Windows Protocols Errata

This topic lists the Errata found in the Windows Protocols Technical Specifications, Overview Documents, and Reference documents since they were last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

Errata are content issues in published versions of protocols documents that could impact an **implementation**. Examples of errata are errors or missing information in the normative sections of the Technical Specifications or in the use cases (examples) in the Technical Specifications and Overview Documents.

Content issues that don't impact an implementation, for example, editorial updates due to typos, formatting updates, and rewrites for readability and clarity, are **not** included in Errata.

The sections below list the Windows Protocols documents that contain active Errata (i.e., Errata not yet released with the documents on Docs.Microsoft.Com [DMC]) and provide links to archived Errata (i.e., Errata already released with the documents on DMC).

Protocols Documents with Active Errata

[MC-NBFX]: .NET Binary Format XML Data Structure

[MC-NMF]: .NET Message Framing Protocol

[MS-ADDM]: Active Directory Web Services: Data Model and Common Elements

[MS-ADFSPIP]: Active Directory Federation Services and Proxy Integration Protocol

[MS-ADFSWAP]: Active Directory Federation Service (AD FS) Web Agent Protocol

[MS-ADSC]: Active Directory Schema Classes

[MS-ADTS]: Active Directory Technical Specification

[MS-BKUP]: Microsoft NT Backup File Structure

[MS-CIFS]: Common Internet File System (CIFS) Protocol

[MS-CRTD]: Certificate Templates Structure

[MS-CSRA]: Certificate Services Remote Administration Protocol

[MS-CSVP]: Failover Cluster: Setup and Validation Protocol (ClusPrep)

[MS-DFSC]: Distributed File System (DFS) Referral Protocol

[MS-DHCPE]: Dynamic Host Configuration Protocol (DHCP) Extensions

[MS-DNSP]: Domain Name Service (DNS) Server Management Protocol

[MS-DRSR]: Directory Replication Service (DRS) Remote Protocol

[MS-DTYP]: Windows Data Types

[MS-ECS]: Enterprise Client Synchronization Protocol

[MS-EMFPLUS]: Enhanced Metafile Format Plus Extensions

[MS-ERREF]: Windows Error Codes

[MS-EVEN]: EventLog Remoting Protocol

[MS-FRS2]: Distributed File System Replication Protocol

[MS-FSA]: File System Algorithms

[MS-FSCC]: File System Control Codes

[MS-GPOL]: Group Policy: Core Protocol

[MS-GPWL]: Group Policy: Wireless/Wired Protocol Extension

[MS-IKEE]: Internet Key Exchange Protocol Extensions

[MS-KILE]: Kerberos Protocol Extensions

[MS-LSAD]: Local Security Authority (Domain Policy) Remote Protocol

[MS-NRBF]: .NET Remoting: Binary Format Data Structure

[MS-NCNBI]: Network Controller Northbound Interface Specification

[MS-NNS]: .NET NegotiateStream Protocol

[MS-NRPC]: Netlogon Remote Protocol

[MS-OCSPA]: Microsoft OCSP Administration Protocol

[MS-PAC]: Privilege Attribute Certificate Data Structure

[MS-PAR]: Print System Asynchronous Remote Protocol

[MS-RAI]: Remote Assistance Initiation Protocol

[MS-RDPBCGR]: Remote Desktop Protocol: Basic Connectivity and Graphics Remoting

[MS-RDPECAM]: Remote Desktop Protocol: Video Capture Virtual Channel Extension

[MS-RDPEDISP]: Remote Desktop Protocol: Display Update Virtual Channel Extension

[MS-RDPEGFX]: Remote Desktop Protocol: Graphics Pipeline Extension

[MS-RDPELE]: Remote Desktop Protocol: Licensing Extension

[MS-RDPEMT]: Remote Desktop Protocol: Multitransport Extension

[MS-RDPEPC]: Remote Desktop Protocol: Print Virtual Channel Extension

[MS-RDPERP]: Remote Desktop Protocol: Remote Programs Virtual Channel Extension

[MS-RDPEUDP]: Remote Desktop Protocol: UDP Transport Extension

[MS-RDPRFX]: Remote Desktop Protocol: RemoteFX Codec Extension

[MS-RMPR]: Rights Management Services (RMS): Client-to-Server Protocol

[MS-RPRN]: Print System Remote Protocol

[MS-RRASM]: Routing and Remote Access Server (RRAS) Management Protocol

[MS-RRP]: Windows Remote Registry Protocol

[MS-SAMR]: Security Account Manager (SAM) Remote Protocol (Client-to-Server)

