Secure QR Code Authentication Version 1.0

Committee Specification 01

01 July 2022

This stage:
https://docs.oasis-open.org/esat/sqrap/v1.0/cs01/sqrap-v1.0-cs01.docx (Authoritative)
https://docs.oasis-open.org/esat/sqrap/v1.0/cs01/sqrap-v1.0-cs01.html
https://docs.oasis-open.org/esat/sqrap/v1.0/cs01/sqrap-v1.0-cs01.pdf

Previous stage:
https://docs.oasis-open.org/esat/sqrap/v1.0/csd01/sqrap-v1.0-csd01.docx (Authoritative)
https://docs.oasis-open.org/esat/sqrap/v1.0/csd01/sqrap-v1.0-csd01.html
https://docs.oasis-open.org/esat/sqrap/v1.0/csd01/sqrap-v1.0-csd01.pdf

Latest stage:
https://docs.oasis-open.org/esat/sqrap/v1.0/sqrap-v1.0.docx (Authoritative)
https://docs.oasis-open.org/esat/sqrap/v1.0/sqrap-v1.0.html
https://docs.oasis-open.org/esat/sqrap/v1.0/sqrap-v1.0.pdf

Technical Committee:
OASIS Electronic Secure Authentication (ESAT) TC

Chairs:
David Kopack (d@trusona.com), Trusona, Inc.
Bojan Simic (bojan@hypr.com), HYPR Corp

Editor:
Don Sheppard (don@concon.com), Individual member

Abstract:
This document describes the use of QR Codes and a mobile phone as a replacement for a username and password in user login authentication. An alternative to passwords that includes QR Codes is described, and typical use cases are described.

This document also provides an overview and context for using QR Codes for security purposes.

This document specifies a “Secure QR Code Authentication Protocol” (SQRAP) and assesses the related security threats and risks.

Status:
This document was last revised or approved by the OASIS Electronic Secure Authentication (ESAT) TC on the above date. The level of approval is also listed above. Check the "Latest stage" location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=esat#technical.

TC members should send comments on this document to the TC's email list. Others should send comments to the TC's public comment list, after subscribing to it by following the instructions at the "Send A Comment" button on the TC's web page at https://www.oasis-open.org/committees/esat/.

Key words:

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] and [RFC8174] when, and only when, they appear in all capitals, as shown here.

Citation format:

When referencing this document, the following citation format should be used:

[SQRAP-v1.0]

Secure QR Code Authentication Version 1.0. Edited by Don Sheppard. 01 July 2022. OASIS Committee Specification 01. https://docs.oasis-open.org/esat/sqrap/v1.0/cs01/sqrap-v1.0-cs01.html. Latest stage: https://docs.oasis-open.org/esat/sqrap/v1.0/sqrap-v1.0.html.

Notices:

Table of Contents

1 Introduction

[Informative]

For online systems, authentication is the process of verifying the claimed identity of a user who desires access to one or more systems or resources. Authentication of a user's identity is critical for all systems that require verifiable authenticity. A challenge facing the IT industry today is the weakness inherent in traditional passwords when used for authentication.

User authentication is typically based on tokens referencing "something you have" (e.g., a physical USB key or dongle), "something you are" (e.g., facial recognition or a fingerprint) or "something you know" (e.g., a password). Strong authentication systems can treat a QR Code as something you have to provide protection against account take-over and identity theft.

A Quick Response Code, usually referred to as a QR Code, is a technique for encoding data into a two-dimensional visual representation. A QR Code can be scanned and its contents (i.e., its payload) can be interpreted by readily available devices including many mobile phones. The QR Code reader extracts the payload and uses it to perform functions such as display a web page, start an application or play a video. QR Codes are generally considered to be convenient, easy-to-use and resilient (for example, a QR Code can still be read even if up to 30% of the visual has been corrupted).

This standard supports trusted online transactions by establishing a general framework for using QR Codes for "no password" authentication. This standard specifies the message flow for a **Secure QR Code Authentication Protocol** (SQRAP) and provides an analysis of potential security threats and risks for specific use cases.

This standard does not advocate the use of QR Codes to exchange shared secrets; rather, the QR Code is used as a means of transport for public/private keys that bind a user identity to an authenticator application. In this technique the payload of the QR Code does not include personally identifiable information (PII). When the QR Code is scanned, an identifier representing the claimant is delivered through a secondary channel (also called a back channel) to the relying party. The claimant can be challenged for user presence via the operating system security of the authenticator device.

This standard also includes:

- An explanation of the role of user authentication within the identity management lifecycle and how it differs from authorization
- Methods that online relying parties and service providers currently use for electronic identity authentication
- A comparison of the methods currently in use and planned for authentication without static credentials or passwords, and with increasing levels of identity assurance, risk mitigation, and authentication certainty
- Reference information on the no-shared-secret authentication techniques and risk mitigation techniques being standardized, marketed, and implemented in the public or private sector.

1.1 Glossary

1.1.1 Definitions of terms

The following terms are used in this standard:

Term	Definition	Source
access control	A procedure by which an administrator can restrict access to resources, facilities, services, or information based on	ITU-T X.1252

	pre-established rules and specific rights or authority associated with the requesting party.	
assertion	A statement made by an entity without accompanying evidence of its validity. NOTE 1: The terms assertion and claim [noun] are agreed to be very similar.	ITU-T X.1252
assurance	See authentication assurance and identity assurance.	ITU-T X.1252
assurance level	A level of confidence in the binding between an entity and the presented identity information.	ITU-T X.1252
attribute	Information bound to an entity that specifies a characteristic of the entity.	ITU-T X.1252
attribute type	That component of an attribute which indicates the class of information given by that attribute.	ITU-T X.501
attribute value	A particular instance of the class of information indicated by an attribute type.	ITU-T X.501
authentication	Formalized process of verification that, if successful, results in a confirmation of claimed identity for an entity. NOTE 1: Use of the term authentication in an identity management context is taken to mean entity authentication.	ISO/IEC 24760-1
authentication assurance	Positive acknowledgement in the authentication process intended to provide confidence that the communication partner is the entity that it claims to be or is expected to be. NOTE 1: The assurance is based on the degree of confidence in the binding between the communicating entity and the identity that is presented.	ITU-T X.1252
authentication challenge and response	A method of protecting against authentication replay attack. Note 1: For example, if entity A wants to obtain a new message from entity B, it can first send a challenge in the form of a nonce (e.g., a cryptographic value that is used only once) to B. A then receives a response from B, based on the nonce that proves B was the intended recipient.	ITU-T X.1124
authentication protocol	Defined sequence of messages between an authenticator and an authentication service.	New
authentication service	An entity that verifies the claimant's identity by verifying the claimant's possession and control of one or two authenticators using an authentication protocol. To do this, the verifier may also need to validate credentials that link the authenticator(s) to the subscriber's identifier and check their status.	Derived from: NIST SP800-63 Appendix A
authenticator	Something the claimant possesses and controls (typically a cryptographic module or password) that is used to authenticate the claimant's identity. *(Previously referred to as a "token")*	NIST SP800-63 Appendix A
authorization	The granting of rights and, based on these rights, the granting of access.	ITU-T X.1252
certificate	A set of security-relevant data issued by a security authority or a trusted third party, that, together with security	ITU-T X.1252

	information, is used to provide the integrity and data origin authentication services for the data	
claim	[*noun*] Digital assertion about identity attributes made by an entity about itself or another entity. [*verb*] To state as being the case, without being able to give proof.	ITU-T X.1252
claim definition	A machine-readable definition of the semantic structure of a claim. NOTE 1: Claim definitions facilitate interoperability of claims and proofs across multiple issuers, holders, and relying parties.	ITU-T X.1252
claimant	An entity that is or represents a principal for the purposes of authentication. NOTE 1: A claimant includes the functions necessary for engaging in authentication exchanges on behalf of a principal.	ITU-T X.1252
claimed identity	An applicant's declaration of unvalidated and unverified personal attributes.	NIST SP800-63 Appendix A
credential	A set of data presented as evidence of a claimed identity and/or entitlements.	ITU-T X.1252
deep link	A hypertext link to a page on a website or web application other than the home page. Note 1: Deep refers to the depth of the page in the website's hierarchical structure of pages. Any page below the top page in the hierarchy (the home page) can be considered to be deep.	--
domain	An environment in which an entity can use a set of attributes for identification and other purposes.	ITU-T X.1252
enrollment	The process of inauguration of an entity into a context. NOTE 1: Enrollment may include verification of the entity's identity and establishment of a contextual identity. NOTE 2: Also, enrollment is a pre-requisite for registration. In many cases, the latter is used to describe both processes.	ITU-T X.1252
entity	Something that has separate and distinct existence and that can be identified in context. NOTE 1: An entity can have a physical or logical embodiment. NOTE 2: An entity can be a physical person, an animal, a juridical person, an organization, an active or passive thing, a device, a software application, a service, etc., or a group of these entities. In the context of telecommunications, examples of entities include access points, subscribers, users, network elements, networks, software applications, services and devices, and interfaces.	ITU-T X.1252
federation assurance level	A category describing the assertion protocol used by the federation to communicate authentication and attribute information (if applicable) to an RP.	NIST SP800-63 Appendix A

identifier (ID)	One or more attributes that uniquely characterize an entity in a specific context.	ITU-T X.1254
identity assurance	The confidence provided in the process of identity validation and verification used to establish the identity of the entity to which the credential was issued, and the degree of confidence that the entity that uses the credential is that entity or the entity to which the credential was issued or assigned.	ITU-T X.1252
identity verification	The process of confirming that a claimed identity is correct by comparing the offered claims of identity with previously proven information.	ITU-T X.1252
passwordless authentication	A method of user authentication that does not require the use of a secret password. Note 1: Implementations are typically driven by public key cryptography leveraging a user's possession of a device and a user's biometric authenticator on the device.	--
QR Code or Quick response code	Two-dimensional machine-readable optical symbol	ISO TS22691: 2021
static QR Code	A QR Code that represents fixed information. Note 1: The information that a static QR Code delivers upon being scanned is encoded directly in the QR Code itself.	--
dynamic QR Code	A QR Code that represents a short URL which can redirect the user to a destination website. Note 1: The destination URL can be changed after the QR Code has been generated, while the short URL embedded in the QR Code remains the same.	--
registration	The process in which an entity requests and is assigned privileges to use a service or resource. NOTE 1: Enrollment is a pre-requisite for registration. Enrollment and registration functions may be combined or separate.	ITU-T X.1252
relying party (RP)	An entity that relies on an identity representation or claim by a requesting or asserting entity within some request context.	ITU-T X.1252
security domain	A set of elements, a security policy, a security authority, and a set of security-relevant activities in which the elements are managed in accordance with the security policy.	ITU-T X.1252
auth-session-ID	An identifier used by the SQRAP authentication protocol.	New
session-ID	An identifier used in an authenticated user's transaction.	New
trust	The confidence of one party or entity that another party or entity will behave in a well- defined way that does not violate agreed-upon rules, policies, or legal clauses of the identity management system.	ITU-T X.1252
trust level	A consistent, quantifiable measure of reliance on the character, ability, strength or truth of someone or something.	ITU-T X.1252

user	Any entity that makes use of a resource, e.g., system, equipment, terminal, process, application, or corporate network.	ITU-T X.1252
verification	Process of establishing that identity information associated with a particular entity is correct. NOTE 1: The process of identification applies verification to claimed or observed attributes. NOTE 2: Verification of (identity) information may encompass examination with respect to validity, correct source, original, (unaltered), correctness, binding to the entity, etc. NOTE 3: Information is correct at the time of verification.	ITU-T X.1252

1.1.2 Acronyms and abbreviations

Item	Description
AD	Access Device
AP	Authentication Party
ATM	Automated Teller Machine
CA	Certification Authority
DNS	Domain Name Server
FIDO	Fast Identity Online
HTTP(S)	Hypertext Transfer Protocol (Secure socket)
ICAM	Identity Credential and Access Management
ID	Identity
IP	Internet Protocol
Ipsec	Internet Protocol Security
MFA	Multi-Factor Authentication
MP	Mobile Phone
OTP	One-Time Password
PC	Personal Computer
PII	Personally Identifiable Information
PIN	Personal Identification Number
QR (Code)	Quick Response (Code)
QP	QR Code Party
RP	Relying Party
SMS	Short Message Service
SQRAP	Secure QR Code Authentication Protocol
TCP	Transmission Control Protocol

USB	Universal Serial Bus
URI	Uniform Resource Identifier
UTC	Coordinated Universal Time
ZTA	Zero Trust Architecture

2 Background

[Informative]

2.1 General

A digital identity is a unique digital representation of an entity that is engaged in an online transaction. A digital identity, which consists of a set of attributes related to the entity, is considered to be valid when the attributes are verified to a specified level of assurance.

A digital identity lifecycle typically includes:

- **Identity enrollment**: the process of inauguration of an entity into a context; the entity is provided with an identity and is registered to an application
- **Authentication**: process of verification that, if successful, results in a confirmation of the claimed identity for an entity
- **Authorization**: an authenticated entity is granted rights that define what it is permitted to do
- **Access control**: the entity is granted access based on the authentication and authorization results
- **Identity deletion** (de-enrollment): the identity is removed and no longer represents an entity.

Within the lifecycle, entity authentication is a critical process because it establishes the level of trust between the communicating entities.

Methods for generating a digital identity and registering it within an application context are out of scope for this standard. This standard assumes that the digital identity has been successfully enrolled prior to the start of the authentication process.

The focus of this standard is on the use of QR Codes to facilitate authentication without passwords.

2.2 Authentication

2.2.1 Authentication purpose

At its most basic, authentication is the act of validating that a digital identity represents a specific entity within the context of an application or a system. Authentication confirms that the referenced entity actually exists, that it is indeed the entity that is requesting access, and that any identity-related information is valid and verifiably correct.

Entities that can be anything that has a separate and distinct existence and can be identified in context. Entities can include a physical person (i.e., a user), a legal person, an organization, an active or passive thing, a device, a software application, a service, etc., or a group of these entities.

Authentication is not equivalent to or dependent on authorization, which is the granting of rights and, based on these rights, the granting of access to an application or system. Authorization assumes a positive authentication result has been received.

2.2.2 Authentication assurance

Entity validation decisions can be a judgement based on how closely the digital identity information matches the pre-defined entity attributes. Verification of identity information may include examination with respect to validity, correct source, origin, correctness, binding to the entity, etc.

Authentication assurance is intended to provide confidence that the communication partner is the entity that it claims to be or is expected to be. Assurance is based on the degree of confidence in the binding between the communicating entity and the identity that is presented.

Three types of assurance contribute to establishing trust in a digital identity:

- **Identity assurance** consists of processes to verify an entity's association with their real-world identity

- **Authentication assurance** establishes that an entity attempting to access a digital service is in control of the technologies used to authenticate
- **Federation assurance** consists of process(es) to communicate, protect, and validate identity assertions being provided across different security domains. Identity federation is the sharing of online identity and authentication information between two or more parties.

Authentication assurance provides confidence that the digital identity with which one is interacting is consistent with the claimed identity. It uses processes to verify that a claimed identity is the same as the one that participated in the enrollment process and that it has previously been authenticated by the system.

Assurance is based on the degree of confidence in the binding between the communicating entity and the identity that is presented.

2.2.3 Authentication factors

Access to a system has traditionally been controlled with a single factor solution using only a username and password. The password had to be a “shared secret” known only to the user and the target application (in theory, at least). The password also needed to be both difficult for an attacker to guess or to infer and easy for the user to remember.

Simple username/password combinations no longer provide a sufficiently secure authentication solution for today’s online environments. Users tend to re-use passwords across accounts and hackers have raided many identity stores, resulting in user passwords being easily discovered. Enterprises attempt to address this problem by developing stronger password policies which only adds to user frustration without significantly enhancing authentication assurance.

Stronger authentication solutions often use a second factor (the password is the first factor), preferably from a different category, to enhance the authentication process security. The following are the typical categories:

- **Something you have**, such as a USB stick, a card, a token, or a mobile phone
- **Something you know**, for example a password, a PIN, or an answer to a question
- **Something you are**, including a fingerprint, a face, an iris, or a voice
- **Something you do**, for example, being at a location, using a specific device or network access point, or even a user’s habits (such as keying style or gestures).

The second factor is usually sent to an enrolled user device as a means to validate the user identity. For example, a security code can be sent to a user email or phone number to be transcribed to the login screen. These methods can be hacked through various phishing scams.

A QR Code, although not an authentication factor by itself, does provide a secure method for transferring authentication data between a user device (such as a PC) and a mobile phone.

2.2.4 Authentication without passwords

Passwordless authentication is, as its name implies, a method for verifying an entity without requiring one of the factors to be a password (or any other shared secret).

Current passwordless solutions require the entity to enter an application account identifier such as a user ID, a username, or a public identifier. The process then proceeds to complete the entity authentication process using a registered device such as a mobile phone.

Passwordless authentication typically relies on public-key cryptography. A public key is provided to the authentication service during enrollment while the private key is kept securely on the user’s device (a PC, mobile phone or external security token). The private key can only be accessed by providing a biometric signature or another authentication factor which isn't knowledge-based.

Passwordless authentication should not be confused with multi-factor authentication (MFA), even though both use a wide variety of authentication factors. MFA uses an added layer of security on top of

password-based authentication, while passwordless authentication doesn't require a memorized secret at all.

Passwordless authentication most often uses just one highly secure factor to authenticate identity, making it faster and simpler for users. It can, however, be combined with the idea of MFA if the authentication flow is both passwordless and uses multiple factors.

2.2.5 Authentication as a zero-trust function

Trust is the confidence of one entity that another entity will behave in a well-defined way that does not violate agreed-upon rules, policies, or legal agreements. Authentication is used to increase trust in the digital identity of an entity.

A basic principle of the zero-trust approach is to “trust but verify” each transaction regardless of its source or destination. This can be applied to a user’s device, the user themselves, the user’s environment, the transaction, the application, and the resources being used by the application.

For a given transaction, mutual authentication of the communicating entities may be required.

The NIST Zero Trust Architecture (SP 800-207) states:

> *“All resource authentication and authorization are dynamic and strictly enforced before access is allowed. This is a constant cycle of obtaining access, scanning and assessing threats, adapting, and continually reevaluating trust in ongoing communication. An enterprise implementing a ZTA would be expected to have Identity, Credential, and Access Management (ICAM) and asset management systems in place. This includes the use of multifactor authentication (MFA) for access to some or all enterprise resources. Continual monitoring with possible reauthentication and reauthorization occurs throughout user transactions, as defined and enforced by policy (e.g., time-based, new resource requested, resource modification, anomalous subject activity detected) that strives to achieve a balance of security, availability, usability, and cost-efficiency.”*

2.3 QR Code Overview

2.3.1 QR Code technology

Appendix D provides a more detailed review of QR Code technology.

The term “QR Code” stands for Quick Response Code. It is analogous to a retail barcode but is two-dimensional. QR Codes were initially developed in 1994 to track vehicles during manufacturing; they were designed to allow high-speed component scanning.

ISO/IEC 18004:2015 defines QR Code requirements and specifies the QR Code symbology characteristics, data character encoding methods, symbol formats, dimensional characteristics, error correction rules, reference decoding algorithm, production quality requirements, and user-selectable application parameters. Several forms of QR Code have now been standardized.

A simple example of a QR Code that points to an OASIS web page is illustrated in Figure 1.

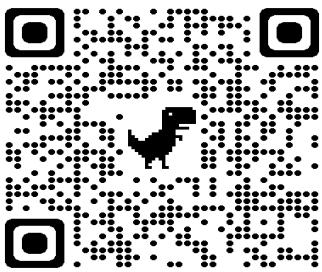

Figure 1: Sample QR Code

QR Codes are displayed in a matrix form. They consist of an array of nominally square modules arranged in an overall square pattern. A unique finder pattern, located at three corners of the symbol, helps make it easy to locate the QR Code position, size, and inclination.

QR Codes form a syntax for visual representation of a specified number of bits. As an analogy, the visual symbol “A” represents the eight binary bits “01000001”. QR Codes visually transfer a specific amount of data from a QR Code presenter to one or more QR Code readers without using data communications. The pictorial format of a QR Code is not easily read, memorized or decoded by humans.

QR Codes are useful when an entity needs to authenticate on a device that has a screen, but typing is not convenient, is impractical, or is too difficult. Examples of such devices include smart televisions, banking machines and public kiosks. QR Code authentication is also suited in situations that require touchless (or very low touch) interactions.

Additional benefits of QR Code authentication include an improved overall user experience and reduced administration costs. Stakeholders who benefit from QR Code authentication include end users (by eliminating passwords), service desks (by reducing password resets), security managers (through more refined access control) and system developers (through strong and consistent methods of verifying identities).

2.3.2 QR Code use cases

Appendix E provides a more detailed description of the use cases.

2.3.2.1 Use Case 1 – User login to a web application

The message sequences for Use Case 1 are provided in Clause 5.

A web-based server application offers the option to login to a pre-existing user account from a browser (i.e., a desktop, laptop, or kiosk browser) using a QR Code and a mobile phone.

The user has previously registered with the server application, has downloaded its mobile application, and has linked the mobile phone application to the server application.

The login process begins when the user navigates to the server application using the browser. The user then selects the QR Code login option.

The browser displays a QR Code that is specific to the login attempt. The user scans the displayed QR Code with the mobile phone. The QR Code is decoded, and the mobile phone application verifies that the QR Code was actually sent by the correct server application. If the session times out, the browser reverts back to the home page.

If the login process is successful, the server application is now accessible, subject to any authorization constraints that may apply.

No passwords need to be remembered or entered manually by the User.

2.3.2.2 Use Case 2 - User login to a web application using a pre-established QR Code

A web-based server application offers the option to login to a pre-existing user account from a browser (i.e., a desktop, laptop, or kiosk browser) using a QR Code instead of a password.

The user has previously registered with the server application but does not have access to a mobile phone application. The user has previously received a secure QR Code which is saved on the browser device.

The user also has a verified email or a phone number that can receive an SMS message with a one-time password (OTP).

The login process begins when the user navigates to the server application using the browser. The user then selects the QR Code login option.

The user's pre-established QR Code is pasted into the browser. The server application forwards an OTP to the email or SMS, which is then entered into the browser. If the session times out, the browser reverts back to the home page.

If the login process is successful, the server application is now accessible, subject to any authorization constraints that may apply.

No secret passwords need to be remembered or entered manually by the user.

NOTE: If the user wants to login to the server application from a different device, the user is required to transfer the QR Code to the new device.

2.3.2.3 Use Case 3 – QR Code on mobile application without a browser

A user wishes to access a server application via a device that does not include a web browser. One example would be accessing a bank account via an automated teller machine (ATM), assuming it is not browser-based. Opening a restricted door could be another example.

The user has a mobile application available that can either:

- generate a QR Code directly for display on the mobile device, or
- retrieve a QR Code from the back-end application and store it on the mobile device.

The user presents the QR Code on their mobile device to the reader device which reads it and then communicates with the server application to verify the QR Code. For some applications it may not be necessary to re-check validity for each use of the QR Code, leading to a simplified process.

Access to the server application via the reader device is then granted or denied.

2.3.2.4 Use Case 4 – Third-party QR Code with reader on device

A recognized authority issues a QR Code to a user to verify a status (e.g., for a vaccination). The status may represent a point in time (e.g., when the QR Code was issued) and is not necessarily current.

Note: A dynamic QR Code could allow realtime updates to the status if the user device has access to a network connection.

A third party (an approver) may need to verify the certificate. The approver has a QR Code reader. The QR Code is either displayed on the user's mobile device or is presented by some other means (e.g., on a card or on paper).

The approver reads the QR Code and obtains confirmation that the QR Code:

- was issued to the claimed user
- was supplied by the correct authority
- states the user's up-to-date status.

The QR Code does not need online interactions to supply the status. However, if the status can change, expire, or be revoked (e.g., due to re-infection of a vaccinated person), then real-time re-verification by the approver would be highly desirable or may even be mandatory.

Using the QR Code data, the approver is able to answer two questions:

- whether the entity presenting the QR Code is the claimed user (i.e., the actual owner of the mobile device and the QR Code)

3 Authentication Reference Model

[Normative]

3.1 General

This authentication reference model uses a QR Code to transfer authentication data from a Client's Web Access Device (e.g., a web browser display) to the Client's Authenticator (e.g., a mobile device) with strong assurance that it is not readily copied. The QR Code is specific to a login interaction and is not considered to be an authentication factor by itself.

A Relying Party mobile application SHALL be pre-installed on the Client's mobile device. This reference model does not include Client registration or enrollment functions.

The mobile device SHALL be sufficiently close to the web browser to allow the QR Code to be reliably read.

Both the web browser and the mobile device SHALL be able to connect to the Relying Party with acceptable service quality. The web browser and the mobile device SHALL use independent communications paths to reach the Relying Party.

Implementations of this authentication reference model SHALL NOT include authorization or access control processes.

Other functions associated with authentication assurance are out of scope for this standard.

3.2 Parties and roles

Implementations of the QR Code authentication reference model SHALL include four parties (logical or physical):

- The **Client Party** is the user that requires access to an application or resource. The Client Party includes two roles:
  - the user interface device, typically a browser-enabled device
  - a mobile device application that can read a QR Code and serve as the authenticator.
- The **Relying Party** is the application that is the target of the login request; typically, it is a web-based server application. The Relying Party assumes the role of service provider and has:
  - a web-based server application that can display QR Codes
  - a mobile device application that links to or contains a QR Code scanner/reader.
- The **Authentication Party** is an authority that the Relying Party trusts to verify the Client Party.
- The **QR Code Party** generates QR Codes corresponding to the authentication-related data (and optionally implementation-specific data).

NOTE – The Relying Party MAY include the QR Code Party and Authentication Party.

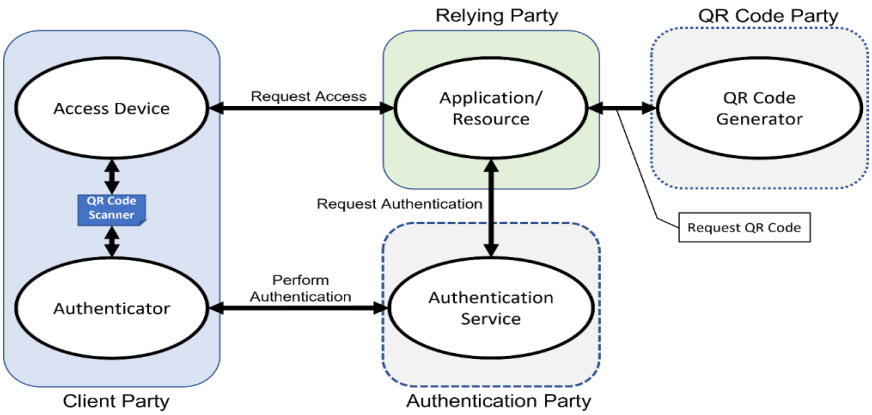

Figure 2: QR Code authentication reference model

In Figure 2 the Relying Party, the QR Code Party and the Authentication Party are shown as separate and distinct, but physical separation is not a requirement. For example, the Authentication Party could implement both the Authentication Service and a QR Code Generator.

3.3 Trust relationships

The relationship between the Relying Party and the Client Party, at the time of an authentication event, has a direct impact on the process for authentication and the scope of the potential threats to the authentication process.

Three scenarios are:

- **Known and registered users**: the Client Party has previously established a trusted relationship with the Relying Party and has identified itself before authentication begins. The Relying Party can identify, without user interactions (e.g., with a browser cookie), the user who is to be authenticated.
- **Unknown and registered users**: the Client Party has previously established a trusted relationship with the Relying Party but has not been identified. This requires user interaction (e.g., entering a username) during the authentication process to help the Relying Party determine who claims to be requesting authentication.
- **Unknown and unregistered users**: the Client Party has not yet established a trusted relationship with the Relying Party. The Client Party cannot complete the QR Code authentication process.

3.4 Dual channel model

In the QR Code authentication reference model of Figure 2, two separate communication channels are used:

- the primary channel is the connection between the access device and the application/resource
- the secondary channel is the connection between the Authenticator and the Authentication Service.

Implementations of the QR Code authentication reference model SHALL include separate connections between:

- the Relaying Party and the Access Device
- the Relying Party and the Authenticator.

4 Secure QR Code Authentication Protocol

[Normative]

4.1 General

The Secure QR Code Authentication Protocol (SQRAP) authenticates a Client Party (i.e., a user) for the purpose of logging on to a Relying Party application or resource without requiring a shared secret such as a pre-established, stored password. SQRAP is a “passwordless” authentication protocol.

SQRAP uses a QR Code as the bridge between the Access Device (i.e., the web browser) and the Authenticator (i.e., the user’s trusted Mobile Device). The Mobile Device is considered to be an authentication factor (i.e., “something you have”) while the QR Code by itself SHALL NOT be considered as an authentication factor.

The SQRAP protocol SHOULD NOT be assumed to guarantee compatibility among different implementations.

This standard does not specify the actions that are to be taken if a process failure or protocol violation occurs. Examples include when the user does not scan the QR Code within a specified time, or the Mobile Phone cannot connect to the relying party.

4.2 Protocol states

The SQRAP protocol SHOULD include five states:

- **Awaiting login request**: The Relying Party is awaiting a login request from a Client Party (i.e., the user via the Access Device).
- **Login pending**: A login request has been received by the Relying Party and the authentication process has been initiated but has not completed.
- **QR Code displayed**: A QR Code has been generated, is displayed on the Access Device via the primary link and is ready to be scanned. The QR Code SHALL be delivered to the Access Device via the primary communications channel.
- **QR Code payload verified**: The Authentication Party has notified the Relying Party of the authentication decision. The Mobile Phone SHALL communicate with the Authentication Service via the secondary channel.
- **Login complete**: The Relying Party applies any access control policies, permissions, limitations, timers, etc. and displays a configured landing page on the Access Device.

The SQRAP protocol assumes the Client Party has a previously established relationship (e.g., an account) with the Relying Party application and has a validated identity. The Mobile Phone application is linked to the Relying Party application.

The SQRAP protocol SHALL use the QR Code as the only means for transferring coded data from the primary channel to the secondary channel. The QR Code is not considered to be a proof of identity.

4.3 Message sequences

Figure 3 specifies the flow of messages for the SQRAP protocol using the following terms to represent the parties described in Figure 2:

- **Mobile Phone (MP)** – a trusted authenticator that serves as “something you have”
- **Access device (AD)** – a device that is being used to login to the Relying Party; it is typically, but not necessarily, a web browser
- **Relying party (RP)** - the application or resource that is to be accessed

– **Authentication Service (AS)** – a function that verifies the user and generates a QR Code (i.e., includes the QR Generator

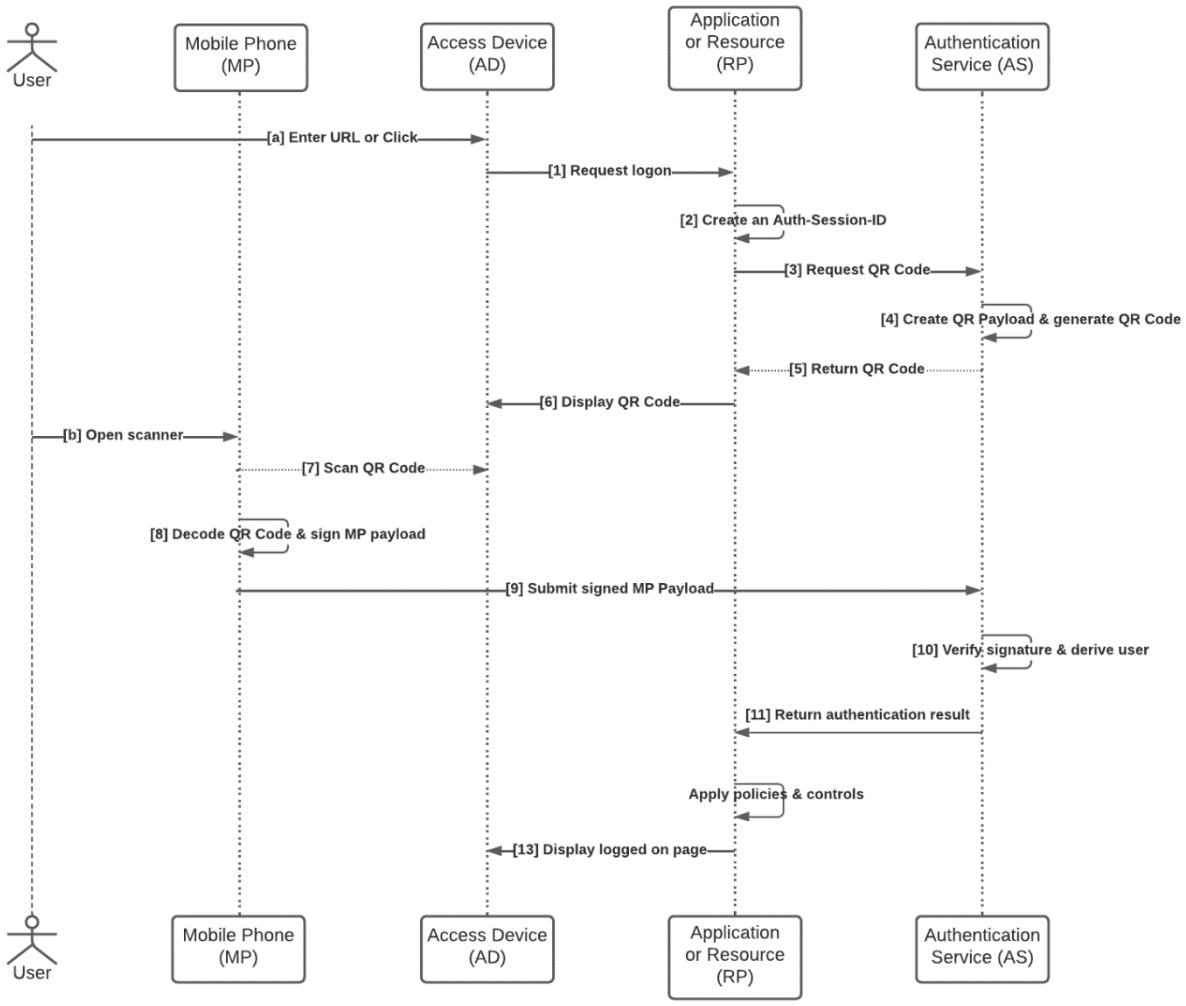

Figure 3: SQRAP message sequence

The RP and AS may be implemented as separate components or can be integrated into a single system. In Figure 3 the AS includes the QR Code Generator.

4.4 Message descriptions

The following describes each of the SQRAP messages that are identified in Figure 3. All messages and functions SHALL be included in implementations of SQRAP.

[a]	**Initiate login**
Source:	Client Party (user) action
Action:	The user SHALL enter a URL in a browser or click on an open web page
Options:	- a username or email address MAY be requested by the browser

[1]	**Request login**
Source:	AD to RP
Action:	This message requests a login to the RP. This message SHALL be generated as a result of [a].
Options:	- additional information such as a device-ID and/or a Username MAY be included in the request

[2]	**Create Auth-Session-ID**
Source:	RP function
Action:	The RP SHALL generate a unique auth-session-ID The RP SHALL establish an authentication session and bind the login request to the session. The auth-session-ID format and content is implementation specific but SHALL be compatible with standard QR Code data syntax and capacity requirements as specified in ISO/IEC 18004.
Options:	- a session timeout SHOULD be provided with a recommended timeout of 30 seconds - the auth-session-ID MAY be encrypted - The implementation SHOULD store the auth-session-ID securely

[3]	**Request QR Code**
Source:	RP to AS/QP
Action:	The RP SHALL deliver the auth-session-ID to the AS in the Request QR Code message.
Options:	- the RP MAY include proprietary, implementation-specific information - the RP SHOULD sign the request, especially if the AS is not physically co-located with the RP.

[4]	**Create QR Payload and Generate QR Code**
Source:	AS function
Action:	The AS SHALL produce the QR Payload. The QR Payload SHALL, at a minimum, include the auth-session-ID and MAY also include implementation-specific information. The QR Payload SHALL be suitable for encoding into a QR Code that is supported by the MP scanner.
Options:	- a time to live indicator MAY be included

[5]	**Return QR Code**
Source:	AS to RP
Action:	The AS SHALL forward the generated QR Code to the RP.
Options:	- the AS MAY sign the return message.

[6]	**Display QR Code**
Source:	AD function
Action:	The RP SHALL display the QR Code on the AD.
Options:	- the RP MAY close the page if a designated time-out occurs.

[b]	**Open scanner**
Source:	User action Note: this action may occur at any time prior to the QR Code time-out.
Action:	The User SHOULD open the QR Code scanner on the MP.
Options:	- the MP SHOULD use the built-in scanner to read the QR Code and automatically open the MP application. - the user MAY manually open the MP application that is linked to the RP application.

[7]	**Scan QR Code**
Source:	AD to MP
Action:	The QR Code displayed on the AD is scanned.
Options:	- the contents SHOULD be displayed on the MP screen prior to taking action

[8]	**Decode QR Code and sign MP Payload**
Source:	MP function
Action:	The MP receives and decodes the QR Code and the content of the QR Payload is extracted by the MP application. The MP SHALL generate and SHALL sign the MP Payload.
Options:	- the MP MAY add implementation-specific information to the MP Payload.

[9]	**Submit signed MP Payload**
Source:	MP to AS
Action:	The signed MP Payload SHALL be submitted to the AS.
Options:	- the MP Payload MAY be passed through the RP

[10]	**Verify signature and derive User**
Source:	AS function
Action:	The AS SHALL verify the MP Payload signature, thereby proving the source was the MP. The AS derives the user from information contained in the MP Payload.

	NOTE: The MP is assumed to have generated a keypair during registration and the AS has the public key stored. The signed MP Payload should include a signature + key-id which can be used to derive the User (and the public-key to validate the signature)
Options:	

[11]	**Return authentication result**
Source:	AS to RP
Action:	The AS SHALL return the authentication result (the assertion) to the RP. A successful authentication SHOULD result in the assertion of an identifier (pseudonymous or non-pseudonymous) to the RP.
Options:	- the AS MAY provide a reason for any negative result. - the AS MAY provide additional identity-related information to the RP.

[12]	**Apply policies (controls)**
Source:	RP function
Action:	The RP SHOULD create a new session-ID for an authenticated user based on the previously established login auth-session-ID and the MP Payload. The RP SHOULD invalidate the original auth-session-ID (as generated in [2]) to prevent session fixation and reuse attacks. The RP application SHALL configure the initial landing page for the user with the policies and controls that have been pre-determined for that user.
Options:	- the RP MAY keep track of the auth-session-IDs for auditing purposes.

[13}	**Display landing page**
Source:	RP to AD
Action:	The RP SHALL display the configured landing page. NOTE: When using a web browser, to minimize the possibility of successful phishing attacks, the AD should be re-directed to a pre-determined URL for the RP.
Options:	

4.5 Implementation guidance

The following serves as guidance for implementers of the SQRAP protocol.

To mitigate attacks related to re-using generated QR Codes:

- Each QR Code SHOULD contain a nonce value, which would be invalidated when the authentication process has been completed
- The QR Code displayed to the client SHOULD be cryptographically signed by an entity that is trusted by the RP application
- The QR Code SHOULD contain an expiration time, which is within reasonable time limits for user to complete the login process

– Service SHOULD make its best effort to conceal the User ID from the QR Code, as that could become a vector to other attacks against the user (e.g., brute force, spam, etc.)

If server-side session storage is available, the per-session auth-session-ID MAY be used to refer to data that is secured in server. Otherwise, the data SHOULD be encrypted using a per-session or per-device key.

Optionally, authentication processes MAY be enhanced using additional challenge-response mechanisms, such as asking the User for something they know.

5 SQRAP Threat Assessment

[Informative]

5.1 General

QR Codes are appealing targets for hackers who are aiming at mobile users, especially for social media QR Codes that can be replaced if they are static QR Codes or altered if they are dynamic QR Codes.

A QR Code is very difficult to compare to a known good reference code and, since it is impossible to remember or validate without an application, it has to be captured whether it is valid or not.

In most cases, MP users are expected to react quickly to commands and provide quick responses (due to time-outs) which tend to hinder their ability to examine and assess the information presented by the QR Code.

This clause briefly reviews various types of attack but is not meant to be a comprehensive listing of all possible QR Code threats.

5.2 Types of attack

5.2.1 Man in the middle

Man-in-the-Middle attacks occur when a hacker or compromised system sits in between two uncompromised people or systems and deciphers the information they're passing to each other, including passwords. If Alice and Bob are passing notes in class, but Jeremy has to relay those notes, Jeremy has the opportunity to be the man in the middle.

An attacker may lure an unsuspecting victim to a phishing site by acting as a “man in the middle.” After authentication is completed, the attacker can assume the identity of the victim. This is possible because the device scanning the QR Code cannot validate the origin of the QR Code.

In this type of attack, a hacker can place a fake QR Code sticker over a relying party legitimate QR Code. In some cases, the hacker can trick user to scan the fake QR Code through false advertisement on social media or printed magazines. Once a QR Code is scanned, the attacker will capture the victim’s credentials and then the hacker can either redirect the user back to the correct website with some kind of error message.

As a variation, the hacker can proceed to log one to the real website pretending to be the legitimate account owner. The hacker can forward the real website messages to the user and could perform additional illegal activities.

Even if a hacker or malicious program inserts itself into the interaction between users and applications and captures the information users enter, MFA would require users to supply credentials from a different device. This can prevent eavesdroppers from intercepting or manipulating communications between the user and application. Push-based authenticators such as mobile phone authenticators are well-suited to provide a secure MFA mechanism without inconveniencing users.

5.2.2 Credential stuffing

Credential stuffing takes advantage of accounts that never had their passwords changed after an account break-in. Hackers will try various combinations of former usernames and passwords, hoping the victim never changed them. credential stuffing attacks, in which cybercriminals automatically and simultaneously try a list of stolen usernames and passwords on multiple sites.

Elimination of passwords eliminates this form of attack.

5.2.3 Keyloggers

Keyloggers are a type of malicious software designed to track every keystroke and report it back to a hacker. Typically, a user will download the software believing it to be legitimate, only for it to install a keylogger without notice.

Cybercriminals install keyloggers on a victim's device, often via a virus. The program captures every keystroke the victim makes and records their usernames, passwords, answers to security questions, banking and credit card details, sites visited, and more. Cybercriminals then use this sensitive information for malicious purposes.

Use of QR Codes to transfer data from the primary display to the mobile phone eliminates the need for keying and therefore negates the danger of keylogging.

5.2.4 QR Code rebroadcast

Due to the machine-readable nature of a QR Code, it can be read and re-displayed by a machine. This allows an attacker to initiate an Authentication with a relying party, capture the QR Code, and rebroadcast the QR to a group of unsuspecting victims.

Without mitigation, this could allow an attacker to social engineer a victim into scanning the Authentication QR Code and completing the Authentication on behalf of the attacker. This results in the attacker being authenticated to the relying party as the victim.

This attack can be used at scale and targets multiple victims at once.

When the relying party has taken steps to mitigate the general QR Code rebroadcast threat, an attacker can still use a targeted attack against a single victim.

In this scenario, the attacker uses social engineering to guide the victim through a registration step while pretending to be relying party. Once known to the relying party, the attacker again uses social engineering to guide the victim through the Authentication process. This results in the attacker being Authenticated to the relying party as the victim.

This attack requires targeting an individual victim each time.

5.2.5 Phishing, spear phishing, whaling, etc.

Phishing is when a hacker posing as a trustworthy party sends you a fraudulent email, hoping you will reveal your personal information voluntarily. Sometimes they lead you to fake "reset your password" screens; other times, the links install malicious code on your device. In this attack, hackers can post, publish or even email QR Codes that entice people to scan them by claiming some rewards and privileges such as free WIFI access and a discount coupon. In this attack, the QR Code is used in a malicious fashion to direct the user to a bad site that can infect the device with Malware or Virus.

The above attack can easily be converted to a Spear Phishing attack via the special targeting of a specific individual or a particular group such as tourists or at a social gathering.

Here are a few examples of phishing:

- Regular phishing: You get an email from what looks like goodwebsite.com asking you to reset your password, but you didn't read closely and it's actually goodwobsite.com. You "reset your password" and the hacker steals your credentials.
- Spear phishing: A hacker targets you specifically with an email that appears to be from a friend, colleague, or associate. It has a brief, generic blurb ("Check out the invoice I attached and let me know if it makes sense.") and hopes you click on the malicious attachment.
- Smishing and vishing: You receive a text message (SMS phishing, or smishing) or phone call (voice phishing, or vishing) from a hacker who informs you that your account has been frozen or that fraud has been detected. You enter your account information, and the hacker steals it.

- Whaling: You or your organization receive an email purportedly from a senior figure in your company. You don't do your homework on the email's veracity and send sensitive information to a hacker.

An attacker may launch a phishing attack to steal a user's credentials. But, if the user's account is protected by MFA, the attacker won't be able to access it. This is because a phishing email won't provide the other authentication factors, such as one-time passwords (OTPs) sent to a different device (e.g. a mobile phone), fingerprints, or other biometric factors required to gain access to the system.

In attacks where the attacker tries to trick a user into entering their credentials, certain types of MFA such as WebAuthn require the user to enter a key or fingerprint from the system they are logging in from. These details cannot be captured by the attacker, thus protecting the system and user.

5.2.6 Brute force attack

If a password is equivalent to using a key to open a door, a brute force attack is using a battering ram. A hacker can try 2.18 trillion password/username combinations in 22 seconds, and if your password is simple, your account could be in the crosshairs.

An attacker may manage to find a working username and password with a brute force, reverse brute force attack, or dictionary attack. However, they don't know or have the other authentication factors required by the MFA system, so they cannot access the system.

In a *Brute Force attack*, the cybercriminal uses a program to generate and use many possible username/password combinations, hoping that at least one will help them gain access to an enterprise system. Brute force attacks are very common and provide many benefits to cybercriminals:

- Place spam ads on websites to make money when the ad is clicked or viewed
- Infect a site's visitors with activity-tracking spyware, steal their data, and sell it to marketers (or on the dark web)
- Hack into user accounts to steal personal data, financial data, or money
- Spread malware or hijack enterprise systems to disrupt operations

In a *reverse brute-force attack*, the attacker tries common passwords, e.g., "password" or "123456" to try to brute-force a username and gain access to many accounts.

A type of brute force attack, dictionary attacks rely on our habit of picking "basic" words as our password, the most common of which hackers have collated into "cracking dictionaries." More sophisticated dictionary attacks incorporate words that are personally important to you, like a birthplace, child's name, or pet's name.

5.2.7 Dictionary attack

Dictionary attacks are a common type of brute force attack, where the attacker works through a dictionary of possible passwords and tries them all to gain access.

A *credential stuffing attack* is a type of brute force attack that also takes advantage of passwords. Many people often use the same username and/or password on multiple accounts. Attackers take advantage of this fact to perpetrate credential stuffing attacks where they steal credentials and try to use them to access many accounts. Sometimes they may obtain credentials from one organization, either through a data breach or from the dark web and use them to access user accounts at another organization. They hope that at least some of the same credentials will enable them to:

- Sell access to compromised accounts
- Steal identities
- Perpetrate fraud
- Steal sensitive enterprise information, e.g., business secrets, Personally Identifiable Information (PII), financial information, intellectual property, etc.

— Spy on the enterprise (corporate espionage).

5.2.8 Implementation errors

In this form of an attack, a hacker can take advantage of errors that are the result from improper implementation of a QR solution from a legitimate relying party. The following errors can happen:

– Relying party forgets to register a domain name that is pointed to by the QR Code. A hacker can register the domain name and take over the official relying party service.

– Relying party mistype the domain name in a QR Code and a hacker registering the misspelled domain name and then using it to defraud users.

6 Conformance

[Normative]

SQRAP conformance is defined as conformance to a user authentication system that is comprised of all the architectural components of the QR Code authentication infrastructure and satisfies at least the minimum conformance requirements for each of the SQRAP reference architecture components.

A system implementing QR Code authentication that conforms to this document SHALL include the following components:

- a mobile authenticator
- an authentication service
- a QR Code generator
- a QR Code reader
- primary and secondary communication channels

The conformance items in clauses 3 and 4 apply to this document as follows:

1. This document is applicable to all specifications. In order to claim conformance to this document, all the requirements in clauses 3 and 4 SHALL be met.
2. This document SHALL be implemented in its entirety. It defines no profiles and no levels.
3. This document allows extensions. Extensions included in a conforming specification would address additional conformance issues and/or contain additional statements contributing to a clearer, more measurable, less ambiguous specification.
4. This document contains no discretionary items.
5. This document's normative language is English. Translation into other languages is permitted.

Appendix A. References

This appendix contains the normative and informative references that are used in this document.

While any hyperlinks included in this appendix were valid at the time of publication, OASIS cannot guarantee their long-term validity.

A.1 Normative References

The following documents are referenced in such a way that some or all of their content constitutes requirements of this document.

ISO/IEC 18004

Information technology — Automatic identification and data capture techniques — QR Code bar code symbology specification, https://www.iso.org/standard/62021.html

A.2 Informative References

The following referenced documents are not required for the application of this document but may assist the reader with regard to a particular subject area.

ISO/IEC 9798-1

Information technology — Security techniques — Entity authentication — Part 1: General https://www.iso.org/standard/53634.html

ISO/IEC TS 29003

Information technology — Security techniques — Identity proofing, https://www.iso.org/standard/62290.html

ISO/IEC 29115

Information technology — Security techniques — Entity authentication assurance framework, https://www.iso.org/standard/45138.html

ITU-T X.1252

Series X: Data Networks, Open System Communications and Security, Cyberspace security – Identity management, Baseline identity management terms and definitions, April 2021, https://www.itu.int/rec/T-REC-X.1252/en

ITU-T X.1254

Series X: Data Networks, Open System Communications and Security, Cyberspace security – Identity management, Entity authentication assurance framework, September 2020, https://www.itu.int/rec/T-REC-X.1254-202009-I/en

NIST Special Publication 800-63-3

Digital Identity Guidelines, June 2017, https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-3.pdf

NIST Special Publication 800-63B

Authentication and Lifecycle Management, June 2017, https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-63b.pdf

NIST Special Publication 800-207

Zero Trust Architecture, August 2020, https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207.pdf

FIDO Alliance Specifications Overview

https://fidoalliance.org/specifications/

QRCode.com

QRCode.com

Appendix B. Acknowledgments

[Informative]

B.1 Special Thanks

Substantial contributions to this document from the following individuals are gratefully acknowledged:

Abbie Barbir, Aetna – Secretary

Jason Burnett, Digital Trust Networks – Voting Member

Ori Eisen, Trusona – Voting Member

David Kopack, Trusona – Chair

Lauri Korts-Parn, NEC Corp. – Member

Clayton Lengel-Zigich, Trusona - Member

Ryan Rowcliffe, Hypr – Member

Bojan Simic, Hypr – Chair

Don Sheppard, Individual Member – Editor

Hiroshi Takechi, NEC Corp. – Voting Member

B.2 Participants

The following individuals were members of this Technical Committee during the creation of this document and their contributions are gratefully acknowledged:

Person	Organization	Role
Abbie Barbir	Aetna	Secretary
Cisa Kurian	Aetna	Member
Erick Verry	Aetna	Member
Drummond Reed	Evernym	Member
Ryan Rowcliffe	HYPR CORP	Member
Bojan Simic	HYPR CORP	Chair
Sander Fieten	Individual	Member
Detlef Huehnlein	Individual	Member
Andreas Kuehne	Individual	Member
John Sabo	Individual	Voting Member
Donald Sheppard	Individual	Member
Lauri Korts-Pärn	NEC Corporation	Member

Person	Organization	Role
Hiroshi Takechi	NEC Corporation	Voting Member
Duncan Sparrell	sFractal Consulting LLC	Member
Ori Eisen	Trusona, Inc.	Voting Member
David Kopack	Trusona, Inc.	Chair

Appendix C. Revision History

Revision	Date	Editor	Changes Made
1.0	May 2, 2022	D. Sheppard	Initial document

Appendix D. QR Code Technology

[Informative]

This annex provides an overview of QR Code technology and capabilities. This annex is based on content from DENSO WAVE (Qrcode.com) and ISO/IEC 18004:2015.

D.1 What is a QR Code

The term “QR Code” stands for Quick Response Code.

QR Codes are symbols that are displayed as in a matrix form. They consist of an array of nominally square modules arranged in an overall square pattern, including a unique finder pattern located at three corners of the symbol (in Micro QR Code symbols, at a single corner). The finder pattern assists in easy location of the QR Code position, size, and inclination. A QR Code can also be thought of as a two-dimensional bar code.

QR Codes form a syntax for visual representation of a specified number of bits. As an analogy – the symbol “A” visually represents the eight binary bits “01000001”. QR Codes visually transfer an amount of data from a presenter to one or more readers without using data communications. Four levels of error correction are available.

Module dimensions are user-specified to enable symbol production by a wide variety of techniques. There are four technically different, but closely related members of the QR Code family, which represent an evolutionary sequence:

(a) QR Code Model 1: the original specification for a QR Code (described in AIM ITS 97-001 International Symbology Specification-QR Code); this version consists of a grid of 21 X 21 modules with 133 modules available for storing encoded data.

(b) QR Code Model 2: an enhanced form with additional features (primarily the addition of alignment patterns to assist navigation in larger symbols) and was the basis of the first edition of ISO/IEC 18004.

(c) QR Code (the basis of the second edition of ISO/IEC 18004) is similar to QR Code Model 2; its format differs only in the addition of the facility for symbols to appear in a mirror image orientation for reflectance reversal (light symbols on dark backgrounds) and the option for specifying alternative character sets to the default.

(d) Micro QR Code: a variant of QR Code with a reduced number of overhead modules and a restricted range of sizes, which enables small to moderate amount of data to be represented in a small symbol (also specified in the second edition of ISO/IEC 18004).

There are over 40 different flavors of QR Code, each with different data and error rate capacities. QR Code Version 40, the largest QR Code version, can support up to 4296 alphanumeric characters.

QR Codes encode the data in two dimensions (vertical and horizontal). To extract the data, a device with a camera captures an image of the QR Code and then decodes it using QR Code reader software. QR Codes can be decoded even in the presence of errors due to the robust error correction capabilities.

D.2 What makes up a QR Code

Visually, a QR Code looks like a twisted crossword puzzle, but its design is crucial to its function. Here are some of its most important elements.

- **Position detection markers:** The prominent squares located in three corners of each code offer easier recognition and assist with reading the QR Code at high speed.
- **Alignment markers:** These can help straighten out codes placed on curved surfaces. It's smaller than a position detection marker but will become larger the more information a QR Code holds.
- **Timing pattern:** The black and white alternating modules configure the data grid and help the scanner calculate how large the data matrix is.
- **Version information:** This determines which of the 40 different QR Code versions is being used, with the most common versions being 1 to 7.
- **Format information:** This pattern holds information about the data mask pattern and error tolerance of the code, making it easier to scan.
- **Data and error correction keys:** The error correction function shares a structural space where all the data in a QR Code is contained. This correction block's mechanism is essential to allowing up to 30% of a code being read if damaged.
- **Quiet Zone:** This white space can be seen as the border of a QR Code to help improve comprehension for scanning and provide structure. It determines what is and isn't part of the code.

D.3 What do QR Codes do

QR Codes are a form of encoding for data in much the same way as ASCII encodes character sets such as the English alphabet. In addition to alphanumeric characters, QR Codes can encode binaries, Kanjis1 or control codes. By itself, a QR Code doesn't do anything other than encode data for display and scanning by a reader.

For example, a QR Code can be printed on a poster, displayed publicly, and then captured by a smartphone, thereby allowing immediate access to detailed descriptions or references that would not fit on the poster.

QR Codes can either be static or dynamic. A static QR Code is a QR Code with fixed information. Dynamic QR Codes are editable and offer more features than Static QR Codes. For example, the destination URL can be changed after the QR Code has been generated, while the short URL embedded in the code remains the same. Dynamic QR Codes are easier to scan than Static QR Codes because the QR Code image is less dense.

D.3.1 Static vs. dynamic QR Codes

QR Codes vary in design depending on the encoded data and function and can be categorized primarily in two ways: static and dynamic.

A static QR Code cannot be modified once it has been created. This is ideal for creating QR Codes in mass for an event. A drawback is its lack of creativity and that it may not allow for analytics on how many times the code may have been scanned. An example of a good static QR Code would be one for your Wi-Fi password.

Dynamic QR Codes allow you to change and edit the code as many times as you need. When the code is scanned, it redirects you to the URL contained inside. These codes offer the freedom to package your design, like adding contrasting colors. They also have the ability to track and measure advertising statistics.

These added insights allow the QR Code creator access to where and with what device the code was scanned. Along with adding in campaign information and resetting scans, all the results collected can be downloaded as comma-separated values or a CSV report.

D.4 When can they be used

QR Codes, which transfer an amount of data as a graphic, can be used in a wide range of situations, especially when data transfer via networking is not practical.

QR Codes can deliver:

- static information (e.g., a location name, a warning)
- pointers to websites (e.g., a link to a restaurant website)
- embedded functions (e.g., a timer or calculation function)
- dynamic or situation-dependent software apps

QR Codes are still used to track products and product information through a supply chain, but they are also used for so much more. You've likely used a QR Code to view a menu, link a social profile or add friends to an account, board a flight, download an application, send and receive payments, access Wi-Fi, and authenticate your login details. The possibilities with QR Codes are truly endless.

Denso Wave has also made some significant improvements to the code's design. Now QR Codes can come with brand protection, anti-forgery measures, and traceability, all features meant to improve the QR Code experience as they become increasingly more common across industries.

D.5 Where do you find QR Codes

QR Codes are an increasingly popular means for enhancing the customer experience in many situations, especially when health and safety may be a factor. Reading a QR Code on a smartphone is much easier than typing a URL for a specific website page.

Some examples of QR Code uses are:

Tracking assets – A QR Code can replace a barcode attached to an asset. Scanning devices are readily available and the QR Code can be read at various angles. A QR Code can operate normally even with only 30% of the QR Code intact while even slight damage to a barcode, such as a small tear or crease, can cause it to fail completely. QR Codes can store significantly more numerical data than a barcode (25 characters vs. 2500 numerical characters). More storage enables QR Code use for enhanced asset management processes such as location tracking, viewing maintenance and repair history, managing lifecycles, and building check-in/check-out capabilities.

Desktop login – Applications can use a QR Code with a smartphone app to verify a user login on a desktop or laptop browser (e.g., WhatsApp or LINE). The QR Code is shown on the login page and, when a registered user scans it with a verified smartphone, they will automatically be logged in. Authentication is performed by the smartphone which contacts the app server. This eliminated the need to remember passwords or authenticator numbers.

Digital data access – Libraries - As many things shift to online, we want to make accessing these resources easier for you. When you use your phone's camera or a QR reader app, the resource that you're interested in will launch in your phone's internet browser. No navigating our website or having to stop and ask. If you're not at the library, you will land on the page to enter your library card number and PIN/password for access. If you use the QR Code in the library, you'll be taken directly to the resource.

News and advertising – Newspapers and magazines can use a QR Code to point to references or more detailed information such as reports and in-depth analyses. QR Codes for print advertisements are an excellent way of sharing additional information without taking up too much space. Local television stations have also begun to utilize codes on local newscasts to allow viewers quicker access to stories or information mentioned in the newscasts overall.

Passports – QR Codes can be used for passports or other certificates including COVID-19 vaccination certificates. For example, QR Codes are being used as proof of vaccination for COVID-19 in various jurisdictions including California, Canada, Australia, and the European Union. A vaccine passport will typically include your name, date of birth, and COVID-19 vaccine history including which doses you got, and when you got them.

Contactless transactions – A smartphone display a QR Code that, when scanned, can open a door or other locked item; a bank ATM such as one NCR model can display a QR Code for smartphone-based access (which is similar to application login).

Customer experience – A QR Code can provide immediate access to information such as warranty or support instructions for a product or service, thereby eliminating the need store and keep track of paper-based documentation. Details for products such as grocery items can be accessed, such as using the SmartLabel application.

Stores – A QR Code can provide a link to specifications for products on the shelf and could be used for promotions and discounts; QR Codes could also be used to scan items into a "shopping cart" for more automated checkout.

Restaurants – A QR Code can point to the restaurant's website, display the menu or initiate a take-out order.

D.6 Pros and cons of QR Codes

QR Codes have many advantages over linear bar codes.

- They consist of black squares arranged on a white background, which can store more data than linear bar codes. The more squares per side there are, the more data is encoded in a QR Code.
- You can scan QR Codes from digital screens. This is not the case for linear bar codes, which you can only scan from paper.
- While the software used to generate QR Codes doesn't collect personal information from users, the location and time of a scan, the number of times a code is scanned, and the operating system of the device that performed the scan are all available to the code's creators.
- A QR Code can't be hacked, but a hacker can generate a malicious QR Code that sends you to a fake website where they'll steal your personal data and can track your location, so always try to verify where your QR Code originated from.
- QR Codes are convenient to use but can suffer from security risk such as replacing the QR Code on public display. For example, hackers can can encode malicious links in QR Codes that direct users to malicious phishing sites. It is very easy for a bad actor to replace QR Codes in public places by their own printed malicious QR.

Appendix E. Detailed Use Case Descriptions

[Informative]

E.1 Use Case 1 – User login with a QR Code and mobile phone

Notes:

- The “User” is a party or person wishing to login to an “Application” provided by a relying party
- The User needs to login from an access “Device” (such as a desktop computer or laptop) via a web browser
- The User has an internet-connected mobile device, referred to as a “Phone”, that is pre-registered to the Application
- Any external party that attempts to intercept, change, or disrupt the login process is called an “Attacker.”

Problem description

The User wishes to access the Application using a desktop or laptop Device.

The relying party requires the User to be securely verified prior to granting access to the Application.

The User, who wishes to avoid yet another password or security key, wants to use a QR Code “no touch” solution that eliminates both the username and a static password.

Example of use

WhatsApp provides an option for desktop login using a QR Code scanned by its mobile app.

A typical example would be to access a messaging service or a shopping cart on a retail website.

Use case assumptions

1. Any web browser other than the User’s Phone can be used.
2. The Application supports login with a QR Code.
3. The Phone supports QR Code scanning and decoding.
4. The User has a previously established relationship with the Application.
5. Successful User validation is a pre-requisite for access to the Application.
6. The User knows a URL for the target Application.

Starting conditions

1. The User has opened the Application on the Phone and has accessed a QR Code reader.
2. The internet is available and performing adequately for both the User’s browser and for the Phone.
3. The User’s Phone is accessible and can read a QR Code displayed on the Device.

Basic login process

1. The User enters the Application URL in the Device browser.
2. The Device displays the Application login page which displays a QR Code. The User may need to select a QR Code option for it to be displayed.

   The QR Code displayed on the Device is dynamic and includes a time-out function.
3. The User scans the QR Code with the Phone. If the QR Code is not scanned within a specified time, the Browser resets the login page. The User can try scanning the QR Code again or can select another login option.

4. If the QR Code times out or the scan is invalid, an error message is displayed on the Phone. After some specified number of failed attempts, an error message is displayed, and the User may be locked out.
5. If the User QR Code is successfully validated, and optionally if other access control conditions are met, then the User is logged in and the Application's landing page is displayed.

Benefits/challenges of the QR Code approach

The use of a QR Code for this use case has the following benefits:

- Passwords are not needed and do not need to be remembered
- A shared secret is not used
- The QR Code cannot be easily phished or shared
- The QR Code approach can coexist with other verification methods

The following challenges arise from this login solution:

- A compatible Phone must be at hand
- Two communication channels must be available
- The login is not easily shared by multiple applications

Threat scenarios

Risk	Title	Description	Mitigation
1	QR Code substitution	An Attacker may find a way to substitute the QR Code that is displayed on the Device. If the QR Code is derived from the Session ID, there is a risk that the Attacker can force the User to use a Session ID that is shared with the Attacker, resulting in the Attacker's session being logged in by the User. If the QR Code contains a URL, then the User can be redirected to a website that is controlled by the Attacker, or the attacker can force a downgrade of the web protocol to http for possible credential capture.	Avoidable if the Session ID is encrypted and signed, or if the relying party includes the Authentication Service.
2	QR Code scanning	An Attacker may be able to scan the QR Code on their own Phone before the User is able to scan it.	If the Phone is pre-registered to the Application, then receiving the QR Code back from an unknown device should be an error.
3	Device access	An Attacker may have full access to the initially unauthenticated Device without the User's knowledge. After the User has logged on, the Attacker can perform activities as a User, and can potentially completely hijack the account.	Any irreversible account activities (like confirmation email/phone number or password change, transfer of funds, account ownership etc.) occurring on the browser should be confirmed separately at a previously logged-in Phone.

E.2 Use Case 2 – User login with a pre-established QR Code and one time password

Notes:

- The “User” is a party or person wishing to login to an “Application” provided by a relying party
- The User needs to login from an access “Device” (such as a desktop computer or laptop) via a standard browser
- The User has a mobile “Phone” with a phone number that has been registered to the Application. The Phone does not support mobile applications.
- Any external party that attempts to intercept, change, or disrupt the login process is called an “Attacker”.

Problem description

The User wishes to access the Application using the Device.

The relying party requires the User to be securely verified prior to granting access to the Application.

The User, who wishes to avoid yet another password or security key, wants to use a QR Code “no touch” solution that eliminates both the username and password.

Example of use

This use case provides an option for desktop login using a QR Code combined with a one-time password.

Use case assumptions

1. The User knows the address (i.e., a URL) for the target Application.
2. The Application supports login with a QR Code.
3. The User has previously established a relationship with the Application.
4. The Application can accept a One Time Password (OTP) as a second authentication factor.
5. Successful authentication is a pre-requisite for User access to the Application via the Device.

Starting conditions

1. The User has previously accessed the Application, has registered a username and an email or SMS address.
2. The Application has delivered a secure login QR Code (including the username and password or any other handles) to the Device.
3. The User has saved the QR Code on the Device or on any other removable media that may be accessible.
4. The Device has internet access which is performing adequately.
5. The email or SMS service is available and is functioning with acceptable performance.

Login process

1. When the User enters the Application URL, the Device displays the Application login page which includes a QR Code option.

   The browser must be on the same Device as the stored QR Code.

2. The User selects the QR Code option and copies the stored QR Code to the Application input block/page.

   The static QR Code is re-useable and is encrypted.

3. If the QR Code is not entered in time, the Device resets to the main login page.

The User can try again or select another login option.

4. The Application validates the QR Code. If the QR Code is correct, a One-Time Password (OTP) is sent to the designated email or an SMS address.

   If the QR Code is invalid, the OTP is not sent, and an error message is displayed on the Device.

5. If the OTP is received successfully, the User inputs it to the Application.

6. The Application then completes the authentication process. If successful, and optionally if any other required conditions (i.e., authorizations) are met, then the User is logged in and the Application landing page is displayed.

   If the OTP is not valid, an error message is displayed.

Benefits/challenges of the QR Code + OTP approach

The use of a QR Code for this use case has the following benefits:

- Passwords do not need to be remembered
- A shared secret is not needed
- The QR Code cannot be readily phished or shared
- A QR Code approach can coexist with other verification methods

The following challenges arise from this login solution:

- The email/SMS delivery of an OTP is less secure than a mobile app with QR Code reader
- The QR Code is static and potentially stored insecurely
- The solution is not easily shared by multiple applications
- Email/SMS must be available and not subject to significant delays

Threat scenarios

Risk	Title	Description	Mitigation
1	QR Code copying	An Attacker with access to the Device may be able to retrieve the QR Code. The Attacker could copy the QR Code to the login page and force an OTP to be delivered. The Attacker could delete or destroy the QR Code to prevent access.	The Attacker must have both the QR Code and the OTP to emulate the User.
2	Email/SMS access	An Attacker with the Phone may have access to the email or SMS to capture the OTP unless a login is required. The Attacker could block access to the Application by deleting or altering the OTP.	The Attacker must have both the QR Code and the OTP to emulate the User.
3	Application login page	The Attacker might intercept the initial URL and divert the login request to a malicious website. The User might not know how to proceed if the QR Code matching fails.	The Attacker must have both the QR Code and the OTP to emulate the User.

E.3 Use Case 3 – QR Code on mobile phone with reader on the device

Notes:

- The “User” is a party or person wishing to access an “Application” provided by a relying party
- The User needs to login from a non-standard “Device” that has a QR Code reader.
- The User has a mobile “Phone” that can store the QR Code.
- Any external party that attempts to intercept, change, or disrupt the process is called an “Attacker”.

Problem description

A User wishes to access the Application using a non-standard Device that may not be web-based. A typical example would be accessing a bank account via a proprietary automated teller machine (ATM).

The User has a Phone (smart phone or mobile device) available, but it may not have internet access at all times.

The relying party requires the User to be verified prior to granting access to the Application.

The User, who wishes to avoid yet another password or security key, prefers a solution that eliminates usernames, passwords, and dedicated access cards.

Examples of Use

(a) Access an Automated Teller Machine (ATM)

- QR Cash Standard Chartered Banks ATMs in Hong Kong https://www.sc.com/hk/bank-with-us/app-sc-mobile/qrcash/
- In Switzerland: https://www.six-group.com/dam/download/banking-services/cash/six-fs-bbs-qr-code-en.pdf

(b) Entry through a restricted door

- https://www.fingertec.com/images/w_brochure/QR110-e.html

Basic assumptions

1. The Device has a QR Code reader that is physically accessible.
2. The Application supports access using a QR Code.
3. The Phone can retrieve and display a QR Code.
4. The User has a previously established relationship (e.g., a bank account) with the Application owner.
5. Successful User authentication is a pre-requisite for access to and use of the Application.

Starting conditions

1. A static QR Code has been stored on the Phone previously.

   Alternatively, the QR Code can be dynamically retrieved if the Phone is connected.

Access process

1. The User presents the QR Code on the Phone for scanning by the Device.

   The QR Code could be dynamic If the Phone is online. Otherwise, it would need to be a static QR Code.

2. The Device submits the QR Code to the Application securely.

   If validated, the Device is activated for use. The functions of the Device will vary by application.

   If not validated, the QR Code is rejected, and an error message is displayed.

Benefits/Concerns of the QR Code Approach

The use of a QR Code for this use case has the following benefits:

- No human physical contact during the transaction
- Compatible with multiple access methods
- The Phone is more secure than an access card
- The QR Code can be changed before or after each use

The following challenges arise from this solution:

- The reader may not be reliable, especially in exposed locations
- The Phone must be available, and the User may need to log into the mobile App

Threat scenarios

Risk	Title	Description	Mitigation
1	Stolen QR Code	An attacker might steal a static QR Code and be able to use it, especially if the Phone is off-line.	The Phone and QR Code should be paired in order to be valid
2	Lost QR Code	The Phone may get lost, or the battery may be drained, preventing access.	The User would need to update their Application registration prior to using the new Phone
3	Compromised QR Code reader	The reader may be replaced by an Attacker's device.	

E.4 Use Case 4 – Third party QR Code with reader on device (e.g., a vaccination status)

Notes:

- The party or person wishing to use the system is called a "User"
- The User wants to show proof of "Status" to an "Approver"
- An "Authority" issues a QR Code that serves as a verification of Status
- The User has a "Phone" (a smartphone or equivalent mobile device)
- The Approver has a device that is able to read QR Codes
- Any external party that attempts to intercept, change, or disrupt the process is an "Attacker."

Problem description

A static QR Code is issued to a User by a recognized Authority to provide a Status. The QR Code is stored on the Phone. The QR Code contains data that:

(a) confirms the QR Code was issued by the Authority;
(b) confirms the QR Code was issued to the User; and
(c) verifies the User's Status (e.g., a COVID-19 vaccination status).

The Status represents validation at a specified point in time (e.g., when the QR Code was issued) and may not be current. If the Status can change, expire, or be revoked, then a real-time update of the verification would be desirable or even mandatory.

The QR Code could be displayed on the User's Phone or presented by any other means (e.g., card or paper).

The Approver has a device that can read the QR Code. The Approver must answer two questions:

(a) Is the party presenting the QR Code the correct User (i.e., is the person presenting the QR Code the owner of the Phone)?
(b) What is the User's current Status?

Basic assumptions

1. The User is able to present the QR Code to the Approver (e.g., in a store).
2. The Phone displaying the QR Code can authenticate the User.

For dynamic re-validation of the QR Code:

3. The Approver can request the Authority to re-confirm the validation.
4. The Authority can verify that the QR Code is valid.

Starting conditions

1. The User has previously received or downloaded a QR Code containing their Status at the time of issuance.
2. For re-validation the internet is available and performing adequately for the Approver.
3. The User has the Phone available and can present the QR Code to the Approver.

Verification process

1. The User opens the Phone and displays the QR Code. It must be possible to accept/deny permission when network access to the Authority is not available.

   Note: It is not required to have a mobile application to display the QR Code, but access to a mobile application would allow retrieval of a dynamic QR Code for each use.

2. The Approver scans the QR Code and extracts the Status.

   The result may be a simple Approve/Deny indicator, or it may include other instructions.

   There is a small risk that the Status is out-of-date (e.g., a vaccination status may not include follow-on infections).

   Displaying a QR Code using means other than a Phone is less secure because the QR Code may not be for the presenter.

3. If verification is successful (and optionally if other conditions are met), the User is approved.

   If the verification is unsuccessful (i.e., QR Code not readable, expired, or negative), then approval may be denied (although a User should not knowingly offer a negative status QR Code).

4. If the Approver has network access, then an updated QR Code can be requested from the Authority.

Benefits/concerns of the QR Code approach

The use of a QR Code for this use case has the following benefits:

- No human physical contact during the transaction
- Compatible with multiple display methods
- The Phone is more secure than an access card or paper
- The QR Code can be changed before or after each use

The following challenges arise from this solution:

- Validates "when produced" status instead of current status
- Users who know they have negative status may be motivated to "borrow" a valid QR Code
- May not be able to explain a denial
- The internet and the Authority app must be available for dynamic updates
- The Phone must be available

Threat scenarios

Risk	Title	Description	Mitigation
1	Off-line mode	Unless an app is used in the Phone, the QR Code cannot be checked, updated, and authenticated for each use	Use an online version of the application
2	Presenter verification	There is no assurance that the person presenting the QR Code is the owner of the Phone of the QR Code	Check picture ID at the same time as the Status is being checked
3	Privacy protection	There may be additional concerns if the QR Code is unencrypted or contains PII	Do not let the Device display PII or other private data

Appendix F. Notices

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website: [https://www.oasis-open.org/policies-guidelines/ipr/].

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. OASIS AND ITS MEMBERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THIS DOCUMENT OR ANY PART THEREOF.

As stated in the OASIS IPR Policy, the following three paragraphs in brackets apply to OASIS Standards Final Deliverable documents (Committee Specifications, OASIS Standards, or Approved Errata).

[OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this deliverable.]

[OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this OASIS Standards Final Deliverable. OASIS may include such claims on its website, but disclaims any obligation to do so.]

[OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this OASIS Standards Final Deliverable or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Standards Final Deliverable, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.]

The name "OASIS" is a trademark of OASIS, the owner and developer of this document, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, documents, while reserving the right to enforce its marks against misleading uses. Please see https://www.oasis-open.org/policies-guidelines/trademark/ for above guidance.