April 17-18, 2024
sambaXP Conference

Microsoft Interoperability Track
FileServer TestSuite Updates 2024

Obaro Ogbo
Senior Software Engineer
Windows Protocols Test Suites Overview

What is the Windows Protocols Test Suites?

Collection of Test Suites
Each Test Suite evaluates the implementation of a family of protocols - FileServer, RDP Server, RDP Client, Active Directory, etc

Evaluates whether a protocol implementation meets certain interoperability requirements.

• Originally developed for in-house testing of Microsoft Open Specifications.
• Used to test/verify Windows behavior.
• Also used to test 3rd-party implementations.

Do not cover every protocol requirement, and do not certify an implementation, but can be a useful indication of interoperability.
Windows Protocol Test Suites Architecture

- PTM Service / PTM Cli
  - FileServer Test Suite
  - RDP Client Test Suite
  - RDP Server Test Suite
  - ....
- Protocol Test Framework
- Proto SDK
- .NET SDK
  - Windows
  - MacOS
  - Linux
  - Others
# Windows Protocol Test Suites

Open sourced on GitHub since 2016

[https://github.com/Microsoft/WindowsProtocolTestSuites](https://github.com/Microsoft/WindowsProtocolTestSuites)

[https://github.com/Microsoft/ProtocolTestFramework](https://github.com/Microsoft/ProtocolTestFramework)

<table>
<thead>
<tr>
<th>Category</th>
<th>Test Suite Name</th>
<th>Protocols</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Sharing</td>
<td><strong>FileServer</strong></td>
<td>MS-SMB2, MS-FSRVP, MS-SWN, MS-DFSC, MS-RSVD, MS-SQOS, MS-FSA, MS-HVRS</td>
</tr>
<tr>
<td></td>
<td>MS-SMB</td>
<td>MS-SMB</td>
</tr>
<tr>
<td></td>
<td>MS-SMBD</td>
<td>MS-SMBD</td>
</tr>
<tr>
<td></td>
<td><strong>BranchCache</strong></td>
<td>MS-PCCRC, MS-PCCRR, MS-PCCRTP, MS-PCHC, MS-CCROD</td>
</tr>
<tr>
<td>Security</td>
<td>Kerberos</td>
<td>MS-KILE, MS-KKDCP, MS-PAC</td>
</tr>
<tr>
<td></td>
<td>MS-AZOD</td>
<td>MS-AZOD</td>
</tr>
<tr>
<td>Active Directory</td>
<td><strong>ADFamily</strong></td>
<td>MS-ADTS-LDAP, MS-ADTS-PublishDC, MS-ADTS-Schema, MS-ADTS-Security, MS-APDS, MS-DRSR, MS-FRS2, MS-LSAD, MS-LSAT, MS-NRPC, MS-SAMR</td>
</tr>
<tr>
<td></td>
<td>MS-ADOD</td>
<td>MS-ADOD</td>
</tr>
<tr>
<td>Remote Desktop</td>
<td><strong>RDP Client/RDP Server</strong></td>
<td>MS-RDPBCGR, MS-RDPEDISP, MS-RDPEDYC, MS-RDPEGFX, MS-RDPEGT, MS-RDPEI, MS-RDPEMT, MS-RDPEUDP, MS-RDPEUSB, MS-RDPEVOR, MS-RDPRFX, MS-RDPELE</td>
</tr>
<tr>
<td>BYOD</td>
<td>MS-ADFSPIP</td>
<td>MS-ADFSPIP</td>
</tr>
<tr>
<td>XCA</td>
<td>MS-XCA</td>
<td>MS-XCA</td>
</tr>
</tbody>
</table>
FileServer Test Suite Scope

**Protocols**

- **[MS-SMB2]**: Server Message Block (SMB) Protocol Versions 2 and 3
- **[MS-FSA]**: File System Algorithms
- **[MS-SQN]**: Storage Quality of Service Protocol
- **[MS-SRN]**: Remote Shared Virtual Disk Protocol
- **[MS-FSVP]**: File Server Remote VSS Protocol
- **[MS-RSVD]**: Remote Witness Protocol
- **[MS-DFSC]**: Distributed File System (DFS): Referral Protocol
- **[MS-HVRS]**: Hyper-V Remote Storage Profile

**Messages**

- **Negotiate** (Negotiate Contexts)
- **Session Setup**
- **Tree Connect/Disconnect**
- **Create/Close** (Create Contexts)
- **IOCTL**
- **Query Directory**
- **Query Info/Set Info**
- **File Access**
- **Leasing**

**Dialects**

- Dialect 2.002
- Dialect 2.1
- Dialect 3.0
- Dialect 3.02
- Dialect 3.11
FileServer Test Suite Approach

• **Traditional Test**
  - Covers basic functionalities and scenarios
  - All new test cases

• **Model-based Test**
  - 2K+ test cases generated by 13 models
  - Still supported, no more expanded
FileServer Test Suite Infrastructure

**Synthetic Client** (Driver Computer)

- Test Cases
- Protocol Adapter
- ProtoSDK

**Protocol Test Framework**

- MSTest

**SUT Control Adapter**
- Managed
- PowerShell
- Shell

**SUT** (System Under Test)

- File Server

**TCP / QUIC**

---

1. Configuration
2. Adapter
3. Logging
4. Network Capture
What’s New (Github Release 4.24.4.0)
FileServer TestSuites Updates

Windows Protocol TestSuites Release v4.24.4.0
https://github.com/Microsoft/WindowsProtocolTestSuites/releases/tag/4.24.4.0

**FileServer Test Suites**
- Update to dotnet 8
- New Tests
- Bug fixes

**Protocol Test Manager Service**
- Expand custom test categories
- Other bug fixes

**Protocol Test Framework**
- Update to dotnet 8
- Bug fixes
New Tests for FSCTL_CREATE_OR_GET_OBJECT_ID

- Tests algorithm for FSCTL_CREATE_OR_GET_OBJECT_ID on a given file system
  - Detailed description available in [MS-FSA] and [MS-FSCC]
    https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-fsa
    https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-fsc

- Verify File Object has valid Object Id after invoking on object
FSCTL_CREATE_OR_GET_OBJECT_ID test sample

Test Driver

Establish connection with file server

Create file

Send FSCTL_CREATE_OR_GET_OBJECT_ID control code

Verify correct response

Verify valid Object Id

SUT
Contact Us - GitHub

Open an Issue
https://www.github.com/Microsoft/WindowsProtocolTestSuites/issues

Create a Pull Request
https://github.com/microsoft/WindowsProtocolTestSuites/pulls

Consult CONTRIBUTING.md
For questions and/or feedback

Please email us
testsuitehelp@microsoft.com

Open an issue on Github
https://www.github.com/Microsoft/WindowsProtocolTestSuites/issues
Overview of SMB2 Dissectors on Wireshark

Adedeji Adeloye
Software Engineer
• What this session is not about
• Why Wireshark?
• Our Goal
• Contributions so far
• Demo
• Open Discussion
What this session is not about...

- Introduction to Wireshark
- How to use Wireshark
- Wireshark vs. other tools
- How Windows protocols work
Why Wireshark?

- A core part of our interoperability efforts with partners and customers is to provide test tools to assist them with implementing and testing Microsoft Windows protocols.
- For many years, Microsoft supported customers with Microsoft Message Analyzer (MMA) as a protocol parsing tool.
- Following the deprecation of MMA, and based on our research and feedback from customers, we adopted Wireshark as a replacement for MMA, to continue providing interop support to our partners and customers.
- Protocol parsers (known as Dissectors on Wireshark) provide users with the capability to decode raw data on the wire to human readable structures.
Our Goal

To contribute dissectors to Wireshark to enable partners, customers, and all users get detailed and up-to-date packet information for Windows protocols.
# Overview of Contributions So Far

<table>
<thead>
<tr>
<th>SMB2 Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tree connect response flags</td>
<td>New flag: isolated transport flag; to indicate to the client that the server prefers communication with the share to be on a separate set of connection</td>
</tr>
</tbody>
</table>
| Query Info                                             | Flags dissection was updated to properly dissect flags: SL_RESTART_SCAN, SL_RETURN_SINGLE_ENTRY, SL_INDEX_SPECIFIED  
These were previously ignored                                                                         |
| FSCTL_SET_INTEGRITY_INFORMATION_Ex                     | This is an FSA control code that is only supported on REFS file systems to request the server to set integrity information for a file or directory.                                                       |
| FSCTL_REFS_STREAM_SNAPSHOT_MANAGEMENT                  | This control code requests the server to perform a specific snapshot operation on a given data stream in a file.                                                                                       |
## Overview of Contributions So Far

<table>
<thead>
<tr>
<th>SMB2 Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server to client notification</td>
<td>This is a recent packet sent from the server to the client to perform implementation specific details without expecting any response</td>
</tr>
<tr>
<td>FileFullEaInformation flags</td>
<td>Control flags for this message was also updated</td>
</tr>
<tr>
<td>Lock response</td>
<td>Previously mislabeled fields were updated in the lock response message</td>
</tr>
<tr>
<td>SMB2&amp;3 code bug fixes</td>
<td>We also fixed a couple of bugs in the SMB2&amp;3 (packet-smb2.c) dissector code file in Wireshark that prevents Wireshark developers from debugging during development</td>
</tr>
</tbody>
</table>
Demo
Open Discussion

- SMB3 decryption on Wireshark?
- Other Windows protocols you would like us to contribute to?
Try out our contributions:

https://www.wireshark.org/download/automated/

Got questions or feedback?

Please email us:

winwiresharkhelp@microsoft.com

To download and start testing with our contributions, go to Index of /download/automated/win64 (wireshark.org) and download the latest installer with version number 4.3.0rc0-1317-gxxxxxxxxxx-x64.exe and above.
Thank you!