[MS-SFU]: Kerberos Protocol Extensions Service for User and Constrained Delegation Protocol

[MS-SMB2]: Server Message Block (SMB) Protocol Versions 2 and 3

[MS-SMBD]: SMB2 Remote Direct Memory Access (RDMA) Transport Protocol

[MS-SSTR]: Smooth Streaming Protocol

[MS-SWN]: Service Witness Protocol

[MS-TSTS]: Terminal Services Terminal Server Runtime Interface Protocol

[MS-VHDX]: Virtual Hard Disk v2 (VHDX) File Format

[MS-WCCE]: Windows Client Certificate Enrollment Protocol

[MS-WKST]: Workstation Service Remote Protocol

[MS-WMIO]: Windows Management Instrumentation Encoding Version 1.0 Protocol

[MS-WSP]: Windows Search Protocol

[MS-WSUSAR]: Windows Server Update Services: Administrative API Remoting Protocol

[MS-XCA]: Xpress Compression Algorithm

Errata Archives

June 30, 2015 - Download

October 16, 2015 - Download

March 2, 2016 - Download

July 18, 2016 - Download

September 26, 2016 - Download

March 20, 2017 - Download

June 1, 2017 - Download

August 21, 2017 - Download

September 15, 2017 - Download

December 1, 2017 - Download

March 16, 2018 - Download

September 12, 2018 - Download

March 13, 2019 - Download

June 24, 2019 – Download

September 23, 2019 - Download

October 14, 2019 - Download

March 4, 2020 - Download

June 15, 2020 - Download

[MC-DTCXA]: MSDTC Connection Manager OleTx XA Protocol

This topic lists the Errata found in [MC-DTCXA] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MC-NBFX]: .NET Binary Format XML Data Structure

This topic lists the Errata found in [MC-NBFX] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

Errata below are for Protocol Document Version V8.0 – 2019/03/13.

Errata Published*	Description
2019/12/09	In Section 2.2.3.30, QNameDictionaryTextRecord(0xBC), the length of the Name field was changed from 3 bytes to variable: Changed from: Name (3 bytes) Changed to: Name (variable) The packet diagram for the message was also changed to reflect the length.

*Date format: YYYY/MM/DD

[MC-NMF]: .NET Message Framing Protocol

This topic lists the Errata found in the MC-NMF document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

Errata below are for Protocol Document Version V9.0 – 2018/03/16.

Errata Published*	Description
2018/07/02	In Section 2.2.6, Preamble Message, the field descriptions have been modified as follows and have been moved to follow the packet diagram. Changed from: The VersionRecord MUST be formatted as specified in section 2.2.3.1. The ModeRecord MUST be formatted as specified in section 2.2.3.2. The ViaRecord MUST be formatted as specified in section 2.2.3.3. The EnvelopeEncodingRecord MUST be formatted as specified in section 2.2.3.4 Changed to: VersionRecord (3 bytes): This field MUST be formatted as specified in section 2.2.3.1. ModeRecord (2 bytes): This field MUST be formatted as specified in section 2.2.3.2. ViaRecord (variable): This field MUST be formatted as specified in section 2.2.3.3. EnvelopeEncodingRecord (variable): This field MUST be formatted as specified in section 2.2.3.4

*Date format: YYYY/MM/DD

[MC-PRCR]: Peer Channel Custom Resolver Protocol

This topic lists the Errata found in [MC-PRCR] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 15, 2017 - Download

[MS-ABTP]: Automatic Bluetooth Pairing Protocol

This topic lists the Errata found in [MS-ABTP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-ADA2]: Active Directory Schