

# Samba Test Suite

Andrew Tridgell  
Samba Team

# Overview

- Protocols
  - SMB, SMB2
  - LDAP, CLDAP
  - RPC (SAMR, LSA, SRVSVC, DRS etc etc)
- Approaches
  - Traditional Tests
  - 'RAW' tests
  - 'Full Coverage' Tests
  - Protocol Scanners
  - Stochastic + Model Tests
  - Dual-server Tests

# Client Libraries

- Flexible, complete
  - The key to successful testing
  - Exposes all fields of the protocol
  - Structure oriented
  - Pad and reserved fields exposed
  - Async design
  - Auto-generated for some protocols (eg. RPC/IDL)

# Structure Oriented Interface

```
struct smb2_request *smb2_close_send(struct smb2_tree *tree, struct smb2_close *io);  
NTSTATUS smb2_close_recv(struct smb2_request *req, struct smb2_close *io);  
NTSTATUS smb2_close(struct smb2_tree *tree, struct smb2_close *io);
```

```
struct smb2_close {  
    enum smb_close_level level;  
    struct {  
        union smb_handle file;  
        uint16_t flags;  
        uint32_t _pad;  
    } in;  
    struct {  
        uint16_t flags;  
        uint32_t _pad;  
        NTTIME create_time;  
        NTTIME access_time;  
        NTTIME write_time;  
        NTTIME change_time;  
        uint64_t alloc_size;  
        uint64_t size;  
        uint32_t file_attr;  
    } out;  
}
```

# 'RAW' Tests

- A high coverage conventional test
  - Tests each field of each operation
  - Uses internal redundancy of protocol for cross-checking
  - Doesn't try many combinations of parameters
- Example
  - RAW-OPEN, RAW-QFILEINFO etc

# Full Coverage Tests

- Enumerate all possibilities
  - Work in a narrow domain
  - Try all bits, or all info levels, or all operations
- Examples
  - All OpenX deny modes (1296 combinations)
  - All UCS2 chars (64k combinations)
  - All file\_attribute bits (32 bits)
  - All desired\_access bits (32 bits)

# Protocol Scanners

- Discover available operations
  - Info levels
  - Opcodes
  - Looks for changes in error return
  - Doesn't need to send valid requests
- Examples
  - SMB2 opcode scanner
  - Trans2/GetInfo/Setinfo level scanners
  - RPC scanners

# Stochastic + Model Tests

- Model a subsystem
  - Simple model of a small piece of semantics
- Random Exploration
  - Try random inputs
  - Compare model to target server
  - May allow control of parameter space
- Examples
  - Wildcard testing
  - NT share mode testing



# Dual Server Testing

- “Protocol Differencing”
  - Compare behaviour of two servers for a protocol
  - Constrained random parameter generation
  - Automatic backtracking for minimal test case when a difference is found
  - Simple control over ignoring fields, parameters and opcodes
- Example
  - locktest (byte range locking)
  - gentest (general SMB/SMB2 testing)
- Tutorial (flash video)
  - See [http://samba.org/~tridge/samba\\_testing/](http://samba.org/~tridge/samba_testing/)