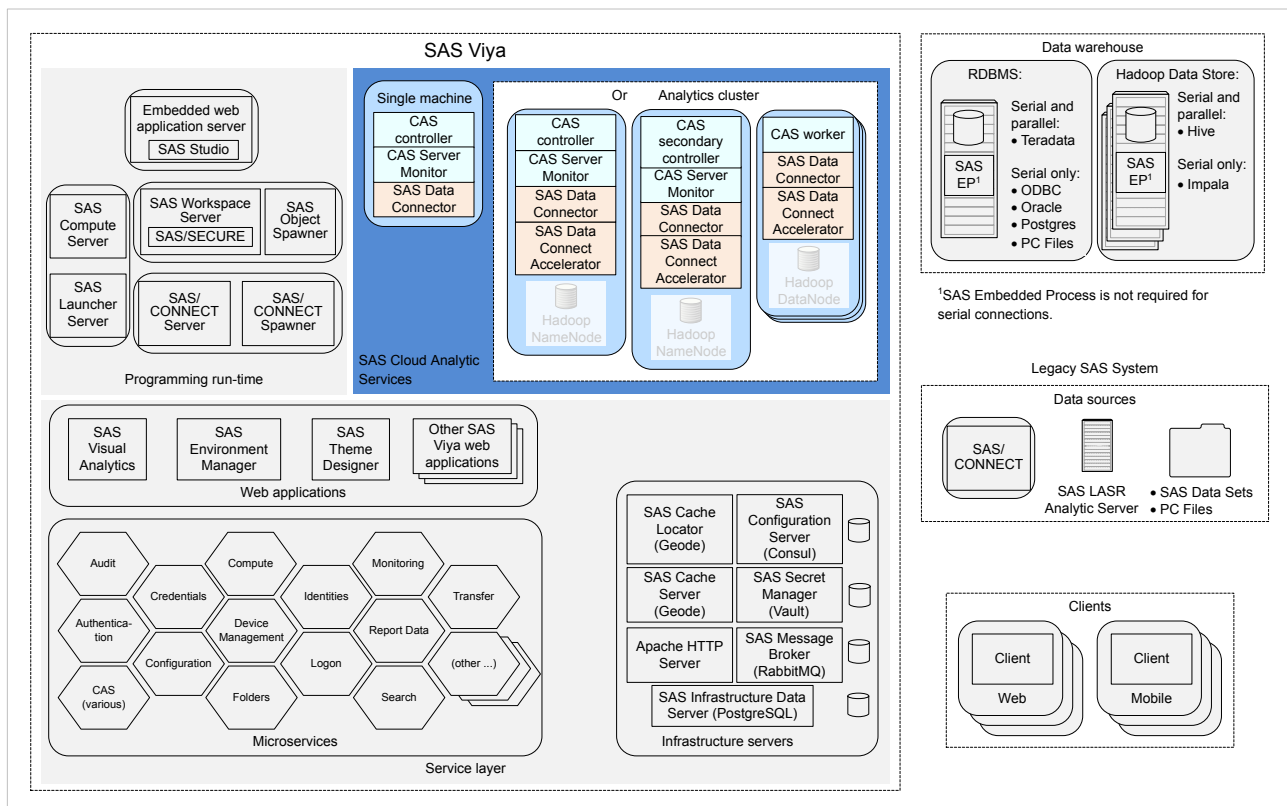
Using the Linux `hostname` command, redefine the machines in your SAS Viya deployment using their fully qualified domain names (for example, `my-machine.example.com`).

SAS Cloud Analytic Services

SAS Cloud Analytic Services: Overview

SAS Cloud Analytic Services (CAS) is a server that provides the cloud-based, run-time environment for data management and analytics with SAS. Suitable for both on-premises and Cloud deployments, CAS uses a combination of hardware and software where data management and analytics take place on either a single machine or as a distributed server across multiple machines.

Figure A.1 SAS Cloud Analytic Services



¹SAS Embedded Process is not required for serial connections.

SAS Cloud Analytic Services: How To (Scripts)

Operate

SAS Viya uses the operating system's default init system or systemd command to launch a script that can stop, start, and restart the Cloud Analytic Services (CAS) controller and its worker nodes. This script also checks the status of the CAS controller only. Residing in `/etc/init.d`, the script is named, `sas-viya-cascontroller-default`.

Note: You must be signed in to the machine where the [CAS controller](#) resides and you must have sudo privileges to run this script. Running `sas-viya-cascontroller-default` affects all worker nodes.

To operate the CAS controller and its worker nodes, run:

```
sas-viya-cascontroller-default stop | start | restart
```

To check the status of the CAS controller, run

```
sas-viya-cascontroller-default status
```

TIP Your site's Linux administrator might want to create a regular account (for example, `sas-service-admin`) and give that account the sudo permissions to manage the SAS services. Make sure that the services are defined as "start on reboot" so that the CAS server automatically starts when the machine is rebooted.

Note: There is a script with which you can manage and view the running state of all SAS Viya services. For more information, see ["All Servers and Services" on page 599](#).

For your convenience, here are a few examples:

- checking the status of the CAS controller using a direct call:

```
sudo /etc/init.d/sas-viya-cascontroller-default status
```

- stopping the CAS controller and its worker nodes using the Red Hat Linux version 6 init system command:

```
sudo service sas-viya-cascontroller-default stop
```

- starting the CAS controller and its worker nodes using the Red Hat Linux version 7 systemd command:

```
sudo systemctl start sas-viya-cascontroller-default
```

- restarting the CAS controller and its worker nodes using a direct call:

```
sudo /etc/init.d/sas-viya-cascontroller-default restart
```

Change the Process Owner Account

- 1 Log on to the CAS controller machine as the SAS install user (`sas`) or with sudo privileges.
- 2 Using a text editor, open `/opt/sas/viya/config/etc/sysconfig/cas/default/sas-cas-usermods`.
- 3 Locate the following lines:

```
SASUSER="user-account"
SASGROUP="primary-group"
```

Note: The default process owner account is `cas`. The default primary group for `cas` is `sas`.

Enter the new CAS process owner account. If needed, enter a new primary group for the CAS process owner, and save the file.

4 Open `/opt/sas/viya/home/SASFoundation/utilities/bin/launchconfig-viya-default`.

5 Locate the following line:

```
restrictServerLaunch=user-account
```

Enter the new CAS process owner account, and save the file.

CAS uses the new process owner account the next time it is run.

Add New Worker Nodes or a Backup Controller

The processes for adding new worker nodes or a backup controller (also known as a secondary controller) to your CAS server are very similar.

1 Make sure that you are licensed for the additional nodes or a backup controller that you are planning to add to your analytic cluster.

2 If you are adding a backup controller, make sure that the backup controller and the CAS controller (the primary controller) both use the same shared file system.

For more information, see [“Set Up a Shared File System for CAS Controllers \(Post-Deployment\)”](#) on page 611.

3 When adding to an existing SAS Viya deployment, SAS downloads and installs the latest software available from the software repositories. There is a risk in adding software to SAS Viya if you are not using a mirror.

For more information, see [“Creating and Using Mirror Repositories”](#) in *SAS Viya for Linux: Deployment Guide*.

4 Every machine on which you are installing CAS worker nodes or a backup controller must have the CAS user account (cas) and group (sas) set up.

For more information, see [“Set Up the cas Account”](#) in *SAS Viya for Linux: Deployment Guide*.

5 Sign in to the Ansible controller as the user account that deploys the software.

For more information, see [“Set Up the User Account that Deploys the Software”](#) in *SAS Viya for Linux: Deployment Guide*.

6 Choose which playbook to use. If you are adding a new:

- backup controller

Use the site playbook (site.yml).

- worker node

Decide which [playbook](#) to use.

Note: When adding worker nodes to a CAS controller running in SMP mode, use the site playbook.

CAUTION! Use the `deploy-casworker` playbook for adding worker nodes only. Do not change other CAS server configuration settings using the `deploy-casworker` playbook. Doing so can cause a mismatch between configuration in memory versus configuration on disk.

7 In the inventory file, define the machines on which you are adding the worker nodes or a backup controller.

TIP If you used the recommended location for uncompressing your playbook, the file is located at `/sas/install/sas_viya_playbook/inventory.ini`.

Here is an example of adding a backup controller:

```
controller-02 ansible_ssh_host=controller-02.example.com ansible_ssh_user=user1
ansible_ssh_private_key_file= ~/.ssh/id_rsa
```

Here is an example of adding a worker node:

```
worker-023 ansible_ssh_host=worker23.example.com ansible_ssh_user=user1
ansible_ssh_private_key_file= ~/.ssh/id_rsa
```

For more information, see [“Specify the Machines in the Deployment” in SAS Viya for Linux: Deployment Guide](#).

8 Also, in the inventory file, add the machines that you are adding to the appropriate group:

- backup controller

Use the `sas-casserver-secondary` group.

Note: If your inventory file does not contain a `sas-casserver-secondary` group, then create one, using the example that follows as a guide.

In this example, a backup controller is being added to the controller-02 machine:

```
[sas-casserver-secondary]
controller-02
```

- worker nodes

Use the `sas-casserver-worker` group.

Note: If your inventory file does not contain a `sas-casserver-worker` group, then create one, using the example that follows as a guide.

In this example, a worker node is being added to the worker-23 machine:

```
[sas-casserver-worker]
worker-019
worker-020
worker-021
worker-022
worker-023
```

TIP In the future, if you want to make a temporary worker a permanent worker, you must remove `persist=false` from the inventory file. Next, you have two options: either run `site.yml` (restarts CAS) or, run `deploy-casworker.yml` (does not restart CAS). If you choose to use `deploy-casworker`, the worker becomes permanent the next time the CAS server is restarted.

TIP In the future, if you want to use the temporary worker again, simply sign in to CAS Server Monitor to [add \(join\) on page 615](#) the new worker.

9 Run Ansible, using the playbook that you chose in [Step 6](#):

```
ansible-playbook -i inventory.ini playbook.yml
```

10 If you ran the `deploy-casworker` playbook and want to immediately start the worker nodes that you have added, sign in to CAS Server Monitor to [add \(join\) on page 615](#) the new worker nodes.

Set Up a Shared File System for CAS Controllers (Post-Deployment)

If you want to make your CAS controller fault tolerant, during installation you can choose to deploy a CAS backup controller (also referred to as a secondary controller). A requirement for operating CAS with a backup controller is that it and the CAS controller (the primary controller) must both use the same shared file system.

1 Shut down the CAS controller.

2 Copy all of the data from `/opt/sas/viya/config/data/cas` to a network share that both the CAS controller and its backup controller can access.

Note: Make sure that the copy preserves directory ownership and permissions.

Here is an example:

```
cp -Rp /opt/sas/viya/config/data/cas /share
```

3 Verify that all files and directories have been copied.

4 On the CAS controller, delete the old data directory.

Here is an example:

```
rm -r /opt/sas/viya/config/data/cas
```

5 On both the CAS controller and the backup controller machines, create a Linux symbolic link in `/opt/sas/viya/config/data/cas` that points to the new shared file system.

Here is an example:

```
ln -sf /share /opt/sas/viya/config/data/cas
```

6 Start the CAS controller.

Recover a Failed Controller

Note: While CAS is operating in the failed-over state, do not restart the primary (failed) controller service.

1 Shut down the CAS controller. (During a failover, the backup controller becomes the primary controller.)

Here is an example of stopping the CAS controller and its worker nodes using the Red Hat Linux version 6 init system command:

```
sudo service sas-viya-cascontroller-default stop
```

For more information, see [“Operate” on page 608](#).

2 Perform whatever steps necessary to either repair or replace the failed primary controller.

3 Restore the permstore directory from the backup controller to the primary controller.

For more information, see [“Restore the Most Recent Permstore in the Event of a Failover” on page 151](#).

4 Do the following:

a Restart all CAS worker nodes in your deployment. (Do not start your CAS controller.)

b Start the CAS controller.

For more information, see [“Operate” on page 608](#).

c At the Linux command prompt, enter the `sas-viya-controller-deployment-instance` command and verify that the CAS controller is indeed running.

Here is an example:

```
./sas-viya-cascontroller-default status
sas-viya-cascontroller-default is running
```

Host role in cluster:



```
Cluster Information:
  Tenant           = shared
  Instance         = default
  Primary Controller =
```

SAS Cloud Analytic Services: How To (SAS Environment Manager)

Introduction

These instructions explain how to view and modify SAS Cloud Analytic Services (CAS) settings using [SAS Environment Manager](#).


Navigation

In the applications menu () , under **Administration**, select **Manage Environment**. In the navigation bar, click .

The tasks described in this section are performed from the Data page and can be performed only by SAS Administrators.

Manage Whitelists and Blacklists

To change whitelist and blacklist settings, you must have [Superuser on page 494](#) privileges.

- 1 In **View**, select **Servers**.
- 2 Right-click the CAS server whose whitelist or blacklist you want to access, and select **Assume the Superuser role**.
- 3 Right-click the CAS server a second time, and select **Properties**.
- 4 Expand **Paths List** to view the active list.
- 5 To modify the active list, or to switch between a whitelist, blacklist, or no list, on the right side of the Server Properties window, click .


If you select the blacklist or whitelist, you can add or remove paths to the list.

Note: By default, the SAS Viya install and various configuration directories are on the blacklist.


- 6 To save any changes, click **Save**. Otherwise, click **Cancel**.
- 7 When you are finished, click **Close**.
- 8 Click **Relinquish** in the top right of the window to relinquish the Superuser role.

Adjust Caslib Management Privileges

To adjust caslib management privileges for a particular CAS server in SAS Environment Manager, you must have [Superuser on page 494](#) privileges.

- 1 In **View**, select **Servers**.
- 2 Right-click the CAS server whose caslib management privileges you want to adjust, and select **Assume the Superuser role**.
- 3 Right-click the CAS server a second time, and select **Properties**.
- 4 Expand **Caslib Management Privileges** to view identities and their caslib privileges.
- 5 To modify privileges, on the right side of the Server Properties window, click .
- 6 For the identities listed, choose to enable (or disable) the ability to add and delete session and global caslibs. Regardless of access controls, the Superuser can add and manage all caslibs.
Note: This display shows directly granted privileges. Indirectly granted privileges and denials of privileges are not reflected in this display.
- 7 To save any changes, click **Save**. Otherwise, click **Cancel**.
- 8 When you are finished, click **Close**.
- 9 Click **Relinquish** in the top right of the window to relinquish the Superuser role.




Terminate a CAS Server Session

- 1 In **View**, select **Servers**.
- 2 Right-click the CAS server whose session you want to terminate, and select **Assume the Superuser role**.
- 3 Right-click the CAS server a second time, and select **Sessions**.
- 4 In the Sessions window, select the check boxes for the sessions that you want to terminate, and click .
- 5 When you are finished, click **Close**.
- 6 Click **Relinquish** in the top right of the window to relinquish the Superuser role.

Stop a CAS Server

- 1 In **View**, select **Servers**.
- 2 Right-click the CAS server that you want to stop, and select **Assume the Superuser role**.
- 3 Right-click the CAS server a second time, and select **Stop server**.
- 4 In the alert box that is displayed, confirm your selection by clicking **Stop the Server**.
- 5 Click **Relinquish** in the top right of the window to relinquish the Superuser role.


Manage CAS Nodes

- 1 In **View**, select **Servers**.
- 2 Right-click the CAS server whose worker nodes you want to manage, and select **Assume the Superuser role**.
- 3 Right-click the CAS server a second time, and select **Properties**.
- 4 Expand **Nodes** to see the worker nodes for the specified CAS server.
- 5 On the right side of the Server Properties window, click .
- 6 Perform one of the following actions:
 - Add a worker node
Click  and enter the fully qualified domain name for the machine of the CAS worker node that you are adding.
 - Remove a worker node
Select the CAS worker node in the table, click , and, in the alert box, confirm the removal by clicking **Yes**.

Note: The process of dropping (removing) a node ensures that active and backup copies of table blocks are preserved. This requires that all sessions complete their actions and pause while the blocks are moved. Long-running actions are canceled, and occasionally, a session might need to be killed so that the operation can proceed. The data movement often takes minutes.
- 7 To save any changes, click **Save**. Otherwise, click **Cancel**.
- 8 When you are finished, click **Close**.
- 9 Click **Relinquish** in the top right of the window to relinquish the Superuser role.

SAS Cloud Analytic Services: How To (CAS Server Monitor)


View CAS Controller and System Information

- 1 [Sign in](#) to CAS Server Monitor with a valid user ID and password.
- 2 In CAS Server Monitor, beneath the **Cloud Analytic Services** banner, click .
- 3 On the System State page, make sure that **Controller** is selected.

View CAS Server Configuration

To use CAS Server Monitor to view the current list of [CAS Server options](#) and their values, follow these steps:


- 1 [Sign in](#) to CAS Server Monitor with a valid user ID and password.

- 2 In CAS Server Monitor, beneath the **Cloud Analytic Services** banner, click .
- 3 On the Configuration page, make sure that **CAS Configuration** is selected.

View CAS Start-up Options and Environment Variables

You can use CAS Server Monitor to view the option used when a CAS server was started and to see the current list of CAS environment variables and their values.

To view CAS start-up options and environment variable values, follow these steps:


- 1 [Sign in](#) to CAS Server Monitor with a valid user ID and password.
- 2 In CAS Server Monitor, beneath the **Cloud Analytic Services** banner, click .
- 3 On the System State page, select **Runtime Environment**.

Manage CAS Nodes

You can use CAS Server Monitor to view, add, and remove CAS nodes in your analytics cluster.

Note: In order to add CAS nodes, the requisite software must already have been deployed on the machines that you are adding. To add new machines, deploy SAS on them first, and then you can add them using the CAS Server Monitor.

To manage a node, follow these steps:

- 1 [Sign in](#) to CAS Server Monitor with a user ID that has CAS Administrator [privileges on page 493](#).
- 2 In CAS Server Monitor, beneath the **Cloud Analytic Services** banner, click .
- 3 On the System State page, select **Nodes**.
- 4 From the **Nodes** table, you can:
 - View information about all the nodes in your analytics cluster.
 - Add nodes to your analytics cluster:


Note: Before you can add *new* nodes to your cluster, you must have already added the CAS worker node software to the machine. For more information, see [“Add New Worker Nodes or a Backup Controller”](#).

- Click **Add Nodes**.
- On the Add Nodes dialog box, in **Hostname**, enter a simple host name, such as mygrid011, and click **OK**. The server monitor runs the CAS addNode action which starts (or restarts) the node and joins it to the cluster.

Separate multiple host names with a comma.

If your hosts are named in numeric order (for example, host002, host003, ...) you can enter a range of host names. Use the form, **host [start-number-end-number]** (for example, mygrid[002-030]).


- Drop worker nodes from your analytics cluster:


Next to the node that you want to drop, click  and select **Remove Nodes**. The server monitor runs the CAS removeNode action which stops the node and redistributes its data to other nodes in the cluster.

Note: The process of dropping (removing) a node ensures that active and backup copies of table blocks are preserved. This requires that all sessions complete their actions and pause while the blocks are

moved. Long-running actions are cancelled, and occasionally, a session might need to be killed so that the operation can proceed. The data movement often takes minutes.

TIP Removing a node is best suited for times when the system is mainly processing batch jobs, where a delay is not a concern.

- View information about processes running on a particular node:
Next to the node that you want to view process information about, click  and select **Show Processes**.
- Stop the server immediately by sending a kill signal to the server process:

Next to the node that you want to stop, click  and select **Terminate Server Instance**. The server monitor issues a command to kill the node process.

Note: Terminate a server instance only after having tried removing a node. Using terminate might not release resources (for example, mapped memory and memory involving database connections, and so on).

Remove CAS Worker Software


- 1 [Sign in](#) to CAS Server Monitor with a valid user ID and password that have administrator privileges.
- 2 Drop the CAS worker node whose software you want to remove.
For more information, see [“Manage CAS Nodes” on page 615](#).
- 3 On the CAS controller machine, remove the machine name of the worker node from `/opt/sas/viya/config/etc/cas/default/cas.hosts`.
- 4 Edit the inventory file on the Ansible controller machine to remove the deploy target definition at the top of the file and the deploy target name from `[sas-casserver-worker]`.
- 5 Sign in to the CAS worker machine with root or sudo privileges.
- 6 Run the following commands from an operating system prompt:

```
sudo yum erase "@SAS*" "@CAS*"
sudo /opt/sas/viya/home/utils/uninstall_viya.sh
sudo mv /opt/sas/viya/ /opt/sas/viya_$(date +"%m_%d_%Y")
```

View User Session Information



You can use CAS Server Monitor to view information about a user's session, such as connection port, length of connection time, and so on.

To view user session information, follow these steps:

- 1 [Sign in](#) to CAS Server Monitor with a valid user ID and password.
- 2 In CAS Server Monitor, beneath the **Cloud Analytic Services** banner, click .
- 3 On the System State page, select **User Sessions**.

Cancel CAS User Session



To cancel your CAS server session, follow these steps:

- 1 [Sign in](#) to CAS Server Monitor with a valid user ID and password.
- 2 In CAS Server Monitor, beneath the **Cloud Analytic Services** banner, click .
- 3 On the System State page, select **User Sessions**.
- 4 At the end of the row for the session that you want to cancel, click  and select **Cancel Session**.

Terminate CAS User Session


TIP Terminate a session only after having tried canceling a session. Using terminate might not release resources (for example, mapped memory and memory involving database connections, and so on).

To terminate your CAS server session, follow these steps:

- 1 [Sign in](#) to CAS Server Monitor with a valid user ID and password.
- 2 In CAS Server Monitor, beneath the **Cloud Analytic Services** banner, click .
- 3 On the System State page, select **User Sessions**.
- 4 At the end of the row for the session that you want to terminate, click  and select **Terminate Session**.

Adjust Caslib Management Privileges

To enable non-administrators to add global caslibs:

- 1 [Sign in](#) to CAS Server Monitor with a valid user ID and password that has administrator privileges.
- 2 In CAS Server Monitor, beneath the **Cloud Analytic Services** banner, click .
- 3 On the Configuration page, select **Access Controls**.
- 4 In the **Caslibs** list, select **Global Caslib Creation**.

TIP If the **Global Caslib Creation** caslib is not listed, you are not signed in as an administrator.

- 5 In the upper right, click **Edit**.
- 6 In the **Edit Access Controls** window, adjust values as needed.

Intent	Instructions
Enable all users to add global caslibs.	In the existing row for Authenticated Users , select the Grant radio button.
Enable a group to add global caslibs.	Click Add Row . Select Group , enter the group name, and select the Grant radio button.

Intent	Instructions
Enable an individual user to add global caslibs.	Click Add Row . Select User , enter the user name, and select the Grant radio button.

- 7 Click **OK** to save your changes.
- 8 Under **Access Controls**, review the results of your changes.
- 9 Verify that users who should be able to add global caslibs can do so.

Here are details:

- User and group names that you enter are not validated.
- Regardless of access controls, administrators can add and manage all caslibs.
- For the special caslibs (**Global Caslib Creation** and **Session Caslib Creation**), the only available value in the **Activity** column is **Manage Access**. The special caslibs are protected by role requirements, not by the `ManageAccess` permission. Granting or denying the `ManageAccess` permission on the special caslibs affects only the ability of non-administrators to manage other caslibs.
- If you want to restrict the ability to manage session caslibs, select **Session Caslib Creation** in the **Caslibs** list. Add direct denials as needed.

SAS Cloud Analytic Services: Concepts

CAS Controller

Controller is one of three roles that can be assigned to a host for SAS Cloud Analytic Services (CAS): controller, backup controller, and worker. For both server architectures—distributed and single-machine—one machine is assigned the controller role. When the server starts, the controller process is started. This process is sometimes referred to as the server controller. The controller accepts connections from clients.

CAS Backup Controller

A SAS Cloud Analytic Services (CAS) backup controller (sometimes referred to as *secondary controller*) provides fault tolerance for the CAS controller. A backup controller is used only in a distributed server architecture. Deploying a backup controller is optional. CAS supports one backup controller only.

Note: A requirement for operating CAS with a backup controller is that it and the CAS controller (the primary controller) must both use the same shared file system. For more information, see [“Set Up a Shared File System for CAS Controllers \(Post-Deployment\)”](#).

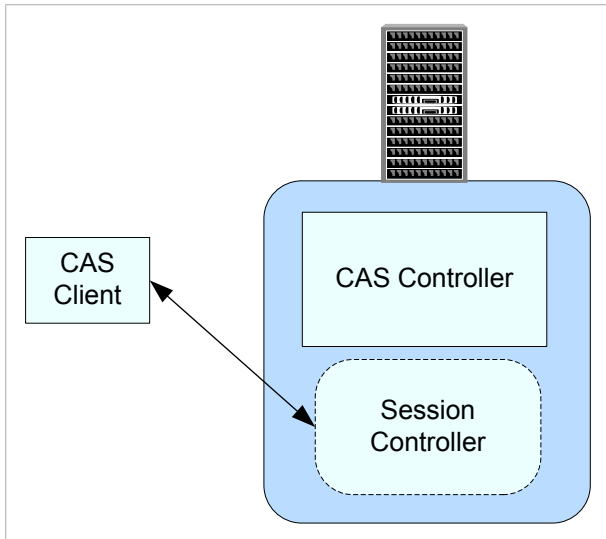
When CAS starts, the backup controller process is also started. In the event that the controller experiences a disruption (such as a loss of network connectivity, disk full scenarios, and so on) the backup controller enables the CAS server to continue running. When the backup controller takes control of client communication, the transfer is seamless. For more information, see [Architecture](#) in *SAS Cloud Analytic Services: Fundamentals*.

Single-machine CAS Server

The single-machine architecture uses symmetric multiprocessing (SMP). The functionality for a single-machine server is nearly identical to MPP, except that there is no cluster communication. In this architecture, the server acts as a controller. Before a client connects, the server listens on a port for connections.

After a client connects, a session is created and the session connects back to the client. (This is identical to the method that is performed by a CAS server that uses MPP.)

Figure A.2 Single-machine CAS Server



CAS Workers

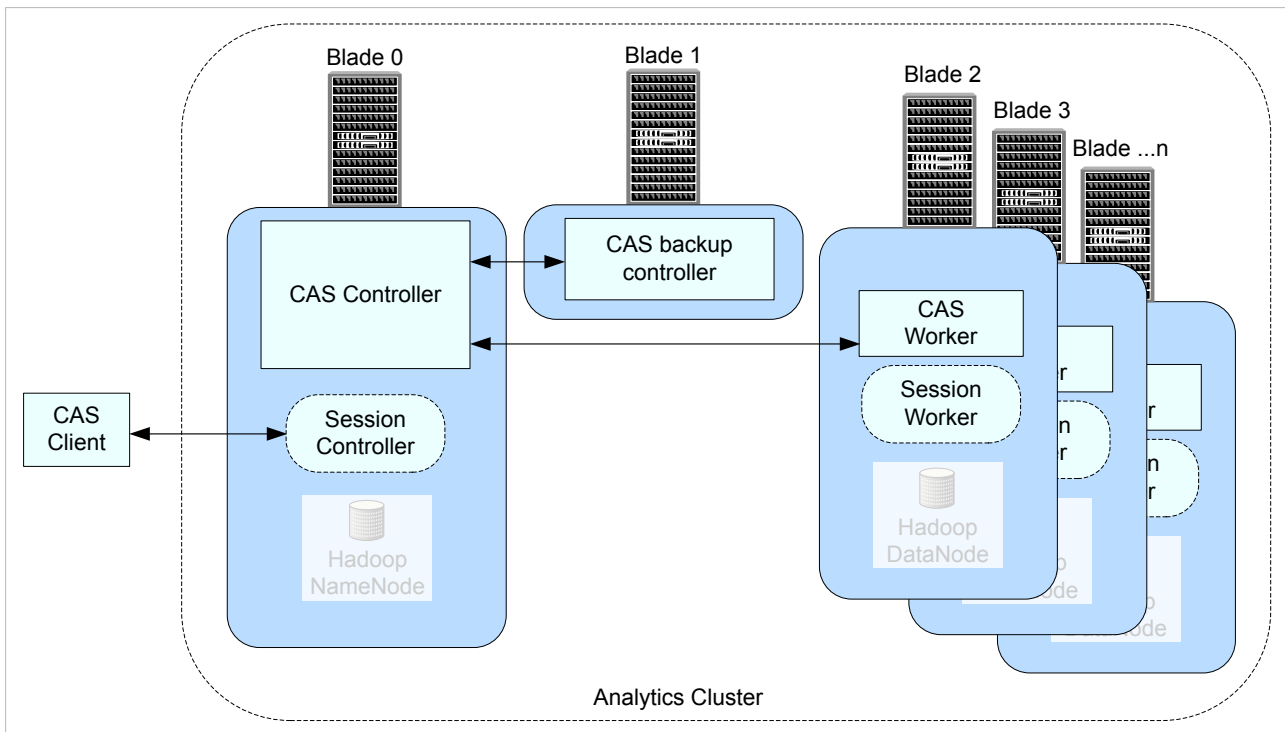
When a server is running in massively parallel processing (MPP) mode, in addition to a controller, the server also has multiple machines that are assigned the worker role.

The controller parses out work to each worker node. Each worker node sends the results of its computations back to the controller.

Distributed CAS Server

CAS can be co-located with Hadoop on a cluster of machines. This massively parallel processing (MPP) architecture is appropriate for analyzing large data sets. Analysis proceeds on tables that are already made available to the server (loaded) or on tables that are gathered or created by the server on demand.

Figure A.3 Distributed CAS Server



Session Processes

When a user connects to the server with a client, the server starts a session process for the user. Afterward, the client communicates with the session process.

A server running in symmetric multiprocessing mode (SMP mode) consists of a controller only, and the server starts a session controller process only. It is the session controller process that operates on rows of data.

In a distributed server (MPP mode), a session process is created on each machine in the cluster. These processes are sometimes referred to as the session controller and session worker processes.

Even though the sessions have their own operating system processes, the server processes must continue to run. When the server process terminates, the session processes also terminate.

Paths List

From a CAS server, all access to file system paths (host and HDFS directories) is through caslibs. To limit the paths that are available to non-administrators when they created or edit a caslib, use one of the following approaches:

- Create a blacklist of paths that should not be available.
- Create a whitelist of paths that should be available.

Here are key points:

- Paths must be absolute.
- All subdirectories of each specified path are affected.
- Paths list constraints do not affect access to existing caslibs.
- If you do not define a blacklist or whitelist, no paths list constraints are in effect.

- Paths list constraints do not apply to users who assume the Superuser role or the Data role.
- Only users who assume the Superuser role for a server can see and manage that server's paths list.

Note: Access to third-party databases is not affected by a server's blacklist or whitelist.

Caslib Management Privileges

Table A.47 Caslib Management Privileges

Task	Who Can Perform the Task*
Add global caslibs.	Superusers and Data administrators. Users who have global caslib management privileges.
Add session caslibs.	Superusers and Data administrators. Users who have session caslib management privileges.
Delete global caslibs.	Superusers and Data administrators. Users who have global caslib management privileges can delete any global caslib for which they have the ReadInfo and ManageAccess permissions.
Delete session caslibs.	Superusers and Data administrators. Users who have session caslib management privileges can delete any session caslib for which they have the ReadInfo and ManageAccess permissions.
Adjust caslib management privileges.	Superusers and Data administrators.

* Global caslib management privileges correspond to the ManageAccess permission on the _GLOBAL caslib. Session caslib management privileges correspond to the ManageAccess permission on the _SESSION caslib.

Note: Data administrators are displayed in CAS Server Monitor only.

Two Playbooks for Adding Worker Nodes

Adding worker nodes to your CAS server is accomplished using the third-party orchestration tool, [Ansible](#). When run on a CAS server that already has workers (MPP mode), or on a CAS server that does not have workers (SMP mode), Ansible performs these steps:

- configures SSH for the new machines and all existing machines that comprise the CAS server.
- installs software on all machines listed in the `sas-casserver-worker` group contained in the Ansible inventory file.

There are two playbooks to add nodes. Each playbook offers a different set of CAS usage characteristics:

- site playbook (site.yml)

Designed for when you want to:

- permanently add workers and have a maintenance window.
Added workers are automatically joined to the CAS server.
- change your CAS server configuration (for settings other than adding workers).

Note: Ansible restarts the CAS server when site.yml is used.

- `deploy-casworker` playbook (`deploy-casworker.yml`)

Designed for when you want to:

- temporarily add workers that persist only until the CAS server restarts or until the worker is dropped manually using the CAS Server Monitor **Remove Node** command.

Added workers must be joined manually to the CAS server using the CAS Server Monitor.

- permanently add workers, but do not have a maintenance window when the CAS server can be restarted.

On first use, added workers must be joined manually to the CAS server using the CAS Server Monitor. On subsequent invocations, the added workers are automatically joined to the CAS server.

Note: Ansible does not restart the CAS server when `deploy-casworker.yml` is used.

Note: When you add worker nodes to a CAS controller running in SMP mode, use the site playbook. The CAS server requires a restart when moving from SMP to MPP mode. The site playbook restarts the server. The `deploy-casworker` playbook does not restart the CAS server.

CAUTION! Use the `deploy-casworker` playbook for adding worker nodes only. Do not change other CAS server configuration settings using the `deploy-casworker` playbook. Doing so can cause a mismatch between configuration in memory versus configuration on disk.

Understanding Configuration Files and Start-up Files

Several SAS applications require application-specific configuration and start-up before SAS Cloud Analytic Services begins processing client requests. It is worthwhile to understand how the files are processed so that you can use the same technique to customize your server deployment.

Configuration Home Directory

The installation and deployment software creates a configuration home directory for each server instance.

Here is an example:

```
/opt/sas/viya/config/etc/cas/default
```

The final directory in the path, `default`, is the deployment instance for the server.

Standard Configuration Files

The configuration home directory includes several files with standard names. The server automatically processes these files when the standard names are used.

The following table describes the purpose and use for each of the standard files.

Table A.48 Server Configuration Files

Standard Filename	Description
<code>casconfig.lua</code>	<p>This file contains most of the configuration settings for the server instance, such as the network port that the server listens on.</p> <p>During deployment, RPM owns the <code>casconfig.lua</code> file and during updates can override any user configuration. For more information about the settings in the file, see “Configuration File Options” on page 627.</p>

Standard Filename	Description
casconfig_deployment.lua	This file contains CAS configuration settings that are created during deployment by Ansible from vars.yml. During updates, user configuration settings are overwritten.
casconfig_usermods.lua	This file contains modifications made by the SAS administrator. Using casconfig_usermods.lua ensures that your modifications are not overwritten when you upgrade CAS.
conf.d/	This directory can contain one or more configuration files that are similar to the casconfig.lua file. The files are processed in alphabetical order. The files in this directory are processed before the casconfig.lua file.
node.lua	This file contains host-specific configuration. One possible use is for security setup that relies on the host name.
node_usermods.lua	This file contains modifications that are made by the SAS administrator. Using casnode_usermods.lua ensures that your modifications are not overwritten when you upgrade CAS.
logconfig.xml	This file contains the SAS logging facility instructions that control server logging.
perms.xml	This file contains the initial permission settings. This file is not used after the first time the server is started and the permstore is populated.
cas.hosts	This file contains the initial set of host names and roles (controller or worker) for the server.
cas.settings*	This file contains CAS and system environment variables that are created during deployment by Ansible from vars.yml. During updates, user configuration settings can be overwritten.
cas_usermods.settings	This file contains modifications that are made by the SAS administrator. Using cas_usermods.settings ensures that your modifications are not overwritten when you upgrade CAS.

* There is a global version of cas.settings that resides in /opt/sas/viya/home/SASFoundation. CAS processes the global version of cas.settings **before** processing the configuration-specific version of cas.settings.

When the server starts, the configuration files that are described in the preceding table are processed. After the configuration is complete, the server runs start-up scripts.

The following table describes the standard names for the start-up files in the configuration home directory. The start-up scripts run before the server accepts any client connections. This is also referred to as *session-zero processing*.

Table A.49 Server Start-up Files (Session 0 Processing)

Standard Filename	Description
casstartup.lua	This file contains the actions to run as the CAS server starts, such as some addFmtLib actions and a setServOpt action, that are created during deployment by Ansible from vars.yml.
casstartup_usermods.lua	<p>This file contains modifications that are made by the SAS administrator to casstartup.lua, such as adding global-scope caslibs and loading global-scope tables. Using casstartup_usermods.lua ensures that your modifications are not overwritten when you upgrade CAS.</p> <p>Use Lua syntax such as the following. Do not forget to use global scope.</p> <pre>s:table_addCaslib{caslib="worldbank", dataSource={srcType="path"}, path="/rdstore/data/smp/world_bank", session=false}</pre>
start.d/	This directory contains one or more start-up scripts that are similar to the startup.lua file. The files are processed in alphabetical order. The files in this directory are processed before the startup.lua file

Fault Tolerance

If the [primary controller](#) fails, the site operates without fault tolerance for the controller until a planned outage. During the planned outage, the site should [recover the failed controller](#) and return to a redundant state.

After the outage, the primary controller accepts connections from clients and the [backup \(or secondary\) controller](#) resumes its role of providing fault tolerance.

If the backup controller fails while the primary controller continues to operate, the site can continue to operate without fault tolerance.

Note: For information about fault tolerance in other parts of SAS Viya, see [“Fault Tolerance in SAS Viya” on page 602](#).

Operating system tuning such as ulimits should be identical on both controller hosts.

For sites that co-locate the server with Hadoop:

- You can set the primary controller and backup controller to use the same hosts as the active NameNodes and the standby NameNodes. This is not a requirement.
- The HADOOP_NAMENODE environment variable can include the two host names. You can specify the active and the standby NameNodes hosts, separated by a colon.

Access controls and caslib information are stored in a directory that is known as a permstore. While the primary controller and the backup controller are running, the permstore for the backup controller is kept in sync with the primary controller. After a failover, the permstore for the backup controller becomes the most current. Part of the task of restoring the primary controller is to copy the files from the permstore on the backup controller to the permstore on the primary controller. For more information, see [SAS Viya Administration: Backup and Restore](#).

See Also

- [“Set Up a Shared File System for CAS Controllers \(Post-Deployment\)”](#)

- “Recover a Failed Controller”

Using CAS Server Monitor

What Is CAS Server Monitor?

CAS Server Monitor is a web application that you use to monitor your CAS Server and to perform some [administration tasks](#).

Monitor Cheat Sheet

In addition to providing information, CAS Server Monitor supports the following tasks:

Task	Navigation	Required Role
Stop the server	System State ⇒ Controller	CAS (Superuser)
Add or remove nodes**	System State ⇒ Grid Nodes	CAS (Superuser)
Terminate your sessions	System State ⇒ User Sessions	(none)
Terminate other sessions	System State ⇒ User Sessions	CAS (Superuser) or Data
Designate administrators	Configuration ⇒ Administrators	CAS (Superuser)
Set caslib permissions on page 87	Configuration ⇒ Access Controls	CAS (Superuser) or Data*

* In addition, any user who has the ManageAccess permission for a global caslib can set permissions on that caslib.

** On Cloud Foundry, do not attempt to add nodes, remove nodes, or terminate a server instance from the CAS Server Monitor (or with the addNode and removeNode CAS actions). Instead, use the appropriate BOSH command.

Access the Monitor

- 1 You can access the monitor without first starting a CAS server session. However, if there are no sessions, the session list in the monitor is empty.

If you already have a CAS server session, skip to [Step 2](#). Otherwise, to start a session, perform the following steps:

- a Open a web browser and sign in to SAS Studio with administrator privileges:

```
https://reverse-proxy-server/SASStudio
```

- b In the **Code** tab, start a CAS server session by entering the following:

```
cas my_session;
```

- c Click .

You should see output similar to the following:

```

56      cas my_session;
NOTE: The session MY_SESSION connected successfully to Cloud Analytic Services 10.120.9.159 using port 5570.
The UUID is
      5120eb8f-ca06-8c44-9f93-2dd2e557b1cb. The user is myacct and the active caslib is CASUSER(myacct).
NOTE: The SAS option SESSREF was updated with the value MY_SESSION.
NOTE: The SAS macro _SESSREF_ was updated with the value MY_SESSION.
NOTE: The session is using 0 workers.

```

2 Open a web browser and enter the following URL in the address field:

`https://http-reverse-proxy-machine-name/cas-tenant-name-deployment-instance-name-http`

Here is an example:

`https://myproxy.example.com/cas-shared-default-http`

TIP To designate administrators, select **Configuration** ⇒ **Administrators**, click **Add**, and select **CAS** (Superuser).

Usage notes:

- When the CAS server terminates, the CAS Server Monitor also terminates.
- The session view remains displayed even if the session is terminated. A session view is displayed until you close it.
- Sessions are removed from the list if the session is terminated. You can click the refresh button to get the current list of sessions.
- You can also access CAS Server Monitor from SAS Studio.
- CAS Server Monitor does not use a session time-out. You must click ▼ in the top right corner of the window and select **Sign Out** to exit.

Set Monitor Preferences

To control the view that you see by default, click ▼ in the top right corner of the window and select **Settings**.

SAS Cloud Analytic Services: Troubleshooting

Failed to open temporary file for upload (80BFE801): /tmp/cas7cache1/_f_43d6c87c_7f5d854996e8.sas7bdat

Explanation:

Insufficient disk space in CAS_DISK_CACHE on the CAS controller.

Resolution:

Add more disk space.

SAS Cloud Analytic Services: Reference

Configuration File Options

How Do I Use CAS Configuration File Options?

You set SAS Cloud Analytic Services options in the CAS controller's configuration file, `casconfig_usermods.lua`. During CAS server start-up, the controller shares the configuration as each worker and the backup controller (if present) connects. By default, `casconfig_usermods.lua` is located in `/opt/sas/viya/config/etc/cas/default`.

If you want to isolate a configuration option change to a particular CAS node, then make your change in `node_usermods.lua` residing on the particular node machine.

TIP For sites that use Ansible, it is recommended that you make your CAS server configuration changes to `vars.yml` and rerun Ansible to apply these changes. For more information, see [“Modify the vars.yml File” in SAS Viya for Linux: Deployment Guide](#).

There are additional CAS configuration files and directories. For more information, see [“Understanding Configuration Files and Start-up Files”](#).

For the order of precedence for server configuration options, see [How the Session Option Values Are Determined](#).

When a session starts, session options specified in the `casconfig` files are set for the session. For the order of precedence for session options specified in the `casconfig` files, see [Table 34.48 on page 622](#).

TIP Remember that you can also set operating system environment variables in `casconfig_usermods.lua`. For example, `env.HADOOP_HOME='/hadoop/hadoop-someversion'`
`env.HADOOP_NAMENODE='name_node.example.com'`.

You can override CAS configuration file settings for a session by changing the equivalent session option. For more information, see [Setting Session Options](#).

Configuration File Options Reference

Note:

Other security-related configuration file options can be found in [“Configuration File Options for Data Transfer” on page 443](#).

cas.APPTAG='tag-string'

specifies an arbitrary string to prefix to log messages.

Using `apptag` helps determine which log messages are associated with an application.

Valid in CAS statement SESSOPTS option

[casconfig_usermods.lua file](#)

Category Session

Default No *tag-string*

Note The CAS server uses `apptag` when writing to its log.

See [cas.LOGCFGLOC](#)

Example `apptag='my_app'`

cas.CMPOPT='optimization-value <optimization-value <...>>' | 'all' | 'none'

specifies the type of code generation optimizations to use in the SAS language compiler.

■ *optimization-value*

specifies the type of optimization that the SAS compiler is to use. Specify one or more of the following as a space-delimited list enclosed in quotation marks:

□ 'dumptkgcode' | 'nodumptkgcode'

specifies whether all CAS server nodes create an output file with the generated program. The CAS log lists where CAS writes the output file.

□ 'extramath' | 'noextramath'

specifies whether the compiler is to retain or remove the extra mathematical operations that do not affect the outcome of a statement.

□ 'funcdifferencing' | 'nofuncdifferencing'

specify **funcdifferencing** to calculate numeric-differencing derivatives for user-defined functions. Specify **nofuncdifferencing** to calculate analytic derivatives for user-defined functions.

□ 'guardcheck' | 'noguardcheck'

specifies whether the compiler checks for array boundary problems.

Note: **noguardcheck** is set when `cmpopt` is set to 'all' or 'none'.

□ 'misscheck' | 'nomisscheck'

specifies whether to check for missing values in the data.

□ 'precise' | 'noprecise'

specify **precise** to handle exceptions at the operation boundary. Specify **noprecise** to handle exceptions at the statement boundary.

■ 'all'

specifies that the compiler is to optimize the machine language code by using the **noextramath**, **nomisscheck**, **noprecise**, **noguardcheck**, and **nofuncdifferencing** optimization values.

Note: 'all' cannot be specified with other values.

■ 'none'

specifies that the compiler is not set to optimize the machine language code by using the **extramath**, **misscheck**, **precise**, **noguardcheck**, and **funcdifferencing** optimization values.

Note: 'none' cannot be specified with other values.

Valid in CAS statement `SESSOPTS` option

[casconfig_usermods.lua](#) file

Category Action

Default **noextramath**, **nofuncdifferencing**, **noguardcheck**, **nomisscheck**, and **noprecise**

Note If the data contains a significant amount of missing data, specify **misscheck** to optimize the compilation. Otherwise, specify **nomisscheck**.

Example In this example, the SAS compiler is set to retain the extra mathematical operations, check for missing values, and handle exceptions at an operation boundary:

```
cas.cmpopt='extramath misscheck precise'
```

cas.COLLATE='mva' | 'uca'

specifies the collating sequence for sorting.

`mva` specifies SAS client collating. `uca` specifies a locale-appropriate collating sequence.

Valid in CAS statement SESSOPTS option

[casconfig_usermods.lua file](#)

Category Sort

Default 'uca'

Example `cas.collate='mva'`

cas.COLOCATION='none' | 'hdfs'

specifies whether to create a personal caslib (`hdfs`) at CAS server start-up.

A server started in MPP mode defaults to `hdfs` because it assumes it is co-located with Hadoop. Specify `none` for the server running in MPP mode not to create a personal caslib at start-up.

Valid in [casconfig_usermods.lua file](#)

Category Caslib

Default `cas.colocation='hdfs'`

Requirement Used with `cas.mode='mpp'` and `cas.hdfsuserloc`.

Example In this example, the CAS server is running in MPP mode and is not co-located with Hadoop. At start-up, the CAS server does not create a personal caslib for the user ID under which the server is run.

```
cas.colocation='none'
```

cas.DATASTEPFMterr=true | false

corresponds to the FMterr in SAS. Specifies how the DATA step reacts when a format is not available. When `true`, the DATA step writes an error and stops. When `false`, the DATA step uses `$w.` or `BEST12.` instead of the unavailable format. (The unavailable format is still associated with variables in the output table.)

Valid in CAS statement SESSOPTS option

[casconfig_usermods.lua file](#)

Category DATA Step

Alias FMterr

Default True

Note The values `true` and `false` are case sensitive.

See [FMterr System Option](#)

Example In this example, the DATA step uses \$w. or BEST12. instead of the unavailable format.
`cas.datastepfmtterr=false`

cas.DATASMSGSUMLEVEL='all' | 'none' | 'put'

specifies the DATA step message summary level. When the DATA step runs on multiple threads, the same message can be generated on each thread. This option controls the summary level of duplicate messages.

- 'all'

The first occurrence of all message and put statements are sent to the client when they occur. Duplicate occurrences of all message and put statements are summarized and sent to the client when the DATA step exits. This is the default.

- 'none'

All message and put statements from every thread are written to the client log. No summarization occurs.

- 'put'

The first occurrence of all message and put statements are sent to the client. Duplicate occurrences of messages are summarized and sent to the client when the DATA step exits. Put statements are not summarized; rather, they are sent to the client when they occur.

Valid in CAS statement SESSOPTS option

[casconfig_usermods.lua file](#)

Category DATA Step

Default All

Example In this example, all message and put statements from every thread are written to the client log. No summarization occurs.
`cas.datastepmsgsumlevel='none'`

cas.DATASPREPLACETABLE=true | false

specifies whether a DATA step can replace an existing table.

Valid in CAS statement SESSOPTS option

[casconfig_usermods.lua file](#)

Category DATA Step

Default true

Note The values `true` and `false` are case sensitive.

Example `cas.datastepreplacetable=true`

cas.DCHOSTNAMERESOLUTION= '[ep | ep_fqdn | cas | cas_ipv6]'

specifies how CAS sends the CAS node machine name to SAS Embedded Process.

- 'ep'

Send CAS node machine names to SAS Embedded Process exactly how they are defined in `cas.hosts`. SAS Embedded Process resolves the names using either IPv4 or IPv6.

- 'ep_fqdn'

Send CAS node machine names to SAS Embedded Process as fully qualified domain names. SAS Embedded Process resolves the names.

- 'cas'

(Default) send CAS node machine names to SAS Embedded Process as IPv4 addresses.

- 'cas_ipv6'

Send CAS node machine names to SAS Embedded Process as either IPv4 or IPv6 addresses.

Valid in [casconfig_usermods.lua file](#)

Category Network

Default 'cas'

Example `cas.dchostnameresolution='ep'`

cas.DQLOCALE='locale-code'

specifies the default locale to use for data quality (DQ) operations, using the five-letter SAS Quality Knowledge Base (QKB) ISO locale code.

For more information, see [QKB Locale ISO Codes](#).

Valid in CAS statement SESSOPTS option

[casconfig_usermods.lua file](#)

Category Data Quality

Example In this example, the default locale for DQ operations is French Canadian:
`cas.dqlocale='fr_CA'`

cas.DQSETUPLOC='QKB-name'

specifies the name of the default SAS Quality Knowledge Base (QKB) to use for data quality (DQ) operations.

QKB-name is the absolute path to a SAS Quality Knowledge Base.

Valid in CAS statement SESSOPTS option

[casconfig_usermods.lua file](#)

Category Data Quality

Example `dqSetupLoc='/opt/sashome/SASQualityKnowledgeBases/en/my_qkb'`

cas.ELASTIC=true | false

indicates that new machines are allowed to join the analytics cluster.

Valid in [casconfig_usermods.lua file](#)

Category Administration

Default false

Requirement Used with `cas.gcport`.

Supports CAS servers running MPP.

Note	The values <code>true</code> and <code>false</code> are case sensitive.
See	cas.GCPORT
Example	In this example, the CAS controller allows new worker nodes to join the analytic cluster: <code>cas.elastic=true</code>

cas.ELASTICSSL=true | false

When `elasticssl=true`, new machines are allowed to join the cluster when the CAS controller can authenticate their identity.

Authentication data is contained in a key file that was used to start the CAS server. Any machine that can access this key can join the CAS controller as a worker node.

Valid in	casconfig_usermods.lua file
Category	Security
Default	<code>false</code>
Requirement	Used with <code>cas.gcport</code> and <code>cas.keyfile</code> .
Supports	CAS servers running MPP.
Note	The values <code>true</code> and <code>false</code> are case sensitive.
See	cas.GCPORT and cas.KEYFILE
Example	<code>cas.elasticssl=true</code>

cas.EVENTDS='event-data-set'

specifies one or more event objects that define custom date events.

event-data-set specifies the name of a data set that contains event definitions. You can use a one-level name or a two-level name, such as `libref.dataset`. When specifying multiple names, separate each name with a space.

Enclose *event-data-set* in single quotation marks.

Valid in	CAS statement SESSOPTS option casconfig_usermods.lua file
Category	Input Control
Example	<code>cas.eventds='mydataset'</code>

cas.FMTSEARCH='logicalformatlibname logicalformatlibname2 ... '

specifies the format search list to be automatically set at session start-up.

Server configuration files can be used to add and promote common format libraries when a server starts. Promotion makes the format libraries available to all sessions.

During session start up the `cas.FMTSEARCH` value is used to generate and execute the `setFmtsearch` action. The `setFmtsearch` action specifies the order in which format libraries are searched. Table variables can have a user-defined format association. During table processing, when a user-defined format needs to be applied, CAS searches the list of format libraries established by the `setFmtsearch` action.

Category	Formats
-----------------	---------

Default	blank
Interaction	Specifying the setServOpt action for format search in startup.lua takes precedence over cas.FMTSEARCH used in casconfig.lua.
Notes	The format library names are logical names known to a session. The names are case insensitive. When <code>userformats1-5</code> are not populated, you can ignore errors similar to the following: Failed to access file (80BFE801): /opt/sas/viya/config/data/cas/default/formats/userformats2.sashdat
Example	<code>cas.fmtsearch='userformats1 userformats2'</code>

cas.GCPORT=port

specifies the network port that is used on a distributed server for communication between the controller and its worker nodes.

The commonly configured port is 5580.

Valid in	casconfig_usermods.lua file
Category	Network
Default	0 (random port in the range 32678–61000)
Supports	CAS servers running MPP.
See	cas.HTTPPORT and cas.PORT
Example	<code>cas.gcport=5580</code>

cas.HDFSUSERLOC='/hdfs-path/%USER'

for CAS servers running in MPP mode, specifies that the server create a personal caslib for each user at session start-up time in the specified HDFS path.

'*hdfs-path*/%USER' refers to a directory named for the user's user ID under the specified HDFS path.

Enclose *hdfs-path* in single quotation marks.

Valid in	casconfig_usermods.lua file
Category	Data
Requirement	cas.MODE='mpp' on page 639 and cas.COLOCATION='hdfs' on page 629
Example	In this example, the user's caslib directory is a subdirectory named for the user ID under / user : <code>cas.hdfsuserloc='/user/%USER'</code>

cas.HTTPPORT=port | port-range

The port (or range of ports) that SAS Cloud Analytic Services listens to for HTTP communication.

The commonly configured port is 8777.

Valid in	casconfig_usermods.lua file
Category	Network
Default	0 (random port)

Note	If the first port in the range is already taken, CAS tries the next port until it finds a port that is free.
See	cas.HTTPPORTMAX , cas.GCPORT , and cas.PORT
Examples	<code>cas.httpport=8777</code>
	<code>cas.httpport=8777-9000</code>

cas.HTTPPORTMAX=*maximum-port-range*

specifies the maximum port range that SAS Cloud Analytic Services listens to for HTTP communication.

Valid in [casconfig_usermods.lua file](#)

Category Network

Default 0

Range 0–65535

See [cas.HTTPPORT](#)

Example `cas.httpport=8777-9000`

cas.INITIALBACKUPS= *-1 | 0 | positive-number*

specifies whether SAS Cloud Analytic Services (CAS) waits for backup controllers to connect to the CAS analytics cluster before CAS begins to accept client connections.

Valid values are:

- -1

Use the value specified in `cas.hosts` for the number of backup controllers to connect to the analytics cluster before CAS begins accepting connections from clients.

- 0 (zero)

Do not wait for any backup controllers to connect to the analytics cluster before CAS begins accepting connections from clients.

- 1

Wait for the backup controller to connect to the analytics cluster before CAS begins accepting connections from clients.

Valid in [casconfig_usermods.lua file](#)

Category Server

Default -1

Range -1–1

Example `cas.INITIALBACKUPS=-1`

cas.INITIALWORKERS=*'n'*

specifies the number of CAS worker nodes that must join the analytic cluster before CAS begins processing user connections.

`cas.initialworkers` enables administrators to establish an expected cluster size for configurations, where it is typical for all or most worker nodes to join elastically.

Valid in [casconfig_usermods.lua file](#)

Category	Server
Default	-1
Range	-1 to 32767
Requirement	cas.elastic must be set to true.
Notes	A value of zero indicates that the controller does not wait for any worker nodes to join the cluster before it begins to establish user connections. A value of -1 indicates that the controller waits for the number of workers that is specified in the machine list file.
See	cas.ELASTIC and cas.MACHINELIST
Example	In this example, the CAS controller waits for 16 workers to join the analytic cluster before it begins processing user connections. <code>cas.initialworkers='16'</code>

cas.INTERVALDS='interval-1=libref.dataset-name-1 <interval-2=libref.dataset-name-2 ...>'

specifies one or more interval-name=value pairs, where the value is the name of a data set that contains user-defined intervals.

Valid in CAS statement SESSOPTS option

[casconfig_usermods.lua file](#)

Category Input Control

See INTERVALDS= System Option

Example `cas.intervalsds='subsid1=subsid.storeHours'`

cas.JREOPTIONS='(JRE-option <JRE-option> <...>)'

specifies the Java Virtual Machine (JVM) options that SAS Cloud Analytic Services uses at start-up. Separate JRE options with a whitespace character. Enclose any paths in quotation marks.

For the list of the valid Java options, and what they do, see <http://docs.oracle.com/javase/6/docs/technotes/tools/windows/java.html>

Valid in [casconfig_usermods.lua file](#)

Category Java

Default (null)

Example In the following example, the initial and maximum sizes of the memory allocation pool are set to 256 and 1024MB, respectively. Also, the log4j configuration file path and Java classpath are set:

```
cas.jreoptions = '(-Xms256m -Xmx1024m
-Dlog4j.configuration=' .. ' -Djava.class.path=' .. env.CAS_HOME .. '/lib/base/base-tkjni.jar)'
```

cas.KEYFILE='pathname'

identifies to the CAS controller the path and filename to the X.509 digital certificate file that is used to start the server. The certificate must be signed by a CA that is trusted by the CAS server.

Enclose *pathname* in single quotation marks.

Valid in [casconfig_usermods.lua file](#)

Category	Security
Requirement	Used with <code>cas.elasticssl</code> and <code>cas.mode='mpp'</code> .
Supports	CAS servers running MPP.
See	cas.GCPORT and cas.ELASTIC
Example	<code>cas.keyfile='/opt/TKGrid/certs/controller.pem'</code>

cas.LIFETIME=*minutes*

indicates the duration, in minutes, that a server remains running.

Valid in	casconfig_usermods.lua file
Category	Administration
Default	0
Example	In the following example, the server shuts itself down in 120 minutes: <code>cas.lifetime=120</code>

cas.LOCALE=*'POSIX-locale-string'*

specifies the locale to use for sorting and formatting. For a list of valid POSIX locale strings, see [SAS National Language Support \(NLS\): Reference Guide](#)

Valid in	CAS statement SESSOPTS option casconfig_usermods.lua file
Category	Localization
Default	'en_US'
Example	<code>cas.locale='fr_FR'</code>

cas.LOGCFGLOC=*'pathname'*

specifies the path to the SAS logging facility logging configuration file.

Enclose *pathname* in single quotation marks.

Valid in	casconfig_usermods.lua file
Category	Log
See	cas.APPTAG on page 627
Example	<code>cas.logcfgloc='/opt/sas/cas1/etc/logconfig.xml'</code>

cas.LOGFLUSHTIME=*-1 | 0 | number*

specifies the log flush time, in milliseconds.

- -1
flushes logs after each action completes.
- 0
flushes logs as they are produced.
- *number*

flushes logs in *number* milliseconds.

Valid in	CAS statement SESSOPTS option casconfig_usermods.lua file
Category	Log
Default	100
Range	-1–86400
Example	In the following example, the CAS server writes buffered lines to the log every 500 milliseconds: <code>cas.logflushtime=500</code>

cas.MACHINELIST='path/machine-list-file'

identifies the path and filename on the controller machine that contains the list of machines in the CAS analytics cluster.

Enclose *path* in single quotation marks.

machine-list-file contains all of the machines in the analytics cluster in the form:

`<fully-qualified-domain-name controller | worker>`

Place each machine on a separate line. For example:

```
my_machine01.example.com controller
my_machine02.example.com worker
my_machine03.example.com worker
my_machine04.example.com worker
my_machine05.example.com worker
```

Valid in	casconfig_usermods.lua file
Category	Administration
Requirement	Used with <code>cas.mode= 'mpp'</code> .
Interaction	Do not specify <code>cas.mode= 'smp'</code> when a valid machine list file is used.
See	cas.MODE
Example	<code>cas.machinelist= '/etc/my_machine_list'</code>

cas.MAXSESSIONS='n'

specifies the maximum number of concurrent sessions. Users who can assume an administrative role are not subject to the limit.

Valid in	casconfig_usermods.lua file
Category	Server
Default	5000
Range	0–100000
Notes	Specifying zero (0) indicates that there is no session limit.

This option cannot be changed after the system initializes.

Example In this example, the maximum number of concurrent CAS sessions is 1,000:
`cas.maxsessions='1000'`

cas.MAXTABLEMEM=*number* | '[*number* *k* | *m* | *g* | *t*]'

specifies the maximum amount of physical memory to allocate for a table.

TIP The intent of `cas.MAXTABLEMEM` is to manage the efficiency of accessing CAS tables on disk, not to control the amount of data that CAS keeps resident in RAM. When you need to manage CAS memory, consider modifying `cas.MEMORYSIZE`.

- *number*
specifies the maximum amount of physical memory, in bytes, to allocate for a table.
- '[*number* *k* | *m* | *g* | *t*]'
specifies the maximum amount of physical memory to allocate for a table in a unit other than bytes: **k** (kilobytes), **m** (megabytes), **g** (gigabytes), and **t** (terabytes).

Valid in CAS statement SESSOPTS option

[casconfig_usermods.lua file](#)

Category Caslib

Default 16M

Note After this threshold is reached, the server uses temporary files and operating system facilities for memory management.

See [cas.MEMORYSIZE](#)

Example In this example, the CAS server can allocate up to 32MB of physical memory for a table:
`cas.maxtablemem='32m'`

cas.MEMORYSIZE=*number* | '[*number* *k* | *m* | *g* | *t*]'

specifies the maximum amount of physical memory to allocate for the CAS cgroup. This limit also applies to the YARN request, when `cas.USEYARN` is specified.

- *number*
specifies the maximum amount of physical memory, in bytes, to allocate for the CAS cgroup and the YARN request.
- '[*number* *k* | *m* | *g* | *t*]'
specifies the maximum amount of physical memory to allocate for the CAS CGroup and the YARN request in a unit other than bytes: **k** (kilobytes), **m** (megabytes), **g** (gigabytes), and **t** (terabytes).

Valid in [casconfig_usermods.lua file](#)

Category Administration

Default 0

See [cas.USEYARN on page 645](#)

“How to Limit Memory Use” in *SAS Cloud Analytic Services: Fundamentals*

Example In the following example, the maximum amount of physical memory allocated for the CAS cgroup and the YARN request is 256GB:

```
cas.memorysize='256g'
```

cas.MESSAGELEVEL='all' | 'default' | 'error' | 'none' | 'note' | 'warning'
specifies the log message level.

Valid in CAS statement SESSOPTS option

[casconfig_usermods.lua file](#)

Category Log

Default 'all'

Example `cas.messagelevel='default'`

cas.METRICS=true | false

causes CAS server metrics information to be displayed (`true`) or not displayed (`false`) in the SAS log.

When `cas.metrics=true`, you see information similar to the following displayed in the SAS log:

```
NOTE: Action 'nobs' used (Total process time):
NOTE:      real time          2.100185 seconds
NOTE:      cpu time           0.010999 seconds (0.52%)
NOTE:      total nodes        6 (192 cores)
NOTE:      total memory       1.11T
NOTE:      memory             7.00K (0.00%)
```

The analytic server processed the request in 2.100185 seconds.

Valid in CAS statement SESSOPTS option

[casconfig_usermods.lua file](#)

Category Log

Default false

Note The values `true` and `false` are case sensitive.

See CASLIB Statement

Example In the following example, CAS server metrics information is not displayed in the SAS log:
`cas.metrics=false`

cas.MODE='smp' | 'mpp'

forces a server to be started in symmetric multiprocessing mode (`smp`) or in massively parallel processing mode (`mpp`).

Valid in [casconfig_usermods.lua file](#)

Category Administration

Interaction `cas.MODE` is implicitly set to `'mpp'` when [cas.ELASTIC](#) is set, or [cas.ELASTICSSL](#) is set, or [cas.MACHINELIST](#) is set and contains at least one CAS worker node or backup controller. Otherwise, `cas.MODE` is implicitly set to `'smp'`.

Note The server returns an error when `cas.mode='smp'` is specified for a server with a valid machine list.

Example In the following example, the CAS server is forced to start in massively parallel processing mode (MPP).

```
cas.mode='mpp'
```

cas.NODE='filename'

specifies the configuration file that is run on all CAS worker nodes.

Valid in	casconfig_usermods.lua file
Category	Server
Default	node.lua
Restriction	Any changes to node.lua should be made to node_usermods.lua.
See	“Understanding Configuration Files and Start-up Files”
Example	<code>cas.node='node.lua'</code>

cas.NWORKERS=number

specifies the number of worker nodes associated with this session.

Valid in	CAS statement SESSOPTS option casconfig_usermods.lua file
Category	Administration
Default	0
Range	0–5000
Example	<code>cas.nworkers=8</code>

cas.ONELOG=true | false

specifies that all server logging is written to a single file. When `cas.onelog=false`, each worker node creates its own log file.

The main controller and main worker processes each create their own log files.

Valid in	casconfig_usermods.lua file
Category	Log
Default	false
Interaction	<code>cas.onelog=true</code> is used with <code>cas.logcfgloc</code> and <code>cas.loghost</code> .
Note	The values <code>true</code> and <code>false</code> are case sensitive.
Example	In this example, each CAS worker node creates a log file: <code>cas.onelog=false</code>

cas.PERMSTORE='path'

specifies the path to a directory where the CAS server stores permissions.

Enclose *path* in single quotation marks.

The server saves its caslib and access control information to the `cas.permstore` directory periodically and when it shuts down.

Each subsequent time that the server starts, caslib and access control information is initialized from the server's `cas.permstore` location.

Note: When you update `cas.PERMSTORE` in `casconfig_usermods.lua`, you must also update `SASPERMSTORE` in `/opt/sas/viya/config/etc/sysconfig/cas/default/sas-cas-usermods`. (Ansible users should update `vars.yml` only.)

CAUTION! Backups of access controls are not automatically performed. It is strongly recommended that you periodically back up each CAS server's stored access control and caslib information. In particular, it is important to create a backup after you modify access controls or add, delete, or modify global caslibs. See [SAS Viya Administration: Backup and Restore](#).

Valid in [casconfig_usermods.lua file](#)

Category Access Control

Note Each CAS server should have its own `cas.permstore` location. To minimize the potential for network timing issues, it is recommended that each `cas.permstore` location be on the controller machine and not on a network file system. The server creates a directory with the name of the fully qualified DNS name of the machine that the main controller is running on in the specified permstore directory. Do not directly edit the files in a `cas.permstore` location.

Example Here is an example showing the CAS permstore location:
`cas.permstore='/opt/sas/viya/config/etc/cas/default/permstore'`

cas.PORT=port

specifies the port to which the CAS server listens.

The maximum allowable port number is 65535. If a valid port is not specified, the server listens on a port selected by the operating system through the TCP/IP ephemeral port range. A common range is 32768-61000.

The commonly configured port is 5570.

Valid in [casconfig_usermods.lua file](#)

Category Network

See [cas.GCPORT](#) and [cas.HTTPPORT](#)

Example `cas.port=5570`

cas.PROVLIST='ext' | 'kerb' | 'oauth'

specifies the authentication providers that the CAS server uses to authenticate incoming user connections.

- 'ext'

The external provider provides support for an external PAM authentication method when root access is required for authentication.

- 'kerb'

The Kerberos provider is used only when a Kerberos ticket is provided for authentication. For more information, see ["Kerberos Security" in SAS Viya for Linux: Deployment Guide](#).

- 'oauth'

OAuth provider is always loaded (even when not listed) to support REST endpoints and communications between CAS worker nodes and the controller.

Valid in [casconfig_usermods.lua file](#)

Category	Security
Default	oauth
Note	The CAS server configures the specified providers and uses each in order until an authenticated connection is successful.
Example	In this example, an external provider provides support for an external PAM authentication method. Although not specified, OAuth is always loaded to support REST endpoints and communications between CAS worker nodes and the controller. <code>cas.provlist='ext'</code>

cas.REMOVENODECANCELTIMEOUT=*interval*

when quiescing sessions in preparation for moving data from nodes that are being removed, the time that CAS waits for long-running actions to complete before canceling them.

Valid in [casconfig_usermods.lua file](#)

Category Server

Default 120 seconds

Note A value of zero indicates that a cancel request should never be sent.

Example `cas.removeNodeCancelTimeout='600'`

cas.REMOVENODEKILLTIMEOUT=*interval*

when quiescing sessions in preparation for moving data from nodes that are being removed, the time that CAS waits for a canceled action to stop before killing its session.

Valid in [casconfig_usermods.lua file](#)

Category Server

Default 15 seconds

Note A value of zero indicates that the sessions should never be killed.

See [cas.REMOVENODECANCELTIMEOUT on page 642](#)

Example `cas.removeNodeKillTimeout='300'`

cas.RESOLVEWORKERADDRESS=true | false

specifies how CAS list node actions return CAS worker node host names.

When `true`, CAS list node actions attempt to return the list of worker node host names. If the directory name service (DNS) lookup is unresponsive, CAS cancels the lookup and `resolveworkeraddress` is automatically set to `false`.

When `false`, list node actions return only the IP address of elastically added nodes.

TIP Setting `cas.RESOLVEWORKERADDRESS` to `false` ensures that the analytic cluster is less impacted by an unresponsive DNS configuration. However, some output is displayed as IP addresses instead of host names.

Valid in [casconfig_usermods.lua file](#)

Category Server

Default	True
See	cas.ELASTIC
Example	<code>resolveworkeraddress=false</code>

cas.SERVICESBASEURL='URL'

specifies the URL that enables CAS server to authenticate and to use SAS Viya services. *URL* points to the deployed SAS Viya web services.

When set, `cas.SERVICESBASEURL` creates a hybrid authentication environment where username-password authentication is converted to an OAuth token and CAS can fetch groups from SAS Viya services and use features such as the credentials vault.

Valid in	casconfig_usermods.lua file
Category	Security
Notes	<i>URL</i> must match the reverse proxy server host name and port to enable CAS to communicate with SAS Viya web services. Enclose <i>URL</i> in single quotation marks.
Example	<code>cas.servicesbaseurl='http://company.example.com'</code>

cas.STARTUP='filename'

specifies the configuration file that the CAS server runs before the server accepts any client connections. This start-up file contains CAS actions that the server runs as it starts up.

Valid in	casconfig_usermods.lua file
Category	Server
Default	<code>casstartup.lua</code>
Restriction	Any changes to <code>casstartup.lua</code> should be made to <code>casstartup_usermods.lua</code> .
See	“Understanding Configuration Files and Start-up Files”
Example	<code>cas.startup='casstartup.lua'</code>

cas.STARTUPDIR='path'

specifies the location for the SAS Cloud Analytic Services start-up directory.

Valid in	casconfig_usermods.lua file
Category	Server
Default	<code>/opt/sas/viya/config/default</code>
See	“Understanding Configuration Files and Start-up Files”
Example	<code>cas.startupdir='/opt/sas/viya/config/default/my_cas_startup'</code>

cas.SUBSETSESSIONCOPIES=number-of-blocks

specifies the number of extra block copies made for failover in either of the following scenarios:

- a session is smaller than the full server.
- CAS reads blocks of an HDFS remotely.

Valid in	CAS statement SESSOPTS option casconfig_usermods.lua file
Category	Administration
Default	0
Example	<code>cas.subsetsessioncopies=3</code>

cas.TAG=*string*

specifies a string to name the CAS server instance that is visible in the operating system, such as in the process list.

The cas.TAG option can be useful when debugging CAS.

Valid in	casconfig_usermods.lua file
Category	Server
Example	In this example, the string 'cas-my_tag' is used to name the CAS server instance: <code>cas.tag='cas-my_tag'</code> When you view the process list from a Linux command prompt, you see something similar to the following: <pre>27019 ? 00:00:01 cas-my_tag</pre>

cas.TENANTID=*string*

specifies the user ID for the CAS tenant. cas.TENANTID is used to validate that the authenticating user belongs to the correct CAS tenant.

Valid in	casconfig_usermods.lua file
Category	Administration
Restriction	Used only in multi-tenant CAS deployments.
Example	<code>cas.tenantid='tenant1'</code>

cas.TIMEOUT=*seconds*

specifies the SAS Cloud Analytic Services session time-out in seconds for a new or existing session.

Valid in	CAS statement SESSOPTS option casconfig_usermods.lua file
Category	Session
Default	In order of descending precedence: 1. CAS statement TIMEOUT= option value, if specified 2. SAS system option CASTIMEOUT=, if you explicitly set it in SAS to a value greater than 0 3. 60
Range	0–31536000
Notes	The session time-out starts when the number of connections to the session becomes zero and no actions are executing.

If a connection is established before the time-out expires, the time-out is canceled. Otherwise, the session is automatically terminated when the time-out expires.

When set to 0, the session is terminated immediately when the connection count becomes zero.

See [CASTIMEOUT= System Option](#)

Example `cas.timeout=100`

cas.USERLOC='%HOME' | 'pathname/%USER'

specifies that the CAS server create a personal caslib for each user at session start-up time in the specified location.

'%HOME' equates to the user's operating system \$HOME directory.

'pathname/%USER' refers to a directory named for the user's user ID under the specified file system path.

Enclose *pathname* in single quotation marks.

Valid in [casconfig_usermods.lua file](#)

Category **Caslib**

Examples In this example, the personal caslib directory is the user's operating system \$HOME directory:

```
cas.userloc='%HOME'
```

In this example, the user's personal caslib directory is named for his or her user ID and is located under `/local`:

```
cas.userloc='/local/%USER'
```

cas.USEYARN=true | false

adds a reservation request to YARN for CAS memory size.

The memory limit for the YARN request is set with [cas.memorysize](#).

Valid in [casconfig_usermods.lua file](#)

Category **Administration**

Default **false**

See [cas.MEMORYSIZE on page 638](#)

"How to Limit Memory Use" in *SAS Cloud Analytic Services: Fundamentals*

Example `cas.useyarn=true`

Grouped by Categories

Access Control Options

- [cas.PERMSTORE on page 640](#)

Action Options

- [cas.CMPOPT on page 628](#)

Administration Options

- `cas.SUBSETSESSIONCOPIES` on page 643
- `cas.ELASTIC` on page 631
- `cas.LIFETIME` on page 636
- `cas.MACHINELIST` on page 637
- `cas.MEMORYSIZE` on page 638
- `cas.MODE` on page 639
- `cas.NWORKERS` on page 640
- `cas.TENANTID` on page 644
- `cas.USEYARN` on page 645

Caslib Options

- `cas.COLOCATION` on page 629
- `cas.MAXTABLEMEM` on page 638
- `cas.USERLOC` on page 645

Data Options

- `cas.HDFSUSERLOC` on page 633

Data Quality Options

- `cas.DQLOCALE` on page 631
- `cas.DQSETUPLOC` on page 631

DATA Step Options

- `cas.DATASTEPFMterr` on page 629
- `cas.DATASTEPMSGSUMLEVEL` on page 630
- `cas.DATASTEPREPLACETABLE` on page 630

Formats Options

- `cas.FMTSEARCH` on page 632

Input Control Options

- `cas.EVENTDS` on page 632
- `cas.INTERVALDS` on page 635

Java

- `cas.JREOPTIONS` on page 635

Localization

- `cas.LOCALE` on page 636

Log Options

- [cas.LOGCFGLOC](#) on page 636
- [cas.LOGFLUSHTIME](#) on page 636
- [cas.MESSAGELEVEL](#) on page 639
- [cas.METRICS](#) on page 639
- [cas.ONELOG](#) on page 640

Network Options

- [cas.DCHOSTNAMERESOLUTION](#) on page 630
- [cas.GCPORT](#) on page 633
- [cas.HTTPPORT](#) on page 633
- [cas.HTTPPORTMAX](#) on page 634
- [cas.PORT](#) on page 641

Security Options

- [cas.ELASTICSSL](#) on page 632
- [cas.KEYFILE](#) on page 635
- [cas.PROVLIST](#) on page 641
- [cas.SERVICESBASEURL](#) on page 643

Note:

Other security-related configuration file options can be found in [“Configuration File Options for Data Transfer”](#) on page 443.

Server Options

- [cas.INITIALBACKUPS](#) on page 634
- [cas.INITIALWORKERS](#) on page 634
- [cas.MAXSESSIONS](#) on page 637
- [cas.NODE](#) on page 640
- [cas.REMOVENODECANCELTIMEOUT](#) on page 642
- [cas.REMOVENODEKILLTIMEOUT](#) on page 642
- [cas.RESOLVEWORKERADDRESS](#) on page 642
- [cas.STARTUP](#) on page 643
- [cas.STARTUPDIR](#) on page 643
- [cas.TAG](#) on page 644

Session Options

- [cas.APPTAG](#) on page 627
- [cas.TIMEOUT](#) on page 644

Sort Options

- [cas.COLLATE](#) on page 629

CAS Environment Variables

Where Do I Set CAS Environment Variables?

With a few exceptions, you set SAS Cloud Analytic Services environment variables in the CAS controller's configuration file, `casconfig_usermods.lua`. During CAS server start-up, the controller shares the configuration as each worker and the backup controller (if present) connects. By default, `casconfig_usermods.lua` is located in `/opt/sas/viya/config/etc/cas/default`.

If you want to isolate an environment variable change to a particular CAS node, then make your change in `node_usermods.lua` residing on the particular node machine.

A few CAS environment variables must be set before CAS starts and therefore you must add these variables to the `cas_usermods.settings` file. If you edit `cas_usermods.settings` on a single node, only that node is affected. If you want to set an environment variable on every node, you must edit `cas_usermods.settings` on every node. By default, `cas_usermods.settings` is located in `/opt/sas/viya/home/SASFoundation`.

TIP For sites that use Ansible, it is recommended that you make your CAS server environment variable changes to `vars.yml` and rerun Ansible to apply these changes. For more information, see [“Modify the vars.yml File”](#) in [SAS Viya for Linux: Deployment Guide](#).

There are additional CAS configuration files and directories. For more information, see [“Understanding Configuration Files and Start-up Files”](#).

CAUTION! SAS Cloud Analytic Services ignores any instance of `LD_LIBRARY_PATH` and `env.TKXTANIO_BINDAT_DIR` found in the server configuration file. Set `LD_LIBRARY_PATH` and `env.TKXTANIO_BINDAT_DIR` in the `cas_usermods.settings` file only.

CAS Environment Variables Reference

Note: For information about SAS Cloud Analytic Services TLS environment variables, see [“CAS TLS Environment Variables”](#) on page 435.

CAS_AUTH_METHOD=authinfo | kerberos

specifies the authentication method (authinfo) that CAS clients use.

Client	Optional
Valid in	operating system command line
Category	Security
Restriction	CAS_AUTH_METHOD is case-sensitive.
See	Authinfo File Authentication
Example	In this example, CAS clients are forced to use the authinfo file: <pre>export CAS_AUTH_METHOD=authinfo</pre>

env.CAS_ACTION_THREAD_NICE='niceness-priority'

specifies the niceness priority for the CPU intensive threads that do CAS action processing.

Use when the CAS server has to share CPU resources with other processes, and when the CAS server is incorrectly detecting disconnected worker nodes.

Valid in	casconfig_usermods.lua file
Category	Environment
Default	1
Range	0–19
Restriction	env.CAS_ACTION_THREAD_NICE is case-sensitive.
See	The man page for the Linux nice command.
Example	<code>env.CAS_ACTION_THREAD_NICE='1'</code>

env.CAS_DISK_CACHE=*path* [[:*path*] ...]

specifies the disk paths to cache data.

Delimit multiple paths with a colon (:).

Valid in	casconfig_usermods.lua file
Category	Data
Restrictions	env.CAS_DISK_CACHE is case-sensitive. Do not set to /tmp.
Tip	There is an advantage to using multiple physical disks. When using multiple threads, mapping files can occur concurrently if multiple disks are used. Hadoop also uses this method. Therefore, there is an advantage to using a set of disks that map to both <code>hadoop_data</code> and <code>CAS_DISK_CACHE</code> directories.
Example	<code>env.CAS_DISK_CACHE = '/data/disk1:/data/disk2'</code>

env.CAS_ENABLE_REMOTE_SAVE

when defined, specifies whether CAS saves blocks on remote HDFS worker nodes.

Valid in	casconfig_usermods.lua file
Category	Data
Restriction	env.CAS_ENABLE_REMOTE_SAVE is case-sensitive.
Example	<code>env.CAS_ENABLE_REMOTE_SAVE</code>

env.CAS_HEARTBEAT_LOST_TIMEOUT=*interval*

specifies the interval (in seconds) since the last heartbeat received from a CAS worker node before the controller treats the node as lost.

Smaller intervals detect machines that silently leave the network more quickly. Larger intervals are more tolerant of machines that might be exceptionally overloaded.

Valid in	casconfig_usermods.lua file
Category	Environment
Default	120 seconds

Range	60 – (no upper limit) seconds
Restriction	env.CAS_HEARTBEAT_LOST_TIMEOUT is case-sensitive.
Example	<code>CAS_HEARTBEAT_LOST_TIMEOUT='300'</code>

env.CAS_INSTALL='install-path'

specifies the installation directory for CAS.

Valid in	casconfig_usermods.lua file
Category	Environment
Restriction	env.CAS_INSTALL is case-sensitive.
Example	Here is an example showing the CAS installation directory: <code>env.CAS_INSTALL='/opt/sas/viya/home/SASFoundation'</code>

env.CAS_LICENSE='path/license-file'

specifies the path and filename that contains the CAS license.

After CAS deployment, env.CAS_LICENSE is set to `/opt/sas/viya/config/etc/cas/default/sas_license.txt`. The deployment process creates a Linux symbolic link between `sas_license.txt` and the actual SAS license file. You must change the symbolic link whenever the name of the license file changes. For more information, see [“Apply New Licenses Manually” on page 502](#).

Valid in	casconfig_usermods.lua file
Category	Administration
Restriction	env.CAS_LICENSE is case-sensitive.
Example	Here is an example showing the location of the CAS license: <code>env.CAS_LICENSE='/opt/sas/viya/config/etc/cas/default/SASViyaV0300_09JB84_Linux_x86-64.txt'</code>

env.CAS_REMOTE_HADOOP_PATH='SASHDAT-executables-directory'

specifies the path to the plug-in location when CAS is using an HDFS caslib to a remote HDFS cluster.

Valid in	casconfig_usermods.lua file
Category	Environment
Default	If not set, defaults to <code>\$HADOOP_HOME/bin</code>
Restriction	env.CAS_REMOTE_HADOOP_PATH is case-sensitive.
Note	Might be needed to accommodate a non-standard Hadoop plug-in.
Example	<code>env.CAS_REMOTE_HADOOP_PATH='\$HADOOP_HOME/bin'</code>

env.CAS_VIRTUAL_HOST='host-name'

The external host or machine name for the controller.

Use env.CAS_VIRTUAL_HOST when an external HTTP client needs to use an external address that differs from the actual host name known by the operating system. A common use is when the controller machine is behind a reverse proxy server.

Valid in	casconfig_usermods.lua file
----------	---

Category	Network
Restriction	env.CAS_VIRTUAL_HOST is case-sensitive.
Example	env.CAS_VIRTUAL_HOST='my_machine'

env.CAS_VIRTUAL_PATH='URL-path-suffix'

Use this environment variable when external HTTP clients must reach the CAS controller through a reverse proxy server. This identifies the path portion of the URL for the reverse proxy.

Valid in	casconfig_usermods.lua file
Category	Network
Restriction	env.CAS_VIRTUAL_PATH is case-sensitive.
Example	env.CAS_VIRTUAL_PATH='/cas-qstgrd-default-http'

env.CAS_VIRTUAL_PORT=port

The external port number for the controller.

Use env.CAS_VIRTUAL_PORT when an external HTTP client needs to use a port that differs from the actual port that is local to the controller machine. A common use is when the controller machine is behind a reverse proxy server.

Valid in	casconfig_usermods.lua file
Category	Network
Restriction	env.CAS_VIRTUAL_PORT is case-sensitive.
Example	env.CAS_VIRTUAL_PORT=5580

env.CAS_VIRTUAL_PROTOCOL='http | https'

Use this environment variable when external HTTP clients must reach the CAS controller through a reverse proxy server. This identifies the protocol portion of the URL for the reverse proxy.

Valid in	casconfig_usermods.lua file
Category	Network
Restriction	env.CAS_VIRTUAL_PROTOCOL is case-sensitive.
Example	env.CAS_VIRTUAL_PROTOCOL='https'

env.HADOOP_HOME='path'

specifies the standard HADOOP_HOME variable used by Hadoop.

Valid in	casconfig_usermods.lua file
Category	Data
Restriction	env.HADOOP_HOME is case-sensitive.
Example	env.HADOOP_HOME='/opt/hadoop'

env.HADOOP_NAMENODE=*machine-name* :*[machine-name]*

identifies which machines in the Hadoop cluster are NameNodes. There can be up to two Hadoop NameNodes. Separate machine names with a colon (:). *Machine-name* can be a name, fully qualified domain name, or an IP address for a machine.

Valid in [casconfig_usermods.lua file](#)

Category Data

Restriction env.HADOOP_NAMENODE is case-sensitive.

Example `env.HADOOP_NAMENODE='my_namenode1:my_namenode2'`

env.TKTXTANIO_BINDAT_DIR=*install-path*

specifies the installation directory for SAS linguistic binary files required to perform text analysis.

Note: This environment variable is valid only on native operating systems such as Linux.

Note: TKTGDat.sh contains the SAS linguistic binary files required to perform text analysis in SAS LASR Analytic Server with SAS Visual Analytics and to run PROC HPTMINE and HPTMSCORE with SAS Text Miner.

Valid in [cas_usermods.settings file](#)

Category Data

Restriction env.TKTXTANIO_BINDAT_DIR is case-sensitive.

Example `env.TKTXTANIO_BINDAT_DIR='/opt/sas/viya/home/SASFoundation/utilities/TKTGDat'`

LD_LIBRARY_PATH=*path* :*[[path]* ...]

specifies the path to search for additional shared libraries.

Separate multiple paths with a colon (:).

CAUTION! SAS Cloud Analytic Services ignores any instance of LD_LIBRARY_PATH found in the server configuration file. Set LD_LIBRARY_PATH in the cas_usermods.setting file only. Or, if your site uses Ansible, in vars.yml.

Valid in [cas_usermods.settings file](#)

Category Data

Restriction LD_LIBRARY_PATH is case-sensitive.

Notes Be careful when setting LD_LIBRARY_PATH. The path order can affect the operation of some applications.

The LD_LIBRARY_PATH export statement must be on a single line without wrapping.

Example `export LD_LIBRARY_PATH=/var/my_libs:/share/groups_libs:$LD_LIBRARY_PATH`

Grouped by Categories**Administration Variables**

- [env.CAS_LICENSE on page 650](#)

Data Variables

- [env.CAS_DISK_CACHE](#) on page 649
- [env.HADOOP_HOME](#) on page 651
- [env.HADOOP_NAMENODE](#) on page 652
- [env.CAS_ENABLE_REMOTE_SAVE](#) on page 649
- [env.TKXTANIO_BINDAT_DIR](#) on page 652
- [LD_LIBRARY_PATH](#) on page 652

Environment Variables

- [env.CAS_INSTALL](#) on page 650
- [env.CAS_REMOTE_HADOOP_PATH](#) on page 650

Network Variables

- [env.CAS_VIRTUAL_HOST](#) on page 650
- [env.CAS_VIRTUAL_PATH](#) on page 651
- [env.CAS_VIRTUAL_PORT](#) on page 651
- [env.CAS_VIRTUAL_PROTOCOL](#) on page 651

Security Variables





- [CAS_AUTH_METHOD](#) on page 648

Note: For information about SAS Cloud Analytic Services TLS environment variables, see “[CAS TLS Environment Variables](#)” on page 435.

SAS Cloud Analytic Services: Interfaces

There are several interfaces that you can use to administer a CAS server. The following table lists these interfaces and the shading indicates the relative amount of CAS administration that each covers:

Table A.50 *Interfaces to CAS Administration*

 CAS Server Properties action set	A programmatic interface for CASL (the CAS procedure), Python, Lua, and R. Used to display server option values.
 Ansible	A software orchestration tool that provides a straightforward approach to deploying and provisioning SAS Viya.
 Administrative scripts	Scripts used to operate CAS server, change the process owner account, and to convert from single- to multi-machine CAS.
 SAS Environment Manager	A graphical enterprise web application used to modify and view a subset of server properties and to adjust caslib management privileges.

● [CAS Server Monitor](#)

A graphical web application that is embedded in the CAS server. Used to view server information and to manage sessions, nodes, and caslib management privileges.

SAS Server Contexts

SAS Viya Server Contexts: Overview

Note: A [programming-only on page 15](#) deployment does not use server contexts.

To learn about SAS Viya server contexts, see “[Server Contexts: Concepts](#)” in [SAS Viya Administration: Server Contexts](#).



To create server contexts, see “[Server Contexts: How To](#)” on [page 655](#).

Server Contexts: How To

Introduction


These instructions explain how to view and modify server contexts using [SAS Environment Manager](#).

Navigation

In the applications menu () , under **Administration**, select **Manage Environment**. In the navigation bar, click .

The Contexts page is an advanced interface that is available to SAS Administrators only. If you are a SAS Administrator and the Contexts page is unavailable to you, then the programming run-time servers have not been deployed in your SAS Viya environment.


Create a Context

- 1 Under **View**, select the [type of context](#) that you want to create.
- 2 On the top left side of the Contexts page, click .
- 3 (Required) Enter a name for your context.
Names must not be longer than 40 characters and can consist of any alphanumeric and special characters.
- 4 If you are creating a [launcher context](#), skip to [Step 7](#). Otherwise, enter values for creating a [compute context](#):
 - **Description**
Enter a description of the context that you are creating.
 - (Required) **Launcher context**

Select a launcher context with which to run the SAS Compute Server.

■ (Required) **Identity type**

Select one of the following:

- Select **Authenticated users** in order for any authenticated user to use this context.
- Select **Identities** by clicking , and then select one or more users or groups to use this context.

- 5 To add any SAS options or additional autoexec file settings that the compute server processes use at start-up, select **Advanced** and enter this information in their respective fields.

For more information, see [Customizing Your SAS Session By Using Configuration and Autoexec Files](#).

- 6 When you are finished, click **Save** to create the compute context.

- 7 Enter values for creating a launcher context:


■ **Description**

Enter a description of the context that you are creating.

■ **Port Range**

Enter a range of ports. The SAS Launcher Server selects a port in the specified range in order to run the compute server.


■ **Environment Variables**

Click  and add the environment variable and its value that you want the launcher server to use when running the compute server.


- 8 Click **Advanced** to override the default server deployment settings that are used by the launcher service at launch time. Next, check the box and provide values for all the fields.

- 9 When you are finished, click **Save** to create the launcher context.

Edit a Context

- 1 Under **View**, select the [type of context](#) that you want to edit.
- 2 On the top right side of the Contexts page, click .
- 3 Make your modifications, and click **Save** when you are finished.

Delete a Context

- 1 Under **View**, select the [type of context](#) that you want to delete.
- 2 On the left side of the Contexts page, click .
- 3 Click **Yes** to confirm the deletion.

Server Contexts: Concepts

Compute Contexts

SAS Compute Servers are run under a compute context. (Contexts are analogous to SAS 9 SAS Application Servers.) A *compute context* is a specification that contains the information needed to run a compute server.

The information contained in a compute context is the user identity and any SAS options, or autoexec parameters to be used when starting the server.

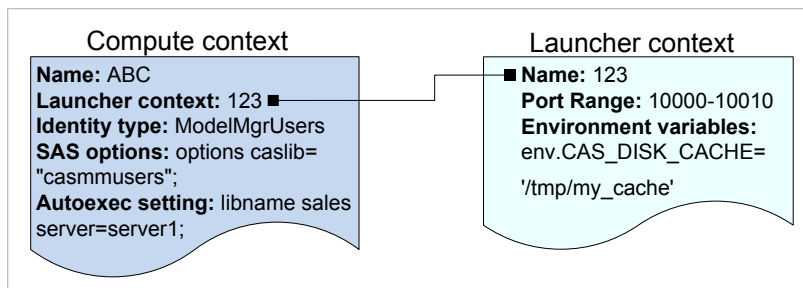
Launcher Contexts

The server that starts the compute server, SAS Launcher Server, itself requires a context.

A *launcher context* is a specification that enables SAS administrators to place environmental and access constraints on processes run by a launcher server.

For example, the administrator can specify a port range for a server process to use when binding a port, or environment variables that are defined in all processes started by the launcher server when it uses a particular launcher context.

Figure A.1 Context Types



SAS Server Contexts: Interfaces

There are several interfaces that you can use to manage SAS server contexts. The following table lists these interfaces and the shading indicates the relative amount of SAS server contexts administration that each covers:

Table A.51 Interfaces to SAS Viya Server Contexts

●	SAS Environment Manager	A graphical enterprise web application used to manage SAS server contexts.
●	Command-line interface	A command-line interface that enables you to manage SAS server contexts.

Programming Run-Time Servers

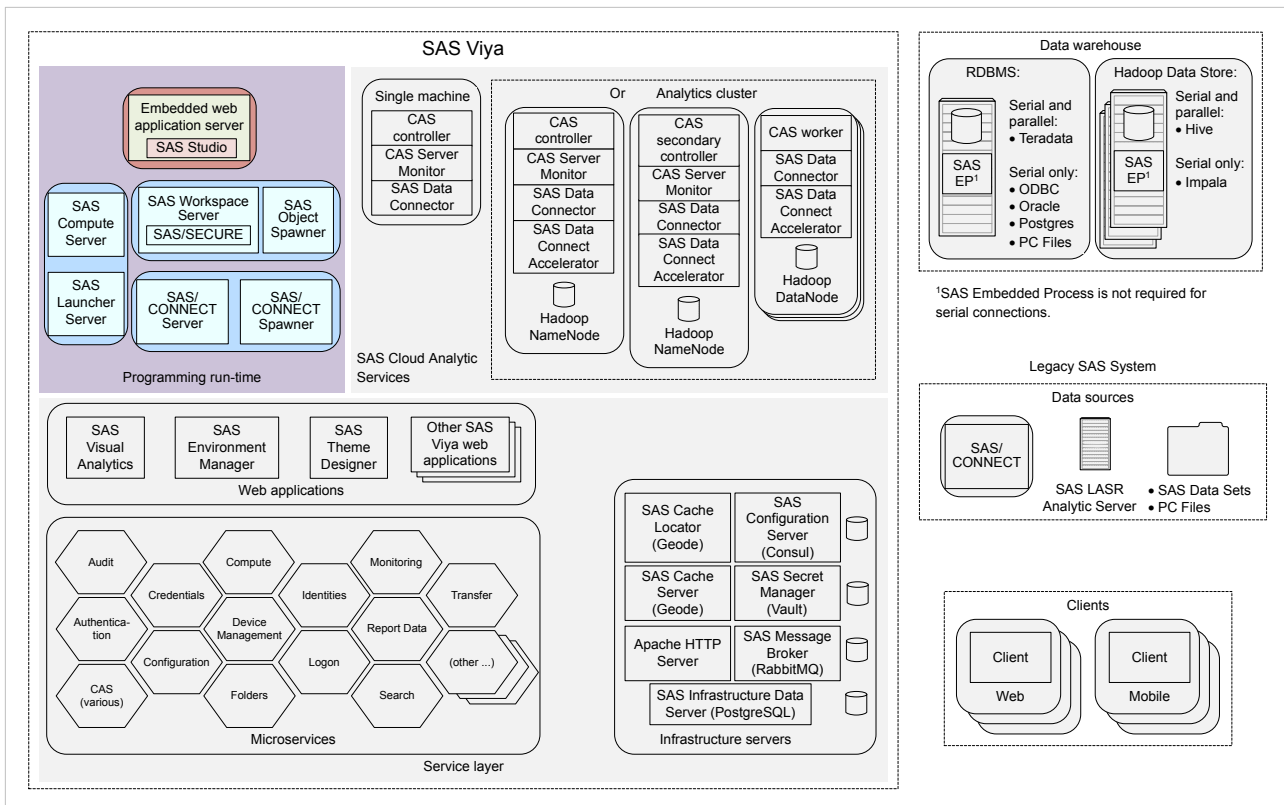
Programming Run-Time Servers: Overview

A programming run-time environment includes several SAS Viya servers. The following table lists the servers (and services, where applicable) and indicates which are available in a [programming-only on page 15](#) deployment:

Server	Full deployment	Programming-only deployment
"SAS Compute Server and Compute Service"	✓	
"SAS Launcher Server and Launcher Service"	✓	
"SAS Workspace Server and SAS Object Spawner"	✓	✓
"Embedded Web Application Server"	✓	✓
"SAS/CONNECT Server and SAS/CONNECT Spawner"	✓	✓

In the following diagram, the highlighted box shows the relationship of the programming run-time servers to other components in the SAS Viya environment:

Figure A.1 SAS Viya Programming Run-Time Servers



SAS Compute Server and Compute Service

Overview

The Compute service enables clients to submit SAS programs and stored procedures in the form of jobs for processing. The SAS Compute Server implements the Compute service. For more information, see [“Concepts” on page 661](#).

Operate the Compute Service

SAS Viya uses the operating system’s default init system or the `systemd` command to launch scripts that can stop, start, restart, and check the status of the compute service. The script `sas-viya-compute-default` resides in `/etc/init.d`.

Note: You must be logged on to the machine where the compute service resides, and you must have `sudo` privileges to run this script.

To operate the compute service, run the following command, as appropriate:

```
sas-viya-compute-default status | stop | start | restart
```

Note: You can use a script to operate and view the running state of all SAS Viya servers and services. For more information, see [“All Servers and Services” on page 599](#).

Here are a few examples of how to operate the compute service:

- To check status of the compute service using a direct call:

```
sudo /etc/init.d/sas-viya-compute-default status
```

- To stop the compute service using the Red Hat Linux version 6 init system command:

```
sudo service sas-viya-compute-default stop
```

- To start the compute service using the Red Hat Linux version 7 systemd command:

```
sudo systemctl start sas-viya-compute-default
```

- To restart the compute service using a direct call:

```
sudo /etc/init.d/sas-viya-compute-default restart
```

Concepts

SAS Compute Server

The SAS Compute Server enables clients to submit SAS programs and stored procedures in the form of jobs for processing using the SAS language. For every job that is processed, the compute server writes a logging message to a SAS log. The job produces results when output is created.

Note: The SAS Compute Server does not support X commands.

Compute servers are launched by a [SAS Launcher Server](#).

Compute Service

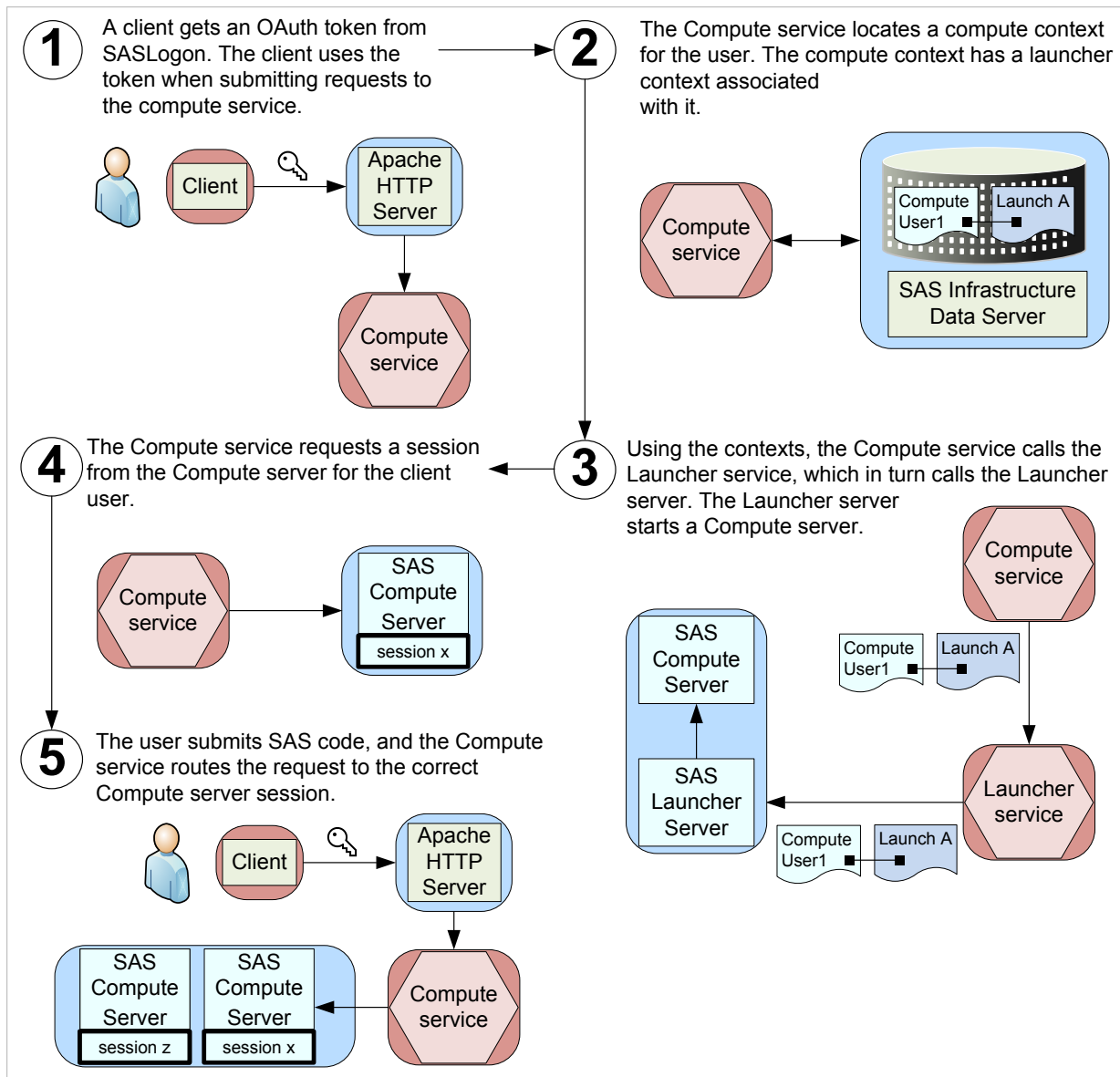
The Compute service is a SAS Viya microservice that provides API endpoints for requesting a SAS Compute Server session. The compute service also provides API endpoints for creating and managing [compute contexts](#), specifications that contain all the information that is needed to run a compute server.

The launcher service provides a specification to the launcher server called a [launcher context](#), which enables the SAS administrator to apply constraints for how the launcher server starts a compute server.

How It Works

The following figure describes how a SAS client submits code to the SAS Compute Server.

Figure A.2 How a Client Submits Code to the SAS Compute Server



Fault Tolerance

You are able to deploy SAS Compute Servers for fault tolerance. You can deploy multiple SAS Launcher Servers on multiple compute server machines, and the Launcher service randomly routes client requests among the registered Launcher servers.

Only machine-level fault tolerance is supported. If a machine goes down, and you have other machines running Launcher and Compute servers, then fault tolerance is applied. If an individual Launcher or Compute Server process abnormally terminates, then no fault tolerance is applied.

Log Files

Log files for the compute service are located in `/opt/sas/viya/config/var/log/compute/default`.

On multi-tenant systems, log files for the compute service are located in `/opt/sas/tenant-ID/config/var/log/compute/default`.

SAS Launcher Server and Launcher Service

Overview

The SAS Launcher Server runs processes in a SAS Viya environment. The Launcher service is a SAS Viya microservice that provides API endpoints for how the launcher server runs a process.

Operate the Launcher Service

SAS Viya uses the operating system's default init system or the `systemd` command to start, restart, and check the status of the launcher service. The script, `sas-viya-launcher-default` resides in `/etc/init.d`.

Note: You must be logged on to the machine where the launcher service resides, and you must have `sudo` privileges to run this script.

To operate the launcher service, run the following command, as appropriate:

```
sas-viya-launcher-default status | stop | start | restart
```

Note: You can use a script to operate and view the running state of all SAS Viya servers and services. For more information, see [“All Servers and Services” on page 599](#).

Here are a few examples of how to operate the launcher service:

- To check status of the launcher service using a direct call:

```
sudo /etc/init.d/sas-viya-launcher-default status
```

- To stop the launcher service using the Red Hat Linux version 6 init system command:

```
sudo service sas-viya-launcher-default stop
```

- To start the launcher service using the Red Hat Linux version 7 `systemd` command:

```
sudo systemctl start sas-viya-launcher-default
```

- To restart the launcher service using a direct call:

```
sudo /etc/init.d/sas-viya-launcher-default restart
```

Concepts

SAS Launcher Server

The SAS Launcher Server starts processes, stops processes, and checks the status of processes in a SAS Viya environment.

For information about clustering, see [“Fault Tolerance”](#).

Launcher Service

The launcher service is a SAS Viya microservice that provides API endpoints for how the launcher server runs a process. These API endpoints are used to create and manage [launcher contexts](#).

Troubleshooting

Failure to launch Compute server sessions

Explanation:

Here are some reasons why a Compute server fails to launch:

- The user account under which the client is running does not have a home directory on the machine where the Compute server resides.
- Client users in a multi-tenant environment have to be a member of the sas group on the machine where the Compute server resides.
- Kerberos is present.

Resolution:

Check for the preceding issues in logs for the client application, [Compute service](#) , and [Launcher service](#).

Log Files

Log files for the launcher service are located in `/opt/sas/viya/config/var/log/launcher/default`.

On multi-tenant systems, log files for the launcher service are located in `/opt/sas/tenant-ID/config/var/log/launcher/default`.

SAS Workspace Server and SAS Object Spawner

Overview

The SAS Workspace Server enables client programs to access SAS libraries, to perform tasks by using the SAS language, and to retrieve the results. One or more SAS Workspace Servers are initialized by the SAS Object Spawner.

How To**Operate**

SAS Viya uses the operating system's default init system command or the systemd command to launch a script that can stop, start, restart, and check the status of the SAS Object Spawner. This script, `sas-viya-spawner-default`, resides in `/etc/init.d`.

Note: You must be logged on to the machine where the object spawner resides, and you must have sudo privileges to run this script.

To the operate the object spawner, run the following command, as appropriate:

```
sas-viya-spawner-default status | stop | start | restart
```

Note: You can use a script to operate and view the running state of all SAS Viya servers and services. For more information, see [“All Servers and Services” on page 599](#).

Here are a few examples of how to operate the object spawner:

- To check status of the object spawner using a direct call:


```
sudo /etc/init.d/sas-viya-spawner-default status
```
- To stop the object spawner using the Red Hat Linux version 6 init system command:


```
sudo service sas-viya-spawner-default stop
```
- To start the object spawner using the Red Hat Linux version 7 systemd command:


```
sudo systemctl start sas-viya-spawner-default
```

- To restart the object spawner using a direct call:

```
sudo /etc/init.d/sas-viya-spawner-default restart
```

Enable X Commands

Because clients can use host commands to perform potentially harmful operations such as file deletion, by default, X commands are disabled for [the SAS Object Spawner](#). However, to enable X commands, follow these steps:

- 1 Log on to the machine on which the object spawner resides.
- 2 Using a text editor, open `/opt/sas/viya/config/etc/spawner/default/spawner_usermods.sh`.
- 3 Add the following line, save, and close the `spawner_usermods.sh` file:

```
USERMODS="$JREOPTIONS -allowxcmd"
```

- 4 Restart the object spawner:

```
sudo service sas-viya-spawner-default restart
```

Set umask or ulimit Values

To control umask and ulimit sessions for the SAS Workspace Server, modify the `workspaceserver_usermods.sh` file. To affect umask and ulimit sessions for the SAS/CONNECT Server, modify the `connectserver_usermods.sh` file. To set umask and ulimit settings for the SAS Workspace Server, the SAS/CONNECT server, and all SAS instances, modify `sasenv_local`.

- 1 Log on to the machine on which the SAS Workspace Server or the SAS/CONNECT Server resides. Log on as the SAS install user or log on with sudo privileges.
- 2 Using a text editor, open one of the following files, as appropriate:

- For the SAS Workspace Server:

```
/opt/sas/viya/config/etc/workspaceserver/default/workspaceserver_usermods.sh
```

- For the SAS/CONNECT Server:

```
/opt/sas/viya/config/etc/connectserver/default/connectserver_usermods.sh
```

- For the SAS Workspace Server, the SAS/CONNECT Server, and all SAS instances:

```
/opt/sas/spre/home/SASFoundation/bin/sasenv_local
```

- 3 Add your umask and ulimit values, and save the file.

Your changes take effect the next time the server or servers are launched.

TIP The umask and ulimit settings can be set for all users (or values can be set conditionally for each user), for collections of users, or for all members of a given Linux group. For more information, see [“Examples of umask and ulimit Settings”](#).

Examples of umask and ulimit Settings

In the following example, the umask command creates all files for all users with effective permissions of `rw-r--r--` (owner:read and write; group:read; other:read):

```
umask 022
```

In the following example, umask is set for user joe00001 only:

```
if [ "$LOGNAME" = joe00001 ]
then
umask 022
fi
```

In the following example, ulimits are set according to user ID or group membership.

```
# determine primary group membership of user
# GP=`groups $LOGNAME | awk '{ print $1 }'`
# assign new ulimit based on userid or group membership as desired
if [ "$LOGNAME" = joe00001 ]
then
MAXSIZE=4096
umask 022
elif [ "$LOGNAME" = fred0002 -o "$GP" = saspower ]
then
MAXSIZE=8192
umask 077
elif [ "$GP" = sasuser ]
then
MAXSIZE=6144
else
MAXSIZE=8192
fi

export MAXSIZE

ulimit -f $MAXSIZE
```

Lock Down SAS Workspace Servers

Using the [LOCKDOWN system option](#) and the [LOCKDOWN statement](#), you can limit access to files and to specific SAS features in a SAS Workspace Server session that executes in a batch mode or a server processing mode in a multi-environment deployment.

To lock down one or more workspace servers, follow these steps:

- 1 With administrator privileges, log on to the machine that contains the workspace server.
- 2 Create a lockdown path list (a whitelist) that contains all the paths that are accessible to the server, and add it to `/opt/sas/viya/config/etc/workspaceserver/default/autoexec_usermods.sas`.

Note:

A path that is declared in the whitelist does not mean that an arbitrary user can read any file in that path. Host permissions on physical files and directories always take precedence over the whitelist. SAS adds certain predefined paths from the SAS configuration file by default. For more information, see [LOCKDOWN Statement Details](#).

TIP For a suggestion about how to implement the whitelist, see [Example 2: Hiding the Whitelist By Locating the Path outside the Whitelist](#).

- 3 Add the following line to `/opt/sas/viya/config/etc/workspaceserver/default/sasv9_usermods.cfg`:

```
-lockdown
```

Changes to the `autoexec_usermods.sas` file are automatically included when the workspace server scripts run. Your changes will take effect the next time SAS starts a workspace server session.

- 4 If your site uses SAS Studio, set `webdms.showSystemRoot=false`.
For more information, see [“Update SAS Studio Configuration Properties”](#) on page 217.
- 5 If your site uses SAS/CONNECT, see [“Lock Down the SAS/CONNECT Server”](#) on page 671.

Restricting SAS System Options

You can restrict SAS system options so that they cannot be changed by a user. An option can be restricted globally, by group, or by user.

Global Restrictions

Create the `/opt/sas/viya/home/SASFoundation/misc/rstropts/rsasv9.cfg` file and add options to this file in the normal configuration file format.

Group Restrictions

Create the `/opt/sas/viya/home/SASFoundation/misc/rstropts/groups/groupname_rsasv9.cfg` file and add options to this file in the normal configuration file format.

For example, for user smith in the group staff, the filename would be `staff_rsasv9.cfg`.

User Restrictions

Create the `/opt/sas/viya/home/SASFoundation/misc/rstropts/users/username_rsasv9.cfg` file and add options to this file in the normal configuration file format.

For example, for user smith, the filename would be `smith_rsasv9.cfg`.

Concepts

SAS Workspace Server

The SAS Workspace Server enables client programs to access SAS libraries, to perform tasks by using the SAS language, and to retrieve results. Each workspace server process is owned by the client user that made the server request.

SAS Object Spawner

SAS Object Spawners interact with SAS by creating a server process for each client connection. SAS Workspace Servers are initialized by the SAS Object Spawner. An object spawner runs on the same machine as the workspace server, listens for requests, and launches the servers as necessary.

SAS Workspace Servers and SAS Cloud Analytic Services

In a SAS Viya environment, you can set up your `autoexec.sas` file to start a CAS session automatically. If you opt for automatic CAS session start-up, SAS uses that CAS session whenever it needs to communicate with SAS Cloud Analytic Services.

Many SAS procedures that are used in a SAS Viya deployment (such as PROC CARDINALITY and PROC NNET) use the CAS engine to communicate with CAS. The CAS engine uses the CAS session. In this context, the workspace server is used to interpret your SAS program and to determine how to run the lower-level actions in CAS.

Use of the `SESSREF= DATA` statement option in a SAS program is another method to inform the workspace server that CAS is being used. To run a DATA step in CAS, you must use a libref from the CAS engine, and you must specify the CAS session name in the `SESSREF=` option. When the workspace server interprets these language elements, it knows to run your DATA step in CAS.

In a SAS Viya environment, the workspace server is also used to do some work outside of CAS. Here are two examples:

- Creating graphics with procedures like PROC SGPLOT does not run in CAS. The data might be read from CAS with a CAS engine libref, but the graphics are created with the workspace server.
- Some data processing (such as with the INFILE, INPUT, and related DATA step statements and functions) do not use CAS. These data processing statements are run in the workspace server session so that the contents of external files can be read before the data can be transferred to CAS for analysis.

SAS Object Spawner Invocation

The SAS Object Spawner uses an `sudo` root program, called `elssrv`, to launch processes under the identity of the requesting client. The user ID must be root in order to switch the identity to another user.

When launching a SAS Workspace Server, the client provides host credentials for the user that is requesting the SAS process to the spawner. The spawner host authenticates the client and receives confirmation of valid credentials from `sasauth`. In addition, `sasauth` returns the UNIX uid and the list of groups. The `sudo` root program launches the workspace server under this identity so that the process runs with the host authority of the requesting client.

Embedded Web Application Server

Overview

The embedded Apache Tomcat server that is used in all of the SAS Viya web applications provides the execution environment for SAS Studio.

How To

Operate

SAS Viya uses the operating system's default `init` system command or the `systemd` command to launch a script that can stop, start, restart, and check the status of SAS Studio. This script, `sas-viya-sasstudio-default`, resides in `/etc/init.d`.

Note: You must be logged on to the machine where SAS Studio resides, and you must have `sudo` privileges to run this script.

To operate SAS Studio, run the following command, as appropriate:

```
sas-viya-sasstudio-default status | stop | start | restart
```

Note: You can use a script to operate and view the running state of all SAS Viya servers and services. For more information, see [“All Servers and Services” on page 599](#).

Here are a few examples of how to operate SAS Studio:

- To check status of SAS Studio using a direct call:

```
sudo /etc/init.d/sas-viya-sasstudio-default status
```

- To stop SAS Studio using the Red Hat Linux version 6 `init` system command:

```
sudo service sas-viya-sasstudio-default stop
```

- To start SAS Studio using the Red Hat Linux version 7 systemd command:

```
sudo systemctl start sas-viya-sasstudio-default
```

- To restart SAS Studio using a direct call:

```
sudo /etc/init.d/sas-viya-sasstudio-default restart
```

Configure Mail

To use the email functionality in SAS Studio, an SMTP server and the following information is required:

- *fully-qualified-SMTP-server-name*

The fully qualified host name of the SMTP server for the outbound mail (for example, `my_mail_server.example.com`).

- *SMTP-server-port*

The port for the SMTP server (for example, 25).

- *site-administrator-email-address*

The user name that accesses the SMTP server.

This user name is not necessarily the person who is sending the mail.

- *site-administrator-password*

The password for the user name that accesses the SMTP server.

- *company-domain*

The domain name for your site (for example, `my_company.example.com`).

To configure SAS Studio for SMTP email, follow these steps:

- 1 Log on to the machine on which the embedded web application server resides.
- 2 Using a text editor, open `/opt/sas/viya/config/etc/sasstudio/default/init_usermods.properties`.
- 3 Add the following lines, save, and close the `init_usermods.properties` file:

```
webdms.SMTP.hostName=fully-qualified-SMTP-server-name
```

```
webdms.SMTP.port=SMTP-server-port
```

```
webdms.SMTP.user=site-administrator-email-address
```

```
webdms.SMTP.password=site-administrator-password
```

```
webdms.domain=company-domain
```

- 4 Restart the embedded web application server:

```
sudo service sas-viya-sasstudio-default restart
```

When sending email, the sender address is derived from the user name that logged on to SAS Studio and the value of the `webdms.domain` property in the `appserver_usermods.sh` file. For example, if the user name is `test`, the sender address would be `test@your-company.com`.

SAS/CONNECT Server and SAS/CONNECT Spawner

Overview

SAS/CONNECT software provides the essential tools for sharing data and processing power across multiple computing environments:

- For SAS 9 users, SAS/CONNECT enables you to use SAS Viya functionality and features.
For more information, see [“SAS 9 and SAS Viya” on page 22](#).
- For SAS Viya users who might also have SAS 9, SAS/CONNECT provides parallel processing for CAS procedures.
For more information, see http://documentation.sas.com/?cdcId=pgmsascdc&cdcVersion=9.4_3.3&docsetId=viyaconnref&docsetTarget=titlepage.htm

How To

Operate

SAS Viya uses the operating system's default init system command or the systemd command to launch a script that can stop, start, restart, and check the status of SAS/CONNECT Spawner. This script, `sas-viya-connect-default`, resides in `/etc/init.d`.

Note: You must be logged on to the machine where the spawner resides, and you must have sudo privileges to run this script.

To operate the spawner, run the following command, as appropriate:

```
sas-viya-connect-default status | stop | start | restart
```

Note: You can use a script to operate and view the running state of all SAS Viya servers and services. For more information, see [“All Servers and Services” on page 599](#).

Here are a few examples of how to operate the spawner:

- To check status of the spawner using a direct call:

```
sudo /etc/init.d/sas-viya-connect-default status
```
- To stop the spawner using the Red Hat Linux version 6 init system command:

```
sudo service sas-viya-connect-default stop
```
- To start the spawner using the Red Hat Linux version 7 systemd command:

```
sudo systemctl start sas-viya-connect-default
```
- To restart the spawner using a direct call:

```
sudo /etc/init.d/sas-viya-connect-default restart
```

Set Configuration Options

Use `/opt/sas/viya/config/etc/connect/default/connect_usermods.sh` to set options such as encryption options, or you can use the SASCMD option in the SAS/CONNECT Spawner invocation.

Your changes take effect the next time the spawner is restarted.

Lock Down the SAS/CONNECT Server

Using the [LOCKDOWN system option](#) and the [LOCKDOWN statement](#), you can limit access to files and to specific SAS features in a SAS/CONNECT server session in a multi-environment deployment.

To lock down your SAS/CONNECT server, follow these steps:

- 1 With administrator privileges, log on to the machine that contains the SAS/CONNECT Server.
- 2 If you have not done so already, create a lockdown path list (a whitelist) that contains all the paths that are accessible to the server, and add it to `/opt/sas/viya/config/etc/connectserver/default/autoexec_usermods.sas`.

TIP If you have already locked down your SAS/CONNECT server, then this step is unnecessary.

Note:

A path declared in the whitelist does not mean that an arbitrary user can read any file in that path. Host permissions on physical files and directories always take precedence over the whitelist. SAS adds certain predefined paths from the SAS configuration file by default. For more information, see [LOCKDOWN Statement Details](#).

TIP For a suggestion about how to implement the whitelist, see [Example 2: Hiding the Whitelist By Locating the Path outside the Whitelist](#).

- 3 Add the following line to `/opt/sas/viya/config/etc/connectserver/default/sasv9_usermods.cfg`:


```
-lockdown
```

Changes to the `autoexec_usermods.sas` file are automatically included when the SAS/CONNECT server scripts run. Your changes will take effect the next time SAS starts a SAS/CONNECT server session.

Note: Do not start the SAS/CONNECT spawner using the `-SHELL` option. As long as the `-SHELL` option is *not* specified, the `-NOXCMD` option is added by default to the server's invocation parameters. `-NOXCMD` prevents clients from executing X commands from their SAS sessions to access system files.

- 4 If your site uses SAS Studio, set `webdms.showSystemRoot=false`.
For more information, see ["Update SAS Studio Configuration Properties"](#) on page 217.

Concepts

SAS/CONNECT software provides the essential tools for sharing data and processing power across multiple computing environments.

Note: SAS/CONNECT is ordered and licensed separately from other SAS Viya products.

SAS code uses these tools to perform tasks such as the following:

- dividing time-consuming tasks into multiple units of work and executing these units in parallel
- moving data from a client machine to a server machine (including legacy data from SAS 9), or vice versa, so that the data is on the same machine as the code processing it

Reference

Server Environment Variables

The following SAS/CONNECT Server environment variables are available for configuring your TCP/IP connections. Place them in the `/opt/sas/viya/config/etc/connectserver/default/connectserver_usermods.sh` script file. For information about configuring environment variables in a Linux environment, see [Defining Environment Variables in UNIX Environments](#).

CONNECTWDWAIT=<seconds>

Specify to limit the possibility that a client session disconnect might orphan a runaway DMR mode session. To ensure the responsiveness of the spawner, SAS starts a “watchdog” thread to monitor the connection. The default interval is five seconds. If a disconnect occurs, CONNECTWDWAIT checks 18 times and then terminates the DMR thread (for a default elapsed time of 90 seconds). Setting the CONNECTWDWAIT value to zero means that the process does not monitor the connection.

Defaults interval: 5 seconds

total elapsed time: 90 seconds

Examples In the following example, the option is set to 10, so the process waits 180 seconds, and then terminates the thread:

```
set CONNECTWDWAIT=10
```

In the following example, the option is set to 0, so the process does not monitor the connection:

```
set CONNECTWDWAIT=0
```

TCP_POLL_INTERVAL=<seconds>

Specify to ensure responsiveness of SAS spawners and servers to various conditions outside of normal request processing. When idle, servers and spawners periodically awaken to check for requests. The interval in seconds for this check is governed by the TCP_POLL_INTERVAL environment variable. Generally, the default setting of 60 seconds should be acceptable.

A value of zero means the server remains idle and awakens for request processing only.

Example In the following example, the option is set to 50, so the process checks every 50 seconds for a connection:

```
TCP_POLL_INTERVAL=50
```

TCPMSGLEN=<size>

Specifies the size of the buffer (in bytes) that the TCP/IP access method uses for breaking up a message that it sends to or receives from the SAS/CONNECT application layer during a SAS/CONNECT session. The application layer uses a message size that is stored in the TBUFSIZE option that you can specify in the SIGNON statement or as a SAS option.

If TBUFSIZE is larger than TCPMSGLEN, the TCP/IP access method breaks the message into a buffer whose size is defined by TCPMSGLEN, and issues the number of send and receive messages that are necessary to complete the message transaction.

The value for TCPMSGLEN must be set at both the client and the server. If the values that are set for TCPMSGLEN at the client and at the server are different, the smaller value of the two is used during the SAS/CONNECT session. If the TCPMSGLEN environment variable is not set, SAS uses the TCP stack’s default size and allows autotuning if implemented by the stack.

Client Optional

Server Optional

Example set TCPMSGLEN=65536

Spawner General Options

-CLEARTEXT

Allows sign-ons from clients that do not support user ID and password encryption. This option allows clients that are running older releases (prior to SAS 6.09E and SAS 6.11 TS040, which do not support user ID and password encryption) to sign on to the spawner program. Use this option only when absolutely necessary because credentials are transmitted unencrypted. The default encodes all communications.

-DEBUG

Turns on debug level output.

-HELP

Specifies to print the Help message.

-LOG | -LOGFILE <filename>

Specifies the filename to use for spawner log output if you are not using the -LOGCONFIGLOC option. The -LOG option should not be used with the -LOGCONFIGLOC option. If both options are specified, then the -LOGCONFIGLOC option takes precedence.

You can specify the -DEBUG or -TRACE options with the -LOG <filename> option to cause the spawner to send detailed log messages to a log file.

Example In this example, the following option is enclosed in double quotation marks and added after USERMODS= in /opt/sas/viya/config/etc/connect/default/connect_usermods.sh. When the spawner starts, it sends debug-level log messages to a file named sas-connect.log:

```
USERMODS="-log /var/log/sas/viya/connect/default/sas-connect.log"
```

-LOGCONFIGLOC <filename>

Enables the SAS logging facility for SAS servers and names the location of the configuration file that is used by the SAS logging facility to create spawner log output. The configuration file is an XML file that specifies and configures loggers and appenders for the SAS/CONNECT spawner.

The file specification that defines the location of the XML configuration file must be a valid filename or a path and filename for your operating environment. If the path contains spaces, enclose the file specification in quotation marks.

Note If LOGCONFIGLOC is specified, spawner messages are routed by default to the App.Connect.Spawner logger.

-NOCLEARTEXT

Prevents sign-ons from clients that do not support user ID and password encryption. This option prevents clients that are running older releases (prior to SAS 6.09E and SAS 6.11 TS040, which do not support user ID and password encryption) from signing on to the spawner program. However, the default permits both encrypted and plaintext user IDs and passwords.

-NOINHERITANCE

Disables socket inheritance.

Socket inheritance enables SAS/CONNECT servers to use the socket connection that is established between the SAS/CONNECT client and the spawner. Socket inheritance saves resources and is easier to configure when clients connect to a server that is within a firewall.

Default Socket inheritance is on.

-NOSCRIPT

Prevents sign-on from clients that use scripts, and allows sign-on only from clients that do not use scripts.

-NOSCRIP can be useful if you want to limit SAS start-up commands to the use of the -SASCMD option. Specifying -NOSCRIP restricts clients from specifying additional options in SAS start-up commands or script files.

Requirement Must be used with -SASCMD

-SASCMD | -CMD <command>

Specifies the SAS command or a command file that starts a SAS session when you sign on without a script. If the client does not specify a script file at sign-on, the -SASCMD option must be specified when starting the spawner.

Example In this example, the following option is enclosed in double quotation marks and added after USERMODS= in `/opt/sas/viya/config/etc/connect/default/connect_usermods.sh`. When the spawner starts, it uses a command file named `mystartup`:

```
USERMODS="-sascmd '/u/username/mystartup' "
```

Here is a sample command file named `mystartup`:

```
#!/bin/ksh
#-----
# mystartup
#-----
. ~/.profile
sas -noterminal -nosyntaxcheck $*
#-----
```

The `$*` positional parameter enables you to specify additional SAS options when you invoke SAS. In addition, `$*` also allows the options that the spawner adds automatically, like -DMR, to be included in the server session.

-SASDAEMONSERVICE <service-name | port>

Specifies the service name or port number that the SAS/CONNECT spawner uses to listen for child SAS/CONNECT server process connections.

If you use a service, its name must be configured in the SERVICES file on the computer that the SAS/CONNECT server session runs on.

-SERVICE <service-name | port>

Specifies the service name or port number to use to listen for client connections.

The -SERVICE option values that are used to start the spawner determine what is used by the client to sign on.

Note If the -SERVICE option is not specified, the spawner listens on Telnet port (23).

Example In this example, the following option is enclosed in double quotation marks and added after USERMODS= in `/opt/sas/viya/config/etc/connect/default/connect_usermods.sh`. When the spawner starts, it uses port 5020 for the -SERVICE option during spawner start-up:

```
USERMODS="-service 5020"
```

The client can then sign on by specifying the explicit port-number in the SIGNON statement:

```
%let myHost=<spawner-host> 5020;
signon myHost user='myuserid' password='mypassword';
```

-SHELL

Specifies that the started SAS/CONNECT servers allow X commands.

Without specifying the -SHELL option to the spawner, X command processing is disabled by default.

-SSPI

Identifies support for the Security Support Provider Interface for single sign-on connections to the spawner. To enable SSPI authentication, you must specify -SSPI in the spawner start-up command.

Default `-NOSSPI`

-TRACE | -VERBOSE

Turns on trace level output.

Spawner Security Options

SAS/CONNECT Spawner uses the [SAS System Options for encryption on page 418](#) .

TCPPORTFIRST System Option

TCPPORTFIRST=<port-number>

TCPPORTLAST=<port-number>

Restricts the range of TCP/IP ports that clients can use to remotely access servers. Within the range of 0 through 32767, assign a beginning value to TCPPORTFIRST and an ending value to TCPPORTLAST. To restrict the range of ports to only one port, set the values for TCPPORTFIRST and TCPPORTLAST to the same number. Consult with your network administrator for advice about these settings.

When `-NOINHERITANCE` is on, you can set TCPPORTFIRST and TCPPORTLAST in a SAS start-up command or in the configuration file.

This applies in the `noinheritance` case only. When socket inheritance is enabled, the child SAS/CONNECT server does not start up as a listening port.

Server	Optional
Range	0–32767
Example	In the example below, the server is restricted to the TCP/IP ports 4020 through 4050: <pre>options tcpportfirst=4020; options tcpportlast=4050;</pre>

Server Configuration Files

Configuration Home Directory

The SAS Viya deployment process creates a configuration home directory for each server instance.

Here is an example:

```
/opt/sas/viya/config/etc/spawner/default
```

The final directory in the path, `default`, is the deployment instance for the server.

Server Configuration Files

Each of the following SAS Viya programming run-time servers uses one or more server configuration files, as appropriate.

- SAS Compute Server
- SAS Workspace Server
- SAS Object Spawner
- SAS/CONNECT Server

■ SAS/CONNECT Spawner

Table A.52 Server Configuration Files

Standard Filename	Description
autoexec.sas	<p>Contains SAS statements that are executed immediately after SAS initializes all components of the SAS Application Server.</p> <p>Do not modify this file. If you need to make changes, modify the <code>appserver_autoexec_usermods.sas</code> file that is in the same directory.</p>
autoexec_deployment.sas	<p>Contains server configuration settings that are created during deployment by Ansible from <code>vars.yml</code>. During updates, user configuration settings are overwritten.</p> <p>Do not modify this file. If you need to make changes, modify the <code>sasv9_usermods.cfg</code> file that is in the same directory.</p>
autoexec_usermods.sas	<p>Contains modifications made by the SAS administrator. Using <code>autoexec_usermods.sas</code> ensures that your modifications are not overwritten when you update SAS Viya.</p>
sasv9.cfg	<p>Specifies start-up options for the server and contains calls to other files that are listed in this table.</p> <p>Do not modify this file. If you need to make changes, modify the <code>sasv9_usermods.cfg</code> file that is in the same directory.</p>
sasv9_deployment.cfg	<p>Specifies start-up options for the server and contains calls to other files that are listed in this table that are created during deployment by Ansible from <code>vars.yml</code>.</p> <p>Do not modify this file. If you need to make changes, modify the <code>sasv9_usermods.cfg</code> file that is in the same directory.</p>
sasv9_usermods.cfg	<p>Contains modifications made by the SAS administrator. Using <code>sasv9_usermods.cfg</code> ensures that your modifications are not overwritten when you update SAS Viya.</p>
logconfig.xml	<p>Specifies the logging configuration for the server or the spawner.</p>
logconfig.trace.xml	<p>Contains alternative logging configuration settings for high-level logging messages (for example, DEBUG and TRACE messages) that can be used by SAS Technical Support to help resolve server issues. The messages are written to the server or spawner rolling log file.</p>
sasenv_deployment	<p>Contains server environmental variable settings that are created during deployment by Ansible from <code>vars.yml</code>. During updates, user configuration settings are overwritten.</p> <p>Do not modify this file. Add local environmental variable settings in the <code>sasenv_local</code> file in the <code>/opt/sas/viya/config/etc/server/default</code> directory.</p>

Standard Filename	Description
<ul style="list-style-type: none">■ connect.sh■ connectserver.sh■ spawner.sh■ workspaceserver.sh	<p>Invoke sas with the default configuration for this SAS Application Server. It changes directories so that the SAS run-time environment is invoked from the root directory of this application server.</p> <p>Do not modify these files. If you need to make changes, modify the <i>server-spawner_usermods.sh</i> file that is in the same directory.</p>
<ul style="list-style-type: none">■ connect_usermods.sh■ connectserver_usermods.sh■ spawner_usermods.sh■ workspaceserver_usermods.sh	<p>Contain modifications made by the SAS administrator to the configurations for these application servers. Using <i>server-spawner_usermods.sh</i> ensures that your modifications are not overwritten when you update SAS Viya.</p>

Infrastructure Servers

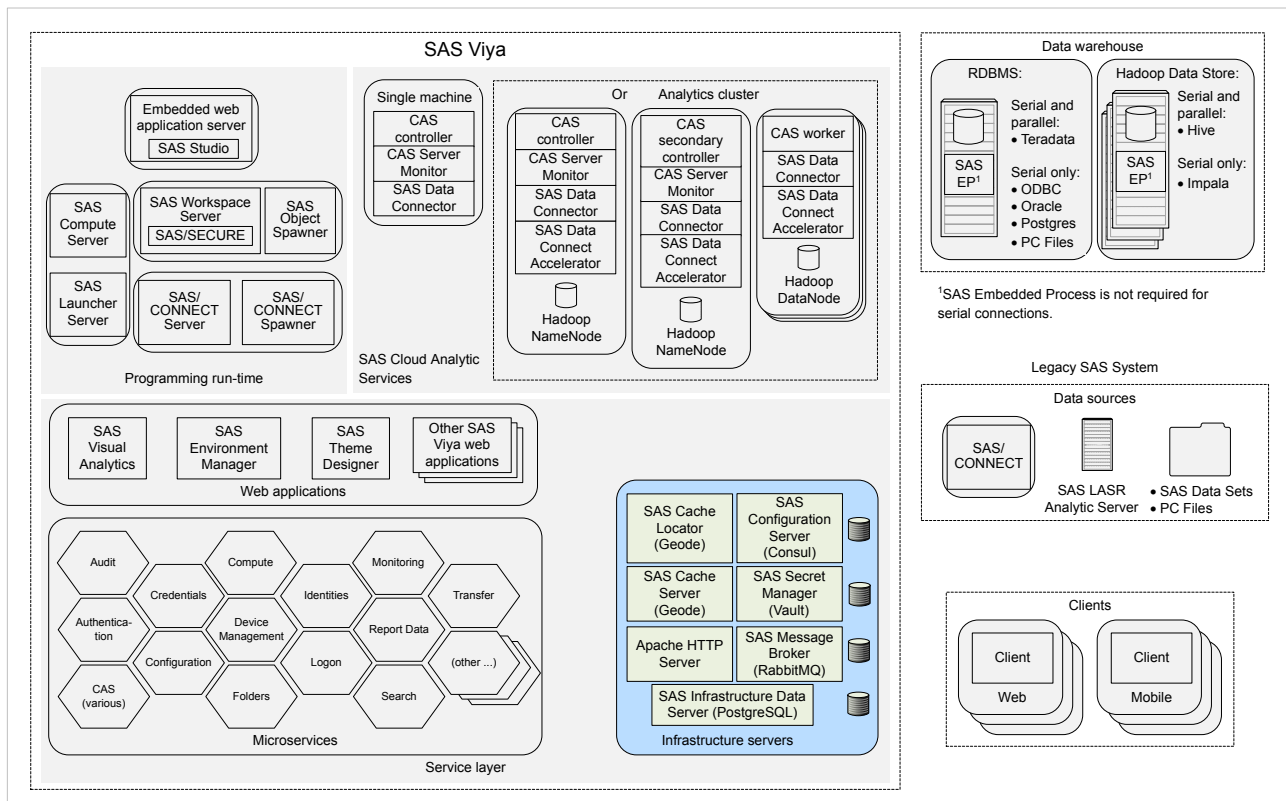
Infrastructure Servers: Overview

Note: A [programming-only on page 15](#) deployment uses only one of the infrastructure servers—Apache HTTP Server.

SAS Viya contains these infrastructure servers:

- “SAS Configuration Server”
- “SAS Secret Manager”
- “SAS Infrastructure Data Server”
- “SAS Message Broker”
- “SAS Cache Locator and Cache Server”
- “Apache HTTP Server”

Figure A.1 SAS Viya Infrastructure Servers



SAS Configuration Server

Overview

SAS Configuration Server is based on HashiCorp Consul 0.7.5. SAS Configuration Server uses Consul as a service configuration registry that serves as a central repository for configuration data, service discovery, and health status.

Note: A [programming-only deployment on page 1](#) does not use SAS Configuration Server.

How To

Operate

SAS Viya uses the operating system's default init system or systemd command to launch a script that can stop, start, restart, and check the status of SAS Configuration Server, which is based on Consul. This script, `sas-viya-consul-default`, resides in `/etc/init.d`.

Note: You must be signed in to the machine where the configuration server resides, and you must have sudo privileges to run this script.

To operate SAS Configuration Server, run the following command, as appropriate:

```
sas-viya-consul-default status | stop | start | restart
```

Note: For multi-machine deployments, run `sas-viya-consul-default` on every SAS Viya machine. Start or restart SAS Configuration Server *first*. Stop SAS Configuration Server *last*.

Note: You can use a script to manage and view the running state of all SAS Viya services. For more information, see [“All Servers and Services” on page 599](#).

Here are a few examples of how to operate this script:

- To check status of SAS Configuration Server using a direct call:

```
sudo /etc/init.d/sas-viya-consul-default status
```

- To stop SAS Configuration Server using the Red Hat Linux version 6 init system command:

```
sudo service sas-viya-consul-default stop
```

- To start SAS Configuration Server using the Red Hat Linux version 7 systemd command:

```
sudo systemctl start sas-viya-consul-default
```

- To restart SAS Configuration Server using a direct call:

```
sudo /etc/init.d/sas-viya-consul-default restart
```

Locate Logs

SAS Configuration Server log files are located in `/opt/sas/viya/config/var/log/consul/default`.

Concepts

What Is SAS Configuration Server?

SAS Configuration Server is based on HashiCorp's Consul. Consul is a distributed, highly available registry that contains service configuration data and availability and overall performance (health) information.

Configuration data resides in SAS Configuration Server as key-value pairs. This data is used by SAS Viya microservices at start-up to load default values and to discover any service dependencies.

During run time, whenever a service's properties change, the service is notified, and it rereads its properties from SAS Configuration Server. (The exceptions are noted in [“What Services Must Be Restarted?”](#).)

Each service registers its health checks when it starts. The Monitoring system periodically queries the status of the health checks.

How Does the SAS Configuration Service Work with SAS Configuration Server?

For information about how the SAS Configuration Service works with SAS Configuration Server, see [“How SAS Viya Configuration Works”](#) on page 220.

SAS Secret Manager

Overview

SAS Secret Manager is based on HashiCorp Vault 0.6.4. SAS Secret Manager uses Vault to store and generate secrets such as Transport Layer Security (TLS) certificates.

Note: A [programming-only deployment on page 1](#) does not use SAS Secret Manager.

How To

Operate

SAS Viya uses the operating system's default init system or systemd command to launch a script that can stop, start, restart, and check the status of SAS Secret Manager, which is based on Vault. This script, `sas-viya-vault-default`, resides in `/etc/init.d`.

Note: You must be signed in to the machine where configuration server resides, and you must have sudo privileges to run this script.

To operate SAS Secret Manager, run the following command, as appropriate:

```
sas-viya-vault-default status | stop | start | restart
```

Note: For multi-machine deployments, run `sas-viya-vault-default` on every SAS Viya machine. Start or restart SAS Secret Manager immediately *after* you run [SAS Configuration Server](#). Stop SAS Secret Manager immediately *before* you stop SAS Configuration Server.

Note: There is a script with which you can manage and view the running state of all SAS Viya services. For more information, see [“All Servers and Services”](#) on page 599.

Here are a few examples of how to operate this script:

- To check status of SAS Secret Manager using a direct call:

```
sudo /etc/init.d/sas-viya-vault-default status
```

- To stop SAS Secret Manager using the Red Hat Linux version 6 init system command:

```
sudo service sas-viya-vault-default stop
```

- To start SAS Secret Manager using the Red Hat Linux version 7 systemd command:

```
sudo systemctl start sas-viya-vault-default
```

- To restart SAS Secret Manager using a direct call:

```
sudo /etc/init.d/sas-viya-vault-default restart
```

Locate Logs

SAS Secret Manager log files are located in `/opt/sas/viya/config/var/log/vault/default`.

Concepts

What Is SAS Secret Manager?

SAS Secret Manager is based on HashiCorp Vault. Vault is a distributed, highly available server used to manage secrets. A *secret* is information that you want to secure, such as keys, passwords, certificates, and so on. Vault provides a secure interface to secrets, in addition to access control, and audit logging.

Here are some features of secret manager and examples of how SAS Viya uses them:

- On-demand generation of secrets
Secret manager generates TLS certificates for SAS Viya servers at startup.
- Secure storage for secrets
Microservices use secure storage so that multiple microservice instances running on the same machine do not request multiple TLS certificates.
- Encrypt and decrypt data without storing it
SAS Compute Server uses this feature when it sends a password to child processes.
- Revocation of secrets
SAS Viya services use this feature when rotating security artifacts. (For example, services use vault tokens to request TLS certificates).

For more information, see [“Concepts” on page 407](#).

Dependency on SAS Configuration Server (Consul)

SAS Secret Manager is installed on the same machines where SAS Configuration Server (Consul) resides. SAS Configuration Server contains a namespace where secret manager stores secrets in encrypted form, which enables all instances of secret manager access to consistent data. Also, secret manager relies on the configuration server for locking and leader election. Therefore, configuration server must be running in order for SAS Secret Manager to be operational.

For information about SAS Secret Manager topology, see [“Fault Tolerance in SAS Viya” on page 602](#).

SAS Infrastructure Data Server

Overview

SAS Infrastructure Data Server is based on PostgreSQL version 9 and is configured specifically to support SAS software. SAS Infrastructure Data Server stores user content, such as reports, custom groups, comments, authorization rules, selected source definitions, attachments, audit records, and user preferences.

Note: A [programming-only deployment on page 1](#) does not use SAS Infrastructure Data Server.

How To (Cluster)

Operate a Cluster

SAS Viya uses the operating system's default init system or systemd command to launch a script that can stop, start, restart, and check the status of the SAS Infrastructure Data Server cluster. (A data server cluster consists of all the PostgreSQL data nodes and Pgpool-II.) The script, `sas-viya-sasdatasvrc-postgres`, resides in `/etc/init.d`.

Note: You must be signed in to the machine where the pgpool server resides, and you must have sudo privileges to run the script.

To operate a data server cluster, run the following command, as appropriate:

```
sas-viya-sasdatasvrc-postgres status | stop | start | restart
```

Note: You can use a script to manage and view the running state of all SAS Viya servers and services. For more information, see [“All Servers and Services” on page 599](#).

Here are a few examples of how to operate this script:

- To check status of the data server cluster using a direct call:

```
sudo /etc/init.d/sas-viya-sasdatasvrc-postgres status
```

- To stop the data server cluster using the Red Hat Linux version 6 init system command:

```
sudo service sas-viya-sasdatasvrc-postgres stop
```

- To start the data server cluster using the Red Hat Linux version 7 systemd command:

```
sudo systemctl start sas-viya-sasdatasvrc-postgres
```

- To restart the data server cluster using a direct call:

```
sudo /etc/init.d/sas-viya-sasdatasvrc-postgres restart
```

Recover a Failed Cluster

The SAS Infrastructure Data Server cluster is considered to be failed under these conditions: when it fails to start, and when all its data nodes are marked as `unhealthy` in the cluster definition file `/opt/sas/viya/config/etc/sasdatasvrc/postgres/pgpool10/pool.cdf`.

Common causes for cluster failure include a power failure, network connectivity issues, or a machine reboot. Another cause of cluster failure can be from lack of system resources. Examples are disk space, memory, number of processes, number of open files, ports, semaphores, and shared memory. In such cases, the data server [logs](#) contain failure information.

The cluster will not start until the problem that caused the cluster failure has been fixed, and the server nodes are marked as `healthy` in `pool.cdf`.

On the pgpool server machine, you can manually update `pool.cdf`, or you can run the `repair_postgres_nodes.sh` script. After the script updates `pool.cdf`, it attempts to start the cluster.

- 1 Before attempting any cluster repair procedures, do the following:
 - Examine the integrity of the data on the data server.
 - One or more failovers might have occurred. Therefore, examine the server with the most current data.
 - Back up the data server data directories.
- 2 Fix the problem that caused the cluster to fail.
- 3 Make sure that the following servers are running and are accessible:
 - [SAS Configuration Server \(Consul\)](#)
 - [SAS Secret Manager \(Vault\)](#)

4 As the SAS install user (`sas`) or with `sudo` privileges, sign in to the pgpool server machine.

5 Choose one of the following methods to update `pool.cdf`:

Note: Each method attempts to restart the cluster after `pool.cdf` has been modified.

- Run the `repair_postgres_nodes.sh` script to mark all cluster nodes as healthy in `pool.cdf`:

```
sudo /opt/sas/viya/home/libexec/sasdatasvrc/script/maintenance/
repair_postgres_nodes.sh
```

- Using a text editor, open the cluster definition file and mark all of the data server nodes as healthy:

```
vi /opt/sas/viya/config/etc/sasdatasvrc/postgres/pgpool0/pool.cdf
```

Change `node0=unhealthy` to `node0=healthy`.

6 [Check the status of the data server cluster.](#)

A **status** of `up` in the cluster status list indicates that the node is connected and is an active part of the cluster. There should be only one primary server, with no standby servers or one or more standby servers, as appropriate.

Failback a Cluster

Failback refers to the restoration of the high availability (HA) SAS Infrastructure Data Server cluster to its original configuration before the failover.

A PostgreSQL *original configuration* is indicated when the cluster status list displays the following:

- `node0` has the **role** of `primary`
- all other nodes have a **role** of `standby`
- all nodes have a **status** of `up`

To failback a SAS Infrastructure Data Server cluster to its original configuration before the failover, follow these steps:

- 1 Make sure that the following servers are running and are accessible:
 - [SAS Configuration Server \(Consul\)](#)
 - [Pgpool server](#)
 - [SAS Infrastructure Data Server \(PostgreSQL\) cluster](#)

- 2 As the SAS install user (sas) or with sudo privileges, sign in to the pgpool server machine.
- 3 To ensure that the cluster is in a failover condition, run the following command:

```
sudo service sas-viya-sasdatasvrc-service-name status
```

Verify that node0 has a **status** of down and a **role** of standby.

Here is an example where the *service-name* is named postgres2:

```
sudo service sas-viya-sasdatasvrc-postgres2 status
```

Here is typical output:

```
Checking status of sas-viya-sasdatasvrc-postgres2...

PGPool is running with PID=11445
Checking Postgresql nodes status...
 node_id | hostname | port | status | lb_weight | role | select_cnt | load_balance_node | replication_delay
-----+-----+-----+-----+-----+-----+-----+-----+-----
  0      | machine1 | 5452 | down   | 0.250000 | standby | 1           | false              | 0
  1      | machine2 | 5452 | up     | 0.250000 | primary | 0           | true               | 0
  2      | machine3 | 5452 | up     | 0.250000 | standby | 0           | false              | 0
  3      | machine4 | 5452 | up     | 0.250000 | standby | 0           | false              | 0
(4 rows)
```

- 4 In the cluster status list, if node0 has a **status** of up and a **role** of standby, you can go directly to [Step 7](#).
- 5 To recover (start) node0, run the following command:

```
sudo service sas-viya-sasdatasvrc-service-name-node0 start
```

Here is an example:

```
sudo service sas-viya-sasdatasvrc-postgres2-node0 start
```

Here is typical output:

```
Starting sas-viya-sasdatasvrc-postgres2-node0 service...

[ OK ]
```

- 6 Run the following command:

```
sudo service sas-viya-sasdatasvrc-service-name status
```

Verify that node0 has a **status** of up.

Here is an example:

```
sudo service sas-viya-sasdatasvrc-postgres2 status
```

Here is typical output:

```
Checking status of sas-viya-sasdatasvrc-postgres2...

PGPool is running with PID=11445
Checking Postgresql nodes status...
 node_id | hostname | port | status | lb_weight | role | select_cnt | load_balance_node | replication_delay
-----+-----+-----+-----+-----+-----+-----+-----+-----
  0      | machine1 | 5452 | up     | 0.250000 | standby | 1           | false              | 0
  1      | machine2 | 5452 | up     | 0.250000 | primary | 0           | true               | 0
  2      | machine3 | 5452 | up     | 0.250000 | standby | 0           | false              | 0
  3      | machine4 | 5452 | up     | 0.250000 | standby | 0           | false              | 0
(4 rows)
```

- 7 To stop the primary node of the cluster, run the following command:

```
sudo service sas-viya-sasdatasvrc-service-name-node-name stop
```

Here is an example where *node-name* is named *node1*:

```
sudo service sas-viya-sasdatasvrc-postgres2-node1 stop
```

Here is typical output:

```
Stopping sas-viya-sasdatasvrc-postgres2-node1 service...
[ OK ]
```

- 8 Run the following command:

```
sudo service sas-viya-sasdatasvrc-service-name status
```

Verify that *node0* has a **status** of up and a **role** of primary.

Here is an example where *service-name* is named *postgres2*:

```
sudo service sas-viya-sasdatasvrc-postgres2 status
```

Here is typical output:

```
Checking status of sas-viya-sasdatasvrc-postgres2...

PGPool is running with PID=11445
Checking Postgresql nodes status...
node_id | hostname | port | status | lb_weight | role | select_cnt | load_balance_node | replication_delay
-----+-----+-----+-----+-----+-----+-----+-----+-----
0       | machine1 | 5452 | up     | 0.250000 | primary | 1           | false              | 0
1       | machine2 | 5452 | down   | 0.250000 | standby | 0           | true               | 0
2       | machine3 | 5452 | up     | 0.250000 | standby | 0           | false              | 0
3       | machine4 | 5452 | up     | 0.250000 | standby | 0           | false              | 0
(4 rows)
```

Note: The remaining running nodes of the cluster initially show a **status** of down while replication is established for the new primary node. Continue to monitor the cluster status until all running nodes have a **status** of up.

- 9 To recover (start) the previous primary node, run the following command:

```
sudo service sas-viya-sasdatasvrc-service-name-node-name start
```

Here is an example where the previous primary node is named *node1*:

```
sudo service sas-viya-sasdatasvrc-postgres2-node1 start
```

Here is typical output:

```
Starting sas-viya-sasdatasvrc-postgres2-node1 service...
[ OK ]
```

- 10 Run the following command:

```
sudo service sas-viya-sasdatasvrc-service-name status
```

Verify that the cluster has returned to its initial configuration:

- *node0* has the **role** of primary
- all other nodes have a **role** of standby

- all nodes have a **status** of up

Here is an example where the data server service is named `postgres2`:

```
sudo service sas-viya-sasdatasvrc-postgres2 status
```

Here is typical output:

```
Checking status of sas-viya-sasdatasvrc-postgres2...

PGPool is running with PID=11445
Checking Postgresql nodes status...
node_id | hostname | port | status | lb_weight | role | select_cnt | load_balance_node | replication_delay
-----+-----+-----+-----+-----+-----+-----+-----+-----
0       | machine1 | 5452 | up     | 0.250000 | primary | 1          | false              | 0
1       | machine2 | 5452 | up     | 0.250000 | standby | 0          | true               | 0
2       | machine3 | 5452 | up     | 0.250000 | standby | 0          | false              | 0
3       | machine4 | 5452 | up     | 0.250000 | standby | 0          | false              | 0
(4 rows)
```

Add a Cluster (Ansible)

- 1 Sign on your Ansible controller with administrator privileges, and locate the file, `/playbook/vars.yml`.
- 2 Using a text editor, open `vars.yml` and locate the `INVOCATION_VARIABLES` section.

```
INVOCATION_VARIABLES:
  Machine1:
    pgpoolc:
      - PCP_PORT: '5430'
      PGPOOL_PORT: '5431'
      SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
      SERVICE_NAME: postgres
    sasdatasvrc:
      - NODE_NUMBER: '0'
      NODE_TYPE: P
      PG_PORT: '5432'
      SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
      SERVICE_NAME: postgres
```

- 3 Copy and paste an existing cluster definition.

In this example, the new cluster is being added to Machine2:

```
INVOCATION_VARIABLES:
  Machine1:
    pgpoolc:
      - PCP_PORT: '5430'
      PGPOOL_PORT: '5431'
      SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
      SERVICE_NAME: postgres
    sasdatasvrc:
      - NODE_NUMBER: '0'
      NODE_TYPE: P
      PG_PORT: '5432'
      SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
      SERVICE_NAME: postgres
```

```
Machine2:
```

```
pgpoolc:
```

```

- PCP_PORT: '5430'
  PGPOOL_PORT: '5431'
  SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
  SERVICE_NAME: postgres
  sasdatasvrc:
- NODE_NUMBER: '0'
  NODE_TYPE: P
  PG_PORT: '5432'
  SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
  SERVICE_NAME: postgres

```

4 Configure the new cluster definition for the pgpool server (pgpoolc) and the data server nodes (sasdatasvrc):

- Pgpool server definition parameters:

- PCP_PORT

Specifies the `pcp` port for the pgpool instance.

- PGPOOL_PORT

Specifies the pgpool port. This is the primary port through which all databases connect.

- SANMOUNT

Specifies the location of the data files. This path is typically the same value as the other data nodes.

- SERVICE_NAME

Specifies the unique service name for the data server cluster. `SERVICE_NAME` should be the same for the pgpool server and all nodes in the cluster.

- Data server node definition parameters:

- NODE_NUMBER

Specifies the sequential node identifier. The primary node is 0. Standby nodes start at 1 and are incremented sequentially.

- NODE_TYPE

Specifies the type of node that you are adding. The primary node should have a value of `P`. Standby nodes should have a value of `S`.

- PG_PORT

Specifies the Postgres database port. The pgpool server communicates with the database on this port. Clients use the `PGPOOL_PORT`. The port must be available for use on the deploy target.

- SANMOUNT

Specifies the location of the data files. This path is typically the same value as the other data nodes.

- SERVICE_NAME

Specifies the unique service name for the data server cluster. `SERVICE_NAME` should be the same for the pgpool server and all the nodes in the cluster.

Here is an example:

```

INVOCATION_VARIABLES:
  Machine2:
    pgpoolc:
      - PCP_PORT: '5430'
        PGPOOL_PORT: '5431'
        SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
        SERVICE_NAME: postgres2

```



```

sasdatasvrc:
- NODE_NUMBER: '0'
  NODE_TYPE: P
  PG_PORT: '5432'
  SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
  SERVICE_NAME: postgres2
- NODE_NUMBER: '1'
  NODE_TYPE: S
  PG_PORT: '5432'
  SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
  SERVICE_NAME: postgres2

```

- 5 Run your Ansible playbook using the `sitedefault.yml` file.

Here is an example:

```
ansible-playbook site.yml
```

For a complete list of playbook commands, see [“Commands” in SAS Viya for Linux: Deployment Guide](#).

Delete a Node or a Cluster

CAUTION! Do not delete a primary node unless you plan to delete the entire cluster. Doing so would increase chances of introducing data corruption. To delete the primary node, failover the node to a standby node, and wait for all remaining nodes to indicate that they are available. When the nodes are available, it is safe to delete the former primary node. Do not delete a pgpool node without first moving the pgpool node of the cluster. Failure to do so will make the cluster unusable. If you choose to delete the node data using the `-d` option, its data files are deleted. Use caution when deciding to use the `-d` option.

- 1 As root or with an account that has sudo privileges, sign in to the machine where the node that you want to remove resides.
- 2 Change the directory to `/opt/sas/viya/home/libexec/sasdatasvrc/script`.
- 3 Run the `sds_delete_node.sh` script with the following options:

Note: When the `sds_delete_node.sh` script runs, it stops the cluster.

- `-s service-name`
- `-n cluster-name | node-name`
- `-d y | n`

CAUTION! A yes (y) value specifies that the script deletes the node or the cluster data files.

- `-c absolute-path/sds_env_var.sh`

Here is an example:

```

sudo ./sds_delete_node.sh -s postgres -n node1 -d y
-c /opt/sas/viya/config/etc/sasdatasvrc/postgres/node1/sds_env_var.sh

```

Every time the script runs, it generates a new log file in `/tmp/sds_uninstall_log`.

- 4 After the script runs, be sure to delete the node or the cluster definition in the `INVOCATION_VARIABLES` section of `vars.yml`. For more information, see [“Add a Node \(Ansible\)” on page 692](#).

How To (Nodes)

Check the Status of a Node

SAS Viya uses the operating system's default init system or `systemd` command to launch a script that can check the status of a SAS Infrastructure Data Server node. This script, `sas-viya-sasdatasvrc-postgres-noden`, resides in `/etc/init.d`.

Note: As the SAS install user (`sas`) or with `sudo` privileges, you must be signed in to the machine where the node resides.

Note: Each node script is numbered, starting at zero (0).

Run the script:

- Red Hat Linux version 6:

```
sudo service sas-viya-sasdatasvrc-postgres-noden status
```

- Red Hat Linux version 7:

```
sudo systemctl status sas-viya-sasdatasvrc-postgres-noden
```

Note: On Red Hat Linux version 7, the first time you run the `sas-viya-sasdatasvrc-postgres-noden` command, you must run the command twice. Red Hat Linux version 7 has backward compatibility for the system V service, and it does not have initial `systemd` unit files. The first time the `sas-viya-sasdatasvrc-postgres-noden` command runs, it builds the service unit file from the system V init file. Therefore, until the unit file is built, the `sas-viya-sasdatasvrc-postgres-noden` commands do not function properly.

Stop a Node

SAS Viya uses the operating system's default init system or `systemd` command to launch a script that can stop a SAS Infrastructure Data Server node. This script, `sas-viya-sasdatasvrc-postgres-noden`, resides in `/etc/init.d`.

CAUTION! The act of stopping individual nodes changes the cluster state. Stopping the primary node causes a failover to occur in a cluster of two or more nodes. In addition, stopping a standby node removes the node from the active cluster. Stopped nodes must be recovered in order for them to be added back to the cluster. The recovery of stopped nodes occurs automatically during node start-up. During failover of the primary node (0), other healthy standby nodes (2 and 3) go through a process of "following" the new primary node (1). During failover, nodes 2 and 3 briefly detach from the cluster and display a status of 3 (unhealthy). Wait several minutes and then recheck the cluster status. Eventually, nodes 2 and 3 should re-attach to the cluster and display a **status** of `up` (healthy).

Note: As the SAS install user (`sas`) or with `sudo` privileges, you must be signed in to the machine where the node resides.

Note: Each node script is numbered, starting at zero (0).

Note: On Red Hat Linux version 7, the first time you run the `sas-viya-sasdatasvrc-postgres-noden` command, you must run the command twice. Red Hat Linux version 7 has backward compatibility for the system V service, and it does not have initial `systemd` unit files. The first time the `sas-viya-sasdatasvrc-postgres-noden` command runs, it builds the service unit file from the system V init file. Therefore, until the unit file is built, the `sas-viya-sasdatasvrc-postgres-noden` commands do not function properly.

Run the script:

- Red Hat Linux version 6:

```
sudo service sas-viya-sasdatasvrc-postgres-noden stop
```

- Red Hat Linux version 7:

```
sudo systemctl stop sas-viya-sasdatasvrc-postgres-noden
```

Start a Node (Recover a Node)

A SAS Infrastructure Data Server data node is considered to be “unhealthy” when it has a **status** of `down` in the cluster status list. If a PostgreSQL data node has been stopped or has been taken offline, the pgpool server removes this node from the cluster.

When you restart an unhealthy node, pgpool server automatically initiates the node recovery process. To recover an unhealthy data node, follow these steps:

- 1 Make sure that the following servers are running and accessible:
 - [SAS Configuration Server \(Consul\)](#)
 - [pgpool server](#)
 - [SAS Infrastructure Data Server \(PostgreSQL\) cluster](#)
- 2 As the SAS install user (`sas`) or with `sudo` privileges, sign in to the pgpool server machine, and run the following command:

Note: On Red Hat Linux version 7, the first time you run the `sas-viya-sasdatasvrc-postgres-noden` command, you must run the command twice. Red Hat Linux version 7 has backward compatibility for the system V service, and it does not have initial systemd unit files. The first time the `sas-viya-sasdatasvrc-postgres-noden` command runs, it builds the service unit file from the system V init file. Therefore, until the unit file is built, the `sas-viya-sasdatasvrc-postgres-noden` commands do not function properly.

```
sudo service sas-viya-sasdatasvrc-service-name status
```

Verify that the unhealthy data node has a **status** of `down` and a **role** of `standby`.

Here is an example where the data server service is named `postgres2`:

```
sudo service sas-viya-sasdatasvrc-postgres2 status
```

Here is typical output:

Note: In this example, the unhealthy node is `node0`.

```
Checking status of sas-viya-sasdatasvrc-postgres2...

PGPool is running with PID=11445
Checking Postgresql nodes status...
 node_id | hostname | port | status | lb_weight | role | select_cnt | load_balance_node | replication_delay
-----+-----+-----+-----+-----+-----+-----+-----+-----
  0      | machine1 | 5452 | down   | 0.250000 | standby | 1           | false              | 0
  1      | machine2 | 5452 | up     | 0.250000 | primary | 0           | true               | 0
  2      | machine3 | 5452 | up     | 0.250000 | standby | 0           | false              | 0
  3      | machine4 | 5452 | up     | 0.250000 | standby | 0           | false              | 0
(4 rows)
```

- 3 As the SAS install user (`sas`) or with `sudo` privileges, sign in to the machine that contains the unhealthy data node.
- 4 Make sure that the unhealthy node is stopped by running the following command:

```
sudo service sas-viya-sasdatasvrc-service-name-node-name stop
```

Here is an example where the unhealthy node is named `node0`.

```
sudo service sas-viya-sasdatasvrc-postgres2-node0 stop
```

Here is typical output:

```
Service sas-viya-sasdatasvrc-postgres2-node0 is not running.
[ OK ]
```

- 5 Recover the node as a standby server by running the following command:

```
sudo service sas-viya-sasdatasvrc-service-name-node-name start
```

The pgpool server automatically starts the unhealthy node.

A node **status** of `up` indicates that the node is connected and is an active part of the cluster. There should be only one server with a **role** of `primary`, with zero or more servers with a **role** of `standby`.

Here is an example:

```
sudo service sas-viya-sasdatasvrc-postgres2-node0 start
```

Here is typical output:

```
Starting sas-viya-sasdatasvrc-postgres2-node0 service...
[ OK ]
```

- 6 Make sure that the node has been successfully added to the cluster by running the following command:

```
sudo service sas-viya-sasdatasvrc-service-name status
```

Here is an example:

```
sudo service sas-viya-sasdatasvrc-postgres2 status
```

Here is typical output:

Note: In this example, the previously unhealthy node (`node0`) has a **status** of `up`.

```
Checking status of sas-viya-sasdatasvrc-postgres2...

PGPool is running with PID=11445
Checking Postgresql nodes status...
node_id | hostname | port | status | lb_weight | role | select_cnt | load_balance_node | replication_delay
-----+-----+-----+-----+-----+-----+-----+-----+-----
0 | machine1 | 5452 | up | 0.250000 | standby | 1 | false | 0
1 | machine2 | 5452 | up | 0.250000 | primary | 0 | true | 0
2 | machine3 | 5452 | up | 0.250000 | standby | 0 | false | 0
3 | machine4 | 5452 | up | 0.250000 | standby | 0 | false | 0
(4 rows)
```

Note: If starting (recovering) a node fails, refer to [the Troubleshooting section on page 705](#).

Add a Node (Ansible)

Adding a data node to your SAS Infrastructure Data Server cluster consists of modifying the `vars.yml` file and running your Ansible playbook.

- 1 With administrator privileges, sign in to your Ansible controller, and locate the file, `/playbook/vars.yml`.
- 2 Using a text editor, open `vars.yml` and locate the `INVOCATION_VARIABLES` section.

```
INVOCATION_VARIABLES:
  Machine1:
    pgpoolc:
      - PCP_PORT: '5430'
        PGPOOL_PORT: '5431'
```

```

SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
SERVICE_NAME: postgres
sasdatasvrc:
- NODE_NUMBER: '0'
  NODE_TYPE: P
  PG_PORT: '5432'
  SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
  SERVICE_NAME: postgres

```

- 3 Copy an existing node definition and place it under the deploy target on which the node will be configured.

Here is an example:

```

INVOCATION_VARIABLES:
Machine1:
  pgpoolc:
    - PCP_PORT: '5430'
      PGPOOL_PORT: '5431'
      SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
      SERVICE_NAME: postgres
  sasdatasvrc:
    - NODE_NUMBER: '0'
      NODE_TYPE: P
      PG_PORT: '5432'
      SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
      SERVICE_NAME: postgres
    - NODE_NUMBER: '0'
      NODE_TYPE: P
      PG_PORT: '5432'
      SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
      SERVICE_NAME: postgres

```

- 4 Configure the node definition in order to meet the requirements of the cluster:

- **NODE_NUMBER**

Specifies the sequential node identifier. Standby nodes start at 1 and are incremented sequentially.

For example, if you have only a primary node, the node that you are adding should have a `NODE_NUMBER` of 1. If the last standby node in your cluster has the value of 1, the node that you are adding should have a `NODE_NUMBER` of 2.

- **NODE_TYPE**

Specifies the type of node that you are adding. The only acceptable value is `s` (standby). After initial deployment, you cannot add a primary node.

- **PG_PORT**

Specifies the Postgres database port. The pgpool server communicates with the database on this port. Clients use the `PGPOOL_PORT`. The port must be available for use on the deploy target.

- **SANMOUNT**

Specifies the location of the data files. This path is typically the same value as the other data nodes.

- **SERVICE_NAME**

Specifies the service name for the data server cluster. It must be an exact match of the name of the cluster to which you are adding a data node.

Here is an example:

```

INVOCATION_VARIABLES:

```

```

Machine1:
  pgpoolc:
    - PCP_PORT: '5430'
      PGPOOL_PORT: '5431'
      SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
      SERVICE_NAME: postgres
  sasdatasvrc:
    - NODE_NUMBER: '0'
      NODE_TYPE: P
      PG_PORT: '5432'
      SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
      SERVICE_NAME: postgres
    - NODE_NUMBER: '1'
      NODE_TYPE: S
      PG_PORT: '5432'
      SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
      SERVICE_NAME: postgres

```

- 5 Run your Ansible playbook using the `sitedefault.yml` file.

Here is an example:

```
ansible-playbook site.yml
```

For a complete list of playbook commands, see [“Commands” in SAS Viya for Linux: Deployment Guide](#).

Move a Node (Ansible)

Moving a data node to your SAS Infrastructure Data Server cluster consists of modifying the `vars.yml` file and running your Ansible playbook.

- 1 With administrator privileges, sign in to your Ansible controller, and locate the file `/playbook/vars.yml`.
- 2 Using a text editor, open `vars.yml` and locate the `INVOCATION_VARIABLES` section.

```

INVOCATION_VARIABLES:
  Machine1:
    pgpoolc:
      - PCP_PORT: '5430'
        PGPOOL_PORT: '5431'
        SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
        SERVICE_NAME: postgres
    sasdatasvrc:
      - NODE_NUMBER: '0'
        NODE_TYPE: P
        PG_PORT: '5432'
        SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
        SERVICE_NAME: postgres
      - NODE_NUMBER: '1'
        NODE_TYPE: S
        PG_PORT: '5432'
        SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
        SERVICE_NAME: postgres

```

- 3 Copy the existing node definition and place it under the deploy target to which you want to move the node.

In this example, the deploy target is `Machine2`:

```

INVOCATION_VARIABLES:
  Machine2:

```

```

pgpoolc:
- PCP_PORT: '5430'
  PGPOOL_PORT: '5431'
  SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
  SERVICE_NAME: postgres2
sasdatasvrc:
- NODE_NUMBER: '0'
  NODE_TYPE: P
  PG_PORT: '5432'
  SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
  SERVICE_NAME: postgres2
- NODE_NUMBER: '1'
  NODE_TYPE: S
  PG_PORT: '5432'
  SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
  SERVICE_NAME: postgres2

```

4 Configure the node definition to meet the requirements of the cluster:

■ NODE_NUMBER

Specifies the sequential node identifier. This number should change to fit with the target cluster. For example, if the last standby node in the cluster is 1, the node that you are moving should have a `NODE_NUMBER` of 2. If there is only a primary node in the target cluster, the node that you are moving should have a `NODE_NUMBER` of 1.

■ NODE_TYPE

Specifies the type of node that you are moving. The only acceptable value is `s` (standby). After initial deployment, you cannot move a primary node.

■ PG_PORT

Specifies the Postgres database port. The pgpool server communicates with the database on this port. Clients use the `PGPOOL_PORT`. The port must be available for use on the deploy target.

■ SANMOUNT

Specifies the location of the data files. This path is typically the same value as other data nodes.

■ SERVICE_NAME

Specifies the service name for the data server cluster.

Note: Do not change this value.

5 Run your Ansible playbook using the `sitedefault.yml` file.

Here is an example:

```
ansible-playbook site.yml
```

For a complete list of playbook commands, see [“Commands” in SAS Viya for Linux: Deployment Guide](#).

Change the Port Number or the Data Directory for a Node (Ansible)

CAUTION! To avoid data corruption, do not change the port number or the data directory on a primary node.

- 1 As the SAS install user (`sas`) or with `sudo` privileges, sign in to the pgpool machine.
- 2 To stop the node whose port or directory you want to change, run the following command:

```
sudo service sas-viya-sasdatasvrc-service-name stop
```

Here is an example:

```
sudo service sas-viya-sasdatasvrc-postgres stop
```

- 3 To check the status of the node, run the following command:

```
sudo service sas-viya-sasdatasvrc-service-name stop
```

Here is an example:

```
sudo service sas-viya-sasdatasvrc-postgres stop
```

Verify that failover has successfully occurred. In the cluster status list, the **status** of the node should be down.

- 4 With administrator privileges, sign in to your Ansible controller, and locate the file `/playbook/vars.yml`.
- 5 Using a text editor, open `vars.yml` and locate the `INVOCATION_VARIABLES` section.

```
INVOCATION_VARIABLES:
  Machine1:
    pgpoolc:
      - PCP_PORT: '5430'
        PGPOOL_PORT: '5431'
        SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
        SERVICE_NAME: postgres
    sasdatasvrc:
      - NODE_NUMBER: '0'
        NODE_TYPE: P
        PG_PORT: '5432'
        SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
        SERVICE_NAME: postgres
      - NODE_NUMBER: '1'
        NODE_TYPE: S
        PG_PORT: '5432'
        SANMOUNT: '{{ SAS_CONFIG_ROOT }}/data/sasdatasvrc'
        SERVICE_NAME: postgres
```

- 6 Make the necessary changes to the port number or the data directory in the definition for the node:

■ **NODE_NUMBER**

Specifies the sequential node identifier.

Note: Do not change this value.

■ **NODE_TYPE**

Specifies the type of node: P (primary) or S (standby).

Note: Do not change this value.

■ **PG_PORT**

Specifies the Postgres database port. The pgpool server communicates with the database on this port. Clients use the `PGPOOL_PORT`. The port must be available for use on the deploy target.

■ **SANMOUNT**

Specifies the location of the data files. This path is typically the same value as other data nodes.

■ **SERVICE_NAME**

Specifies the service name for the data server cluster.

Note: Do not change this value.

- 7 Run your Ansible playbook using the `sitedefault.yml` file.

Here is an example:

```
ansible-playbook site.yml
```

For a complete list of playbook commands, see [“Commands” in SAS Viya for Linux: Deployment Guide](#).

- 8 Check the status of the node. In the cluster status list, the **status** of the node should be up.

How To (General)

Get Current Passwords

- 1 As the SAS install user (sas) or with sudo privileges, sign in to any SAS Infrastructure Data Server machine.

- 2 Obtain the security token from the configuration server, and set it as an environment variable, using the appropriate command:

- Install user or root accounts:

```
export CONSUL_TOKEN=$(cat /opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/consul/default/client.token)
```

- With sudo privileges (but not as the install user), install accounts:

```
export CONSUL_TOKEN=$(sudo cat /opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/consul/default/client.token)
```

- 3 Run the sas-bootstrap-config script for the [data server user ID](#) whose password you want to obtain:

- sas

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv read config/postgres/sas.dataserver.pooluser/common/sr_check_password
```

- dbmsowner

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv read config/application/sas/database/postgres/password
```

Change User Passwords

The script, sds_change_user_pw.sh, changes SAS Infrastructure Data Server passwords and synchronizes them with SAS Configuration Server (Consul) and configuration files.

CAUTION! To avoid data loss, change the sas user account password only during a scheduled maintenance when users are not accessing SAS Viya. The data server must be running when you change the sas user's password Changing the password for the database user, sas, causes all nodes on the database cluster to restart.

Note: To change the password, you must know the current password. For more information, see [“Get Current Passwords”](#).

- 1 As the SAS install user (sas), sign in to the SAS Infrastructure Data Server Pgpool machine.

Note: The change user password script requires sudo execution privileges.

- 2 You can determine the status of your cluster by running the cluster init status command as a service.

- 3 Run the following command:

```
sudo service sas-viya-sasdatasvrc-service-name status
```

Before you run the change password script, verify that the cluster is in its initial configuration state (running and healthy):

- node0 has the **role** of `primary`
- all other nodes have a **role** of `standby`
- all nodes have a **status** of `up`

Here is an example where the data server service is named `postgres2`:

```
sudo service sas-viya-sasdatasvrc-postgres2 status
```

Here is typical output:

```
Checking status of sas-viya-sasdatasvrc-postgres2...

PGPool is running with PID=11445
Checking Postgresql nodes status...
node_id | hostname | port | status | lb_weight | role | select_cnt | load_balance_node | replication_delay
-----+-----+-----+-----+-----+-----+-----+-----+-----
0       | machine1 | 5452 | up     | 0.250000 | primary | 1          | false              | 0
1       | machine2 | 5452 | up     | 0.250000 | standby | 0          | true               | 0
2       | machine3 | 5452 | up     | 0.250000 | standby | 0          | false              | 0
3       | machine4 | 5452 | up     | 0.250000 | standby | 0          | false              | 0
(4 rows)
```

- 4 Locate the data server environment variables file, `sds_env_var.sh`, and record its location.

By default, `sds_env_var.sh` resides in `/opt/sas/viya/config/etc/sasdatasvrc/postgres/pgpool0`.

- 5 The script prompts for the following information. Have this information ready when you run the script in a later step:
 - database user name
 - current database password
 - new database password

Note: Your password must conform to the data server [password policy on page 707](#).

- 6 Using the location of `sds_env_var.sh` noted in [Step 4](#), run the script using the following command:

```
sudo -Hu sas /opt/sas/viya/home/libexec/sasdatasvrc/script/sds_change_user_pw.sh
-config_path
/opt/sas/viya/config/etc/sasdatasvrc/postgres/pgpool0/sds_env_var.sh
```

TIP If you run the script from the directory where it resides, you might see several `cannot open [No such file or directory]` messages. This is a known issue, and you can safely ignore these messages.

- 7 Enter the information that you collected in [Step 5](#) as the script prompts you for it.

After you provide the values in response to the prompts, the script connects to SAS Configuration Server and updates all instances of the database user password that it finds. Changes made in the configuration server are synchronized with the proper SAS Infrastructure Data Server configuration files. Finally, the script issues the necessary SQL commands in the data server to update the permissions for the database user.

- 8 To validate that your password has successfully changed, connect to the data server [first database](#), `postgres`, using the PostgreSQL interactive terminal, `psql`:

```
/opt/sas/viya/home/bin/psql -h data-server-machine-name -U user-IDservice-name
```

- 9 When prompted, enter the new password for `dbmsowner`.

10 Restart all SAS Viya services.

For more information, see [“All Servers and Services” on page 599](#).

Clean Up after a Hardware Failure

If the machine on which the high availability (HA) SAS Infrastructure Data Server cluster runs was stopped unexpectedly, you might need to perform some cleanup steps after you restart the machine.

These steps involve removing any socket-lock files and any PID files that might have become orphaned after the PostgreSQL and pgpool servers were improperly shut down.

- 1 As the SAS install user (sas) or with sudo privileges, sign in to the pgpool machine.
- 2 Stop the HA data server cluster by running the appropriate command:
 - Red Hat Linux version 6:


```
sudo service sas-viya-sasdatasvrc-postgres stop
```
 - Red Hat Linux version 7:


```
sudo systemctl stop sas-viya-sasdatasvrc-postgres
```
- 3 Delete any socket-lock file (in the form `.s.PGSQL.xxxx`) or any PID file (in the form `server.pid`) that corresponds to your HA data server cluster ports.

For the default HA data server instance with one data node, remove the following files:

- `/tmp/.s.PGSQL.5430`
 - `/tmp/.s.PGSQL.5431`
 - `/tmp/.s.PGSQL.5432`
 - `/tmp/.s.PGSQL.5432.lock`
 - `/opt/sas/viya/config/data/sasdatasvrc/postgres/node0/postmaster.pid`
 - `/opt/sas/viya/config/data/sasdatasvrc/postgres/pgpool0/run/pgpool.pid`
- 4 Restart the HA data server cluster by running the appropriate command:
 - Red Hat Linux version 6:


```
sudo service sas-viya-sasdatasvrc-postgres start
```
 - Red Hat Linux version 7:


```
sudo systemctl start sas-viya-sasdatasvrc-postgres
```

Remove a Persistent Lock on a Database Table

Persistent locks on a SAS Infrastructure Data Server database table are caused by an uncommitted transaction or a long running query. To fix this problem, you must identify the process IDs of the client connections that are locking the table and terminate these connections.

- 1 As the SAS install user (sas) or with sudo privileges, sign in to the pgpool server machine.
- 2 If you know the PostgreSQL dbmsowner (superuser) password, go to [Step 3](#). Otherwise, follow the steps in [“Get Current Passwords”](#).
- 3 If you have not already done so, install pgAdmin on any machine (including Microsoft Windows) that has access to the machine that is running the pgpool server.

4 In pgAdmin, perform the following steps:

a Create a **New Server Registration** object and specify the following information:

- **Host:** *machine-name*
The name of the machine on which the pgpool server resides.
- **Port:** *pgpool-client-connection-port*
The default port is 5431.
- **Maintenance DB:** *SharedServices*
Do not use the default, *postgres*.
- **Username:** *superuser*
The database superuser. The default is *dbmsowner*.
- **Password:** string
The superuser password.

b Connect to the pgpool server.

c Highlight the server name, and choose **Tools** ⇒ **Server Status**.

The status panel shows all the client connections in the top panel. The second panel will show the persistent locks.

d Choose **Actions** ⇒ **Refresh** multiple times in order to determine whether the listed locks are transient or persistent. A transient lock disappears, and a persistent lock remains throughout refreshes.

e (Optional) You can cross-reference the process identifiers (PIDs) for the locked tables with the connection listing in order to identify the client (the application name) that has locked the table.

Note: If you choose this option, open a SAS Technical Support track about this issue. Include the **PID**, **Application Name**, **Connection State**, and **Query** (if applicable) from the top connections section. Also include the PID and the persistent locked table names from the **Lock** section.

f To clear the locks, run the **pg_terminate_backend()** command on each PID that has a persistent lock.

To do this, go back to the main pgAdmin panel. Highlight the **SharedServices** database name in the server **Object Browser** and choose **Tools** ⇒ **Query Tool** to open an SQL query execution window.

g Execute the **pg_terminate_backend(__PID__)** command to close each connection that is associated with a table that has a persistent lock.

Here is an example:

```
SELECT pg_terminate_backend(14826);
SELECT pg_terminate_backend(16697);
SELECT pg_terminate_backend(22246);
```

h Select **Tools** ⇒ **Server Status** and refresh the panel (**Actions** ⇒ **Refresh** from the menu).

If all the persistent locks have been removed from the second **Locks** section, the persistent locks are successfully removed.

i Exit pgAdmin.

Routine Maintenance Tasks

Overview

Routine maintenance for a SAS Infrastructure Data Server consists of the following tasks:

- Adhering to a rigid schedule of performing [database backups](#).
- Performing [a re-index, vacuuming, and analyzing each table in the database](#) during a maintenance cycle.
- Inspecting the data server [logs](#) periodically to make sure that there is no data corruption.
- [Removing large orphaned data objects](#) from the database periodically to free disk space.
- Track PostgreSQL software patches, and apply the patches that contain critical fixes, such as the [CVE-2017-7547 security patch](#).

Re-Index, Vacuum, and Analyze Database Tables

Follow these steps to re-index, vacuum, and analyze each table in the SAS Infrastructure Data Server databases. SAS recommends that you perform these steps during a maintenance cycle in order to reduce the chance of a PostgreSQL database command hanging because of a long-term lock on a table. If you encounter a hang condition, to remove the lock, you might need to restart the SAS Infrastructure Data Server.

- 1 As the SAS install user (sas) or with sudo privileges, sign in to the pgpool server machine.
- 2 If you know the PostgreSQL dbmsowner (superuser) password, go to [Step 3](#). Otherwise, follow the steps in [“Get Current Passwords”](#).
- 3 Run the following commands to set up the PostgreSQL command-line environment:


```
export PATH=/opt/sas/viya/home/bin:$PATH
export LD_LIBRARY_PATH=/opt/sas/viya/home/lib:/opt/sas/viya/home/lib64:$LD_LIBRARY_PATH
```
- 4 (Optional) [Stop all SAS Viya services](#), and [run only SAS Infrastructure Data Server](#).
- 5 Run the following commands:

Note: For illustration, 5431 is used as the client connection port, /opt/sas/viya/home is used as the installation directory, and dbmsowner is used as the database superuser. Substitute the values that are appropriate for your site.

TIP To prevent having to enter the superuser password multiple times, you can create a ~/.pgpass file.

- Re-index all databases:


```
./reindexdb -a -p 5431 -h localhost -U dbmsowner
```
- Perform a full vacuum and analyze all databases:


```
./vacuumdb -p 5431 -h localhost -U dbmsowner -f -v -z -a
```

Note: If you encounter a hang condition, you might need to [restart the SAS Infrastructure Data Server](#) to remove the lock.

If there were no errors in the previous step, then you are done.

If you stopped the all of the SAS Viya services, then you can now [stop the SAS Infrastructure Data Server](#) and [restart all the SAS Viya services](#).

Remove Large Orphaned Data Objects

Large objects in the SAS Infrastructure Data Server database are stored separately from the tables that reference them. When a particular row is updated or deleted, these objects can become orphaned (unattached) from a table. Periodically, these orphaned large objects must be manually removed to free disk space.

- 1 Create an SQL command file named `lo-cleanup.sql` with the following content:

```
DROP FUNCTION IF EXISTS sas_lob_cleanup();
CREATE FUNCTION sas_lob_cleanup() RETURNS VOID AS $function$
DECLARE
    possible_lob_col record;
    possible_oid_row record;
    possible_oid_val bigint;
BEGIN
    DROP TABLE IF EXISTS sas_possible_lob;

    CREATE TABLE sas_possible_lob (table_schema TEXT NOT NULL, table_name
    TEXT NOT NULL, column_name TEXT NOT NULL, column_value BIGINT);

    FOR possible_lob_col IN SELECT * FROM information_schema.columns WHERE
    udt_name IN ('int4', 'int8', 'numeric', 'oid', 'text', 'varchar',
    'char', 'lo') AND NOT (table_schema = 'pg_catalog' AND table_name =
    'pg_shdepend') AND NOT (table_schema = 'pg_catalog' AND table_name =
    'pg_largeobject') AND table_name != 'sas_possible_lob'

    LOOP

        BEGIN

            FOR possible_oid_val IN EXECUTE 'SELECT CAST(' ||
            possible_lob_col.column_name || ' AS BIGINT) FROM ' ||
            possible_lob_col.table_schema || '.' || possible_lob_col.table_name
            || ' WHERE ' || possible_lob_col.column_name || ' IS NOT NULL AND
            CAST(' || possible_lob_col.column_name || ' AS BIGINT) < ((2 ^ 32) -
            1) AND CAST(' || possible_lob_col.column_name || ' AS BIGINT) > 0'

            LOOP

                --raise notice 'successfully cast % % %',
                possible_lob_col.table_schema,
                possible_lob_col.table_name,
                possible_lob_col.column_name;

                INSERT INTO sas_possible_lob (table_schema, table_name,
                column_name, column_value) VALUES (possible_lob_col.table_schema,
                possible_lob_col.table_name, possible_lob_col.column_name,
                possible_oid_val);

            END LOOP;

        EXCEPTION

            WHEN cannot_coerce THEN

                --RAISE NOTICE 'error coercing % % %',
```

```

possible_lob_col.table_schema, possible_lob_col.table_name,
possible_lob_col.column_name;

    WHEN invalid_text_representation THEN

        --RAISE NOTICE 'error casting % % %',
possible_lob_col.table_schema, possible_lob_col.table_name,
possible_lob_col.column_name;

    WHEN others THEN

        RAISE NOTICE 'unexpected failure';

END;

END LOOP;

SELECT LO_UNLINK(lo.loid) FROM pg_catalog.pg_largeobject lo GROUP BY loid
HAVING (NOT EXISTS (SELECT 1 FROM public.sas_possible_lobs pl WHERE lo.loid
= pl.column_value));

DROP TABLE IF EXISTS sas_possible_lobs;

END;

$function$ LANGUAGE PLPGSQL;

SELECT sas_lob_cleanup();

```

2 As the SAS install user (sas) or with sudo privileges, sign in to the pgpool server machine

3 Run the following command for each database:

Note: For illustration, 5431 is used as the client connection port, /opt/sas/viya/home is used as the installation directory, and dbmsowner is used as the database superuser. Substitute the values that are appropriate for your site.

```
psql -p 5431 -h localhost -U dbmsowner -d postgres -a -f lo-cleanup.sql
```

```
psql -p 5431 -h localhost -U dbmsowner -d SharedServices -a -f lo-cleanup.sql
```

Apply the CVE-2017-7547 Security Patch

A new security patch, [CVE-2017-7547](#), fixes a password security issue in PostgreSQL databases.

Sites that deploy SAS Viya 3.3 have a newer version of PostgreSQL (version 9.4.13) that contains the fix for this security issue.

However, sites running SAS Viya 3.2 (and earlier) that upgrade to SAS Viya 3.3, must manually apply patch SAS Infrastructure Data Server with patch CVE-2017-7547.

- 1** Make sure that you have upgraded to SAS Viya 3.3.
- 2** As the SAS install user (sas) or with sudo privileges, sign in to the pgpool server machine.
- 3** If you know the PostgreSQL dbmsowner (superuser) password, go to [Step 4](#). Otherwise, follow the steps in [“Get Current Passwords”](#).
- 4** [Stop all SAS Viya services](#), and [run only the SAS Infrastructure Data Server](#).

5 Run the CVE maintenance script:

```
sudo /opt/sas/viya/home/libexec/sasdatasvrc/script/maintenance/CVE-2017-7547.sh
```

6 Stop the SAS Infrastructure Data Server and restart all the SAS Viya services.

Concepts

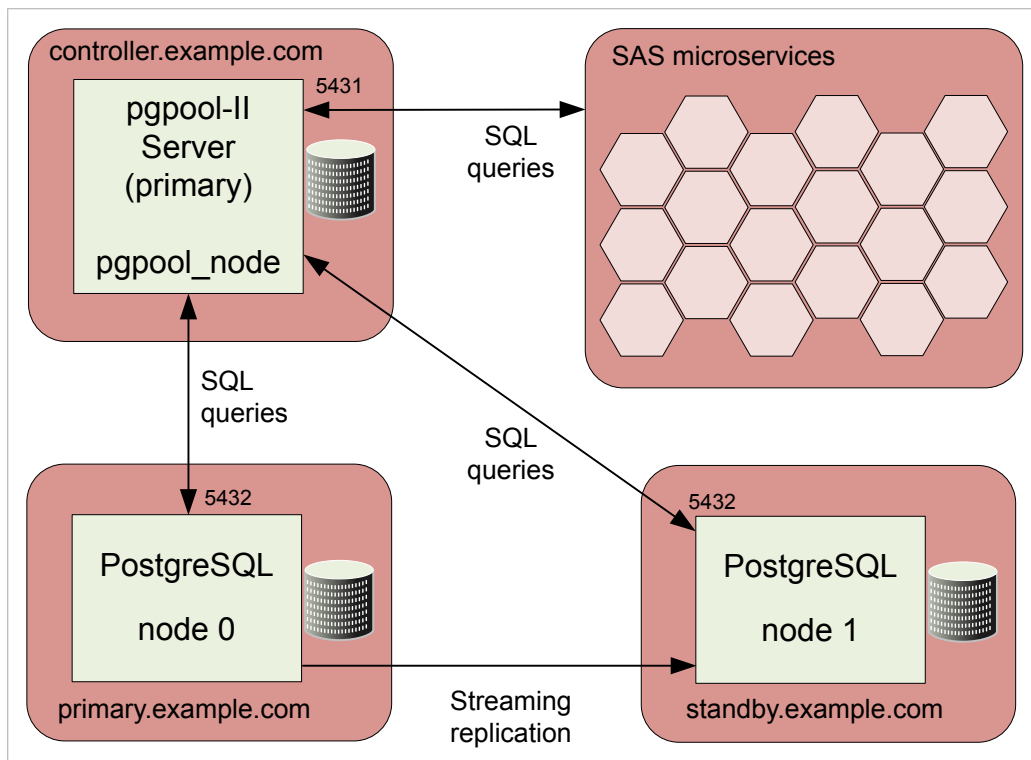
What is the SAS Infrastructure Data Server?

SAS Infrastructure Data Server is used for transactional storage by SAS middle-tier software. It is also used by some SAS solutions software for user content such as reports, custom groups, comments, authorization rules, selected source definitions, attachments, and user preferences. The server is configured specifically to support SAS software, and is based on PostgreSQL version 9.

By default, the SAS installer account is used to start the server.

The databases that are managed by the server are backed up and restored with the Backup and Recovery Deployment Tool. For more information, see [SAS Viya Administration: Backup and Restore](#).

Figure A.2 SAS Infrastructure Data Server Architecture



Pgpool-II

SAS provides Pgpool-II (version 3) open-source software to enable you to manage PostgreSQL clusters. The Pgpool-II software resides and operates between SAS Infrastructure Data servers and clients. All data connections and database requests are routed through the pgpool service.

- High availability (failover management)
- Load balancing (read SELECT query scaling, but not writes)

- Connection pooling

Troubleshooting

psql: server closed the connection unexpectedly. This probably means the server terminated abnormally before or while processing the request.

Explanation:

The SAS Viya environment was shut down abnormally.

Resolution:

Restart the SAS Viya environment using the `sas-viya-all-services start` command. For more information, see [“All Servers and Services” on page 599](#).

`/opt/sas/viya/config/etc/sasdatasvrc/./node.cdf` was already marked with 'recoveryInProgress=y'. Exiting from auto-recovery.

Explanation:

PostgreSQL was in the process of recovering the SAS Infrastructure Data Server node when it encountered an error, and stopped the recovery process. Whenever it restarts a data server node, PostgreSQL always inserts the line, `recoveryInProgress=y`, in the `node.cdf` file to avoid a simultaneous recovery.

Resolution:

- 1 Review the recovery log to determine what the problem is.

(The recovery log is located here: `/opt/sas/viya/config/var/log/sasdatasvrc/cluster/nodex/sds_auto_recovery_node.log`.)

- 2 Fix the problem.

- 3 Remove the following line from the node's `node.cdf` file:

```
recoveryInProgress=y
```

- 4 Restart (recover) the node.

**EDTERROR: missing chunk number 0 for toast value 9558737 in pg_toast_2619
EDTCONTEXT: automatic analyze of table "SharedServices.public.sas_audit"
EDTERROR: could not read block 3062 in file "base/18797/19703": read only 0 of 8192 bytes
yyyy-mm-dd EDTERROR: unexpected data beyond EOF in block 0 of relation base/16715/107679**

Explanation:

There is a high probability that your SAS Infrastructure Data Server database is corrupted.

Resolution:

After you correct the cause of the data corruption, and [recover the database using a restored backup](#).

**ERROR: Cluster stop failed. Please review the log file. `/opt/sas/viya/config/var/log/sasdatasvrc/postgres/pgpool0/sas-viya-sasdatasvrc-postgres-service_YYYYMMDD_####.log` [FAILED]
Unexpected response code: 500 ERROR: Unable to read a key Unexpected response code: 500 (rpc error: failed to get conn: dial tcp someotherhost.com:8300: getsockopt: connection refused) ERROR: Unable to list the nodes that provide the service 'postgres'**

Explanation:

SAS Infrastructure Data Server fails to stop because it cannot connect to SAS Configuration Server (Consul) even though Consul is running.

This problem occurs in a multi-machine deployment where the primary data server node is not on the same system as the primary Consul server (the server designated in the `inventory.ini` file). If the primary Consul server has already been shut down, the data server fails to stop, even if a local Consul service is running.

Resolution:

Always stop SAS Infrastructure Data Server before the primary Consul server, regardless of which machine it is located on. For more information, see [“Read This First: Start and Stop Servers and Services” in SAS Viya Administration: General Servers and Services](#).

Reference

Recommended Connection, ulimit, and Semaphore Settings

Connection Settings

To change the number of connections available to clients, [modify the following configuration properties](#):

- `sas.dataserver.conf.common.max_connections`

For more information, see <https://www.postgresql.org/docs/9.1/static/runtime-config-connection.html#RUNTIME-CONFIG-CONNECTION-SETTINGS>.

- `sas.dataserver.pool.common.num_init_children`

For more information, see http://www.pgpool.net/docs/pgpool-II-3.5.4/doc/pgpool-en.html#NUM_INIT_CHILDREN.

Note: The `max_connections` value should be slightly higher than the `num_init_children` value to allow for direct connections outside `pgpool` for administrative use, such as backup and recovery.

ulimit Settings

Recommended settings for the `sas` user in `/etc/security/limits.conf`:

```

sas soft nofile 150000
sas hard nofile 150000
sas soft nproc 100000
sas hard nproc 100000
sas soft stack 10240
sas hard stack 10240

```

Semaphore Settings

Recommended semaphore settings in `/etc/sysctl.conf`:

```

kernel.sem=512 32000 100 1024
net.core.somaxconn=2048

```

for `SEMMSL`, `SEMMNS`, `SEMOPM`, and `SEMMNI`

For more information, including formulas and minimum values, see <https://www.postgresql.org/docs/9.5/static/kernel-resources.html>.

Note: Changing Linux semaphore settings requires a machine reboot.

Note: You might need to adjust additional Linux operating system settings in order to support these recommended `ulimit` and semaphore settings. To optimize your PostgreSQL resources, you should also scale the server's working memory settings in accordance with [Tuning the PostgreSQL Data Server](#) in *SAS Web Applications: Tuning for Performance and Scalability*.

Database

TIP All PostgreSQL data servers have a *first database* named **postgres**. For more information, see [Creating a Database](#) in PostgreSQL documentation.

In a SAS Viya deployment, SAS Infrastructure Data Server is configured to manage the SharedServices database. SAS Viya microservices create database schemas within SharedServices.

If your deployment includes SAS solutions software that supports SAS Infrastructure Data Server, more databases might be configured on the server.

Default Users

dbmsowner

The PostgreSQL database owner and the SAS database administrator user.

sas

The SAS Viya install user and the account used for SAS Infrastructure Data Server cluster management.

Network Access

SAS Infrastructure Data Server is configured to accept connections on all network interfaces, and it requires password authentication. By default, SAS configures the server to use network port number 5431.

PostgreSQL instances are configured with JDBC data sources that reference the SharedServices database.

Password Policy

The user name and password for the SAS Infrastructure Data Server administrator are specified during deployment. The password can be updated. Passwords for SAS Infrastructure Data Server are subject to the following guidelines:

- The password must not contain any non-alphanumeric characters.
Examples are underscores (_), hyphens (-), and periods (.).
- The password must be at least six characters long.
- The password can contain alphanumeric characters.
- There are no restrictions for including numbers or mixed-case characters.

Environment Parameters

Export the following path in order to execute PostgreSQL and pgpool commands:

```
export LD_LIBRARY_PATH=/opt/sas/viya/home/lib:/opt/sas/viya/home/lib64
```

Configuration Files

- /opt/sas/viya/config/etc/sasdatasvrc/postgres/node0/node.cdf
- /opt/sas/viya/config/etc/sasdatasvrc/postgres/pgpool0/pool.cdf
- /opt/sas/viya/config/data/sasdatasvrc/postgres/pgpool0/pgpool.conf
- /opt/sas/viya/config/data/sasdatasvrc/postgres/pgpool0/pcp.conf
- /opt/sas/viya/config/data/sasdatasvrc/postgres/pgpool0/pool_hba.conf
- /opt/sas/viya/config/data/sasdatasvrc/postgres/pgpool0/pool_passwd

- `/opt/sas/viya/config/data/sasdatasvrc/postgres/node0/postgresql.conf`
- `/opt/sas/viya/config/data/sasdatasvrc/postgres/node0/pg_hba.conf`

Log Files

SAS Infrastructure Data Server log files are located in `/opt/sas/viya/config/var/log/sasdatasvrc`.

SAS Message Broker

Overview

SAS uses a set of event APIs that are dependent on Spring Integration and Spring AMQP to interact with the message broker. The AMQP-compliant message broker that SAS uses is Pivotal's RabbitMQ, version 3. RabbitMQ includes the Erlang platform, version 19.

Note: A [programming-only deployment on page 1](#) does not use SAS Message Broker.

How To

Operate

SAS Viya uses the operating system's default init system or `systemd` command to launch a script that can stop, start, restart, and check the status of SAS Message Broker. This script, `sas-viya-rabbitmq-server-default`, resides in `/etc/init.d`.

Note: You must be signed in to the machine where message broker resides, and you must have `sudo` privileges to run this script.

To operate the message broker, run the following command, as appropriate:

```
sas-viya-rabbitmq-server-default status | stop | start | restart
```

Note: You can also run a script to manage and view the running state of all SAS Viya servers and services. For more information, see ["All Servers and Services" on page 599](#).

Here are a few examples of how to operate this script:

- To check status of the message broker using a direct call:

```
sudo /etc/init.d/sas-viya-rabbitmq-server-default status
```

- To stop the message broker using the Red Hat Linux version 6 init system command:

```
sudo service sas-viya-rabbitmq-server-default stop
```

- To start the message broker using the Red Hat Linux version 7 `systemd` command:

```
sudo systemctl start sas-viya-rabbitmq-server-default
```

- To restart the message broker using a direct call:

```
sudo /etc/init.d/sas-viya-rabbitmq-server-default restart
```

Concepts

What is SAS Message Broker?

SAS Message Broker is an integral part of the event-driven architecture in which SAS Viya services participate. SAS uses a set of event APIs that are dependent on Spring Integration and Spring AMQP for interacting with the message broker. The AMQP-compliant message broker that SAS uses is Pivotal's RabbitMQ. The SAS event APIs provide a layer of abstraction between the message broker and its clients. The SAS event APIs also prevent code from breaking, which could result if SAS changed its third-party message broker from RabbitMQ to another third-party message broker in the future.

How Does Message Broker Work?

SAS Message Broker accepts messages in a standard format and routes them through exchanges and queues, which provide transaction acknowledgment, message persistence, and redundancy. Message broker exchanges accept messages from publishers and route them to queues, as appropriate. The exchange type controls whether messages are sent to a specific queue, to all associated queues, or only to queues that accept a particular message routing key or that match a key pattern.

SAS Message Broker Reference

Exchanges

SAS Message Broker uses the following exchanges:

- `sas.application`
- `sas.application.backup`
- `sas.backup.topic`
- `sas.ledger`
- `sas.log`
- `sas.metric`
- `sas.notification`
- `sas.search.schema.topic`

Configuration Files

Note: Change these configuration files only when instructed to do so by SAS Technical Support.

- `/opt/sas/viya/config/etc/rabbitmq-server/rabbitmq.config`
- `/opt/sas/viya/config/etc/rabbitmq-server/rabbitmq-env.conf`

Log Files

SAS Message Broker log files are located in `/opt/sas/viya/config/var/log/rabbitmq-server/default`.

SAS Cache Locator and Cache Server

Overview

SAS Cache Locator and SAS Cache Server provide a distributed cache technology to microservices in SAS Viya.

Operate

SAS Viya uses the operating system's default init system or systemd command to launch scripts that can stop, start, restart, and check the status of SAS Cache Locator and SAS Cache Server. These scripts, `sas-viya-cachelocator-default` and `sas-viya-cacheserver-default`, reside in `/etc/init.d`.

Note: You must be signed in to the machine where SAS Cache Locator and SAS Cache Server reside, and you must have sudo privileges to run this script.

To operate SAS Cache Locator or SAS Cache Server, run the following command, as appropriate:

```
sas-viya-cachelocator-default status | stop | start | restart
```

```
sas-viya-cacheserver-default status | stop | start | restart
```

Note: You can use a script to manage and view the running state of all SAS Viya servers and services. For more information, see [“All Servers and Services” on page 599](#).

Here are a few examples of how to operate this script:

- To check status of the cache locator or the cache server using a direct call:

```
sudo /etc/init.d/sas-viya-cachelocator-default status
```

```
sudo /etc/init.d/sas-viya-cacheserver-default status
```

- To stop the cache locator or the cache server using the Red Hat Linux version 6 init system command:

```
sudo service sas-viya-cachelocator-default stop
```

```
sudo service sas-viya-cacheserver-default stop
```

- To start the cache locator or the cache server using the Red Hat Linux version 7 systemd command:

```
sudo systemctl start sas-viya-cachelocator-default
```

```
sudo systemctl start sas-viya-cacheserver-default
```

- To restart the cache locator or the cache server using a direct call:

```
sudo /etc/init.d/sas-viya-cachelocator-default restart
```

```
sudo /etc/init.d/sas-viya-cacheserver-default restart
```

Concepts

SAS Cache Locator

SAS Cache Locator is a server that provides discovery information to SAS Viya microservices for the purpose of forming a distributed data cache. SAS Cache Locator is based on the open source Apache Geode project.

SAS Cache Server

SAS Cache Server hosts long-lived data regions (a cache) and serves the contents to SAS Viya microservices. Like SAS Cache Locator, SAS Cache Server is based on the open source Apache Geode project.

Configuration

SAS Cache Locator and SAS Cache Server embed the Apache Geode API within their respective SAS Viya microservices, `cachelocator` and `cacheserver`.

The `cachelocator` and `cacheserver` microservices enable the cache locator and the cache server to gain access to SAS Configuration Server (Consul) in order to dynamically register and to retrieve properties with the SAS Viya Configuration service. For more information, see [“Non-Spring-Based Servers” on page 221](#).

When configuration changes are made to `cachelocator` and `cacheserver`, you must restart SAS Cache Locator and SAS Cache Server in order for their changes to take effect. For information about how to modify the configuration for `cachelocator` and `cacheserver`, see [“Edit Configuration Instances” on page 215](#).

Log Files

Log files for SAS Cache Locator and SAS Cache Server are located in `/opt/sas/viya/config/var/log/cachelocator/default` and `/opt/sas/viya/config/var/log/cacheserver/default`.

Apache HTTP Server

Overview

SAS Viya uses Apache HTTP Server to serve static HTML content and to proxy client connections. A high-availability proxy environment is not installed by default, but is a supported configuration.

Red Hat Linux version 6 uses Apache HTTP Server upstream v2.2 and Red Hat Linux version 7 uses Apache HTTP Server upstream v2.4. For more information, see [“Apache httpd” in SAS Viya for Linux: Deployment Guide](#).

How To

Operate

Note: You must be signed in to the machine where Apache HTTP Server resides, and you must have `sudo` privileges to run this script.

Note: For complete information about `httpd` arguments, see <https://httpd.apache.org/docs/2.0/programs/httpd.html>

- To the operate HTTP Server on Red Hat Linux version 6, use the `init` system command:

```
sudo service httpd status | stop | start | restart
```

In this example, the following command checks the status of Apache HTTP Server on the current machine:

```
sudo service httpd status
```

- To the operate HTTP Server on Red Hat Linux version 7, use the `systemd` command:

```
sudo systemctl status | stop | start | restart httpd
```

In this example, the following command stops Apache HTTP Server on the current machine:

```
sudo systemctl stop httpd
```

Change Time-Out Interval

When SAS web applications return HTTP 502 proxy errors, you might have to change the time-out interval for your Apache HTTP Server.

- 1 Sign in as the SAS install user (sas), or sign on with sudo privileges, to the Apache HTTP Server machine.
- 2 Using a text editor, open `/etc/httpd/conf/httpd.conf`.
- 3 Modify the `Timeout` and `Keepalive` values as follows:

```
Timeout 2400  
Keepalive On
```

- 4 Add the following two parameters, and save the file:

```
ProxyTimeout 2400  
ProxyBadHeader Ignore
```

- 5 **Restart** Apache HTTP Server.

Locate Log Files

Note: You must be signed in with sudo privileges to the machine where the service resides in order to view log files.

By default, Apache HTTP Server log files are located in `/var/log/httpd`.

Concepts

SAS Viya uses Apache HTTP Server as a web server. HTTP Server serves static HTML content and proxies client communication.

A third-party load balancer is required in order to provide high availability for HTTP Server. You can also install your own web server on a separate machine in order to proxy connections from the internet to HTTP Server. For more information about making HTTP Server highly available, see [“Apache httpd” in SAS Viya for Linux: Deployment Guide](#).

Tuning

Overview

In SAS Viya, you can tune your environment for performance and scalability. This document includes the following tuning methodologies and tuning parameters:

- Java Virtual Machine (JVM)
- Java Database Connectivity (JDBC) connection pool
- Lightweight Directory Access Protocol (LDAP) connection pool
- Apache HTTP Server
- SAS Infrastructure Data Server
- Linux operating system

Performance requirements are usually identified in terms of transaction response time, number of transactions per second, throughput, resource utilization, total cost per transaction, availability, and more. Scalability often refers to the ability of a component to adapt readily to a greater or lesser intensity of use, volume, or demand, while meeting integral business objectives. The common objective of scaling a component or system is to increase the capacity for growth, increase the speed of the component, improve the efficiency, or shift or reduce the load on the component.

Tuning the Java Runtime Environment

Overview

Note: This information does not apply to Cloud Foundry.

The goal of Java Runtime Environment (JRE) tuning is to improve performance in the services, particularly in the area of memory usage and garbage collection cycles. The goal is to also maximize the number of clients that the SAS web applications can support.

JRE Tuning Recommendations

The default JRE tuning options that are applied for each service should be sufficient. However, you might need to limit how much the native memory usage grows for each Java process. To limit the growth, add the following lines to the `viya-installation-directory/etc/sysconfig/sas-javaesnt1/sas-java-services` file:

```
# Limit the number of "malloc arenas" to 4 (default behavior is to use (# of cores * 8))
export MALLOC_ARENA_MAX=4
```

Tuning the JDBC Connection Pool

Overview




Note: This information applies to Cloud Foundry and Linux.

In Java Database Connectivity (JDBC) connection pooling, instead of creating connections every time they are requested, connections are reused. The JDBC connection pool is a collection of database connection objects that are available for reuse. It is maintained by a connection pooling module as a layer on top of the JDBC driver.

Configure Deployment Size

SAS Viya provides default JDBC connection pool settings for small, medium, and large deployments. By default, all services are configured to use the medium deployment settings.

To configure a different deployment size for a service, complete the following tasks:

- 1 From SAS Environment Manager, navigate to the **Definitions** view.
- 2 In the **Definitions** list, select **jvm**.
- 3 In the top right corner of the window, click .
- 4 In the New jvm Configuration dialog box, complete the following tasks:
 - a Choose one or more services to which the new settings apply by clicking  and selecting the services.
 - b Click **OK**.
 - c Click .
 - d In the **Name** field, specify `java_option_springdatasource_default`.
 - e In the **Value** field, specify `Dsas.deployment.springdatasource.defaults=size`, where *size* is small, medium, or large.
 - f Click **Save**.
- 5 Click **Save**.
- 6 Restart all SAS Viya services.

See Also




- [“Create Configuration Instances” on page 216](#)
- [“All Servers and Services” on page 599](#)

Default Datasource Properties

By default, there are predefined property settings for small, medium, and large deployments. You can override these values. For example, you can set the Preferences Service to use the default property values for a small system, but override the default value of the `spring.datasource.tomcat.maxIdle` property by changing it from 2 to 3.

Override Default Property Values

To change the default property settings, complete the following tasks:

- 1 From SAS Environment Manager, navigate to the **Definitions** view.
- 2 In the **Definitions** list, select **spring**.
- 3 In the top right corner of the window, click .
- 4 In the New spring Configuration dialog box, complete the following tasks:
 - a Choose one or more services to which the new settings apply by clicking  and selecting the services.
Note: Not all services use the default property settings. Instead, those services specify a scaling factor that enables them to have larger pool sizes, based on the deployment size that is specified in the `sas.deployment.springdatasource.defaults` property. For more information, see [“Datasource Scaling Factor” on page 716](#).
 - b Click **OK**.
 - c Click .
 - d In the **Name** field, specify a property from the [Property Settings Table on page 715](#).
 - e In the **Value** field, specify the new size that you want to set the property.
 - f Click **Save**.
- 5 Click **Save**.

See Also

[“Create Configuration Instances” on page 216](#)

Default Datasource Property Settings

The following table provides the default property settings for the JDBC connection pool, based on the deployment size:

Table A.53 Property Settings Table

Property	Small Deployment	Medium Deployment	Large Deployment
<code>datasource.tomcat.initialSize</code>	2	2	2
<code>datasource.tomcat.maxActive</code>	6	10	20
<code>datasource.tomcat.maxIdle</code>	2	2	2
<code>datasource.tomcat.minIdle</code>	2	2	2




A service can also provide default files for small, medium, and large deployments in the service resource directory.

Datasource Scaling Factor

Not all services use the default property settings. Instead, those services specify a scaling factor that enables them to have larger pool sizes, based on the deployment size that is specified in the `sas.deployment.springdatasource.defaults` property. For example, the Authorization Service specifies a scaling factor of 10. Therefore, its `maxActive` value is 60 for small, 100 for medium, and 200 for large deployments. For a list of the default property values, see [Table 38.53 on page 715](#)

Configure the Scaling Factor

To define a scaling factor for a service, complete the following tasks:

- 1 From SAS Environment Manager, navigate to the **Definitions** view.
- 2 In the **Definitions** list, select **jvm**.
- 3 In the top right corner of the window, click .
- 4 In the New jvm Configuration dialog box, complete the following tasks:
 - a Choose one or more services to which the new settings apply by clicking  and selecting the services.
 - b Click **OK**.
 - c Click .
 - d In the **Name** field, specify the `java_option_datasource_factor` property.
 - e In the **Value** field, specify `Dsas.datasource.custom.factor=multiplier`, where *multiplier* is the multiplier factor by which the property in the [Property Settings Table on page 715](#) will be multiplied.
 - f Click **Save**.
- 5 Click **Save**.
- 6 Restart all SAS Viya services.

See Also

- [“Edit Configuration Instances” on page 215](#)
- [“All Servers and Services” on page 599](#)

Scaling Factor Example

You can specify a multiplier factor for a service by using the `sas.datasource.custom.factor` property. The default value for a property is multiplied by the value that you specify. For example, for a medium deployment, the default value for the `spring.datasource.tomcat.maxActive` property is 10. If you set the multiplier factor to 5, the new `maxActive` value is 50. The factor must be greater than 0. The resulting `maxActive` value will be no less than `spring.datasource.tomcat.initialsize` and no more than 200.

Tuning the LDAP Connection Pool

Overview

Note: This information applies to Cloud Foundry and Linux.

The Lightweight Directory Access Protocol (LDAP) service provider supports connection pooling. In LDAP connection pooling, the service provider maintains a pool of previously used connections. When a connection is closed or goes to garbage collection, it goes back to the pool to be used again.

By default, no configuration is required for the LDAP service provider to use connection pooling. However, configuration is needed to customize the setting for optimal performance.

LDAP Tuning Recommendations

- 1 In SAS Environment Manager, edit the **Identities service**.
- 2 Navigate to the **sas.identities.providers.ldap.connection** configuration instance and configure the following:

Property	Value
pool.maxActive	30
pool.maxIdle	30

See Also

[“Edit Configuration Instances” on page 215](#)

Tuning the Apache HTTP Server

Overview

Note: This information does not apply to Cloud Foundry.

You can improve the performance of the Apache HTTP Server by configuring other aspects of the web server. For example, to improve performance, rotate log files and configure the Multi-Processing Modules (MPMs).

For more information about MPMs, see <https://httpd.apache.org/docs/2.4/mpm.html>.

Apache HTTP Server Recommendations

- 1 For sites with upward of 400 users, it is recommended that you enable the following Apache HTTP modules:

- Apache 2.2 and later: **worker**

In `/etc/sysconfig/httpd`, uncomment the following line:

```
HTTPD=/usr/sbin/httpd.worker
```

- Apache 2.4 and later: **mod_mpm_worker.so**

In `/etc/httpd/conf.modules.d/00-mpm.conf`, *comment* the line ending in `mod_mpm_prefork.so`, and *uncomment* the line ending in `mod_mpm_worker.so`:

```
#LoadModule mpm_prefork_module modules/mod_mpm_prefork.so
LoadModule mpm_worker_module modules/mod_mpm_worker.so
#LoadModule mpm_event_module modules/mod_mpm_event.so
```

2 Configure the Apache HTTP Server to use the worker MPM as follows:

- For Apache 2.2, modify the `/etc/httpd/conf/httpd.conf` file to adjust worker MPM settings. Add the `ServerLimit` setting and change the value for the other settings that are highlighted in the sample file below:

```
# worker MPM
# StartServers: initial number of server processes to start
# MaxClients: maximum number of simultaneous client connections
# MinSpareThreads: minimum number of worker threads which are kept spare
# MaxSpareThreads: maximum number of worker threads which are kept spare
# ThreadsPerChild: constant number of worker threads in each server process
# MaxRequestsPerChild: maximum number of requests a server process serves
<IfModule worker.c>
    ServerLimit      32
    StartServers    10
    MaxClients      1024
    MinSpareThreads 25
    MaxSpareThreads 75
    ThreadsPerChild 32
    MaxRequestsPerChild 0
</IfModule>
```

- For Apache 2.4, add the following configuration block to the existing configuration file (it is recommended that you modify either the `/etc/httpd/conf.modules.d/00-mpm.conf` file or the `/etc/httpd/conf/httpd.conf` file):

```
<IfModule mpm_worker_module>
    ServerLimit      32
    StartServers    10
    MaxRequestWorkers 1024
    MinSpareThreads 25
    MaxSpareThreads 75
    ThreadsPerChild 32
    MaxConnectionsPerChild 0
</IfModule>
```

- ## 3 Identify a suitable log rotation strategy and modify the `/etc/httpd/conf/httpd.conf` file to configure the Apache `rotatelogs` tool to perform log rotation. For information about rotation strategies and configuration options, see <https://http.apache.org/docs/2.4/programs/rotatelogs.html>.

The following are sample `httpd.conf` file entries for configuring daily log rotation:

```
#ErrorLog logs/error_log
ErrorLog "|/usr/sbin/rotatelogs logs/error_log 86400"

#CustomLog logs/access_log combined
CustomLog "|/usr/sbin/rotatelogs logs/access_log 86400" combined
```

Tuning SAS Infrastructure Data Server

Overview

Note: This information does not apply to a programming-only deployment.

Note: This information applies to Cloud Foundry and Linux.

SAS Infrastructure Data Server provides a transactional store that is used to support SAS Viya. The server is configured automatically during deployment. However, to optimize its performance, it is recommended that you perform the tuning recommendations in this section.

SAS Infrastructure Data Server Tuning Recommendations

In SAS Environment Manager, edit the **SAS Infrastructure Data Server** service and modify the following properties:

Configuration Instance	Property	Value
sas.dataserver.conf: common	max_connections	1027
sas.dataserver.conf: common	max_prepared_transactions	1027
sas.dataserver.pool: common	num_init_children	1024

See Also

[“Edit Configuration Instances” on page 215](#)

Tuning SAS Message Broker

Overview

SAS Message Broker, which is based on RabbitMQ, is an intermediary program that converts messages from the protocol of the sender of the message to the protocol of the receiver. The server is configured automatically during deployment. However, to optimize its performance, it is recommended that you perform the tuning recommendations in this section.

SAS Message Broker Tuning Recommendations

Memory Allocation

By default, SAS Message Broker is configured to use up to 40% of the physical RAM on the machine on which an instance runs. This value does not guarantee that more than 40% will be used, but it sets a threshold at which publishers are throttled (notified to slow down message sending). You must decide what percentage of memory to dedicate to the message broker.

For example, if your system has 250 GB and you want to dedicate 50 GB to SAS Message Broker, use the following calculation to begin throttling back at 40% of the dedicated memory:

$$(0.4 * 50 \text{ GB}) / 250 \text{ GB} = 0.08 \sim 0.10$$

In the above example, you start throttling back the message broker when it has consumed more than 10% of the available memory. It is difficult to determine the value that the memory threshold, **vm_memory_high_watermark**, should be set to on a system where SAS Message Broker is sharing resources with other services. When the threshold is reached, producers are blocked from sending additional messages until used memory falls below this threshold again. Alternatively, an absolute limit high watermark might be set. However, this value must be less than the amount of available RAM. Otherwise, the message broker will not start.

To set the memory threshold for SAS Message Broker, complete the following tasks:

- 1 Set the **vm_memory_high_watermark** parameter by editing one of the following files:
 - If your environment is enabled for Transport Layer Security (TLS), edit the `/opt/sas/deploymentId/config/etc/rabbitmq-server/rabbitmq.config.ssl` file.
 - If your environment is not enabled for TLS, edit the `/opt/sas/deploymentId/config/etc/rabbitmq-server/rabbitmq.config.tcp` file.
- 2 Specify the following in the configuration file:

```
vm_memory_high_watermark, percentRAM
```

For more information, see [Configuring the Memory Threshold](#).

Disk Space Allocation

By default, SAS Message Broker requires at least 50 MB of free disk space to operate. If this threshold is reached, SAS Message Broker slows down message sending and blocks connections. Therefore, it is recommended that you set the minimum free disk size to the amount of memory that is installed on the machine, if it is available. By configuring a large amount of free disk space, a constrained system is more likely to recover under heavy usage scenarios by providing adequate space for paging considerations. By default, paging of transient messages, which are written to the disk under high memory consumption, starts when the system gets halfway to the **vm_memory_high_watermark**.

To set the free disk size, complete the following tasks:

- 1 Set the **disk_free_limit** parameter by editing one of the following files:
 - If your environment is enabled for Transport Layer Security (TLS), edit the `/opt/sas/deploymentId/config/etc/rabbitmq-server/rabbitmq.config.ssl` file.
 - If your environment is not enabled for TLS, edit the `/opt/sas/deploymentId/config/etc/rabbitmq-server/rabbitmq.config.tcp` file.
- 2 Specify the following in the configuration file:

```
disk_free_limit, {mem_relative, 1.0}
```

For more information, see [Configuring the Disk Free Space Limit](#).

Note: Using a disk space setting that is relative to the memory size assumes that the available disk space is greater than the amount of available memory.

Tuning the Linux Operating System

Overview

Note: This information does not apply to Cloud Foundry.

There are a number of configuration changes and variables that you can set to tune the SAS Viya environment for your performance and scalability needs. The following sections show how to configure the settings that are relevant to SAS Viya post-deployment. For information about tuning Linux during deployment, see [Perform Linux Tuning](#).

Linux Tuning Recommendations

Tuning TCP/IP

- Ensure that IPv6 is enabled.
- Permanently set the SAS recommended TCP/IP settings by using the following commands:

```
/sbin/sysctl -w net.ipv4.tcp_fin_timeout=30
/sbin/sysctl -w net.core.netdev_max_backlog=3000
/sbin/sysctl -w net.core.somaxconn=3000
/sbin/sysctl -w net.ipv4.tcp_keepalive_intvl=15
/sbin/sysctl -w net.ipv4.tcp_keepalive_probes=5
```

Tuning for SAS Studio

The following options can be modified in the `/etc/sysctl.conf` file, when these conditions exist:

- For sites with upward of 40 concurrently logged-on users, who are running tasks that require rendering of graphs, the SEMMNI parameter should be increased to 4096.
- For sites with upward of 600 logged-on users, increase the PID_MAX parameter to 131072.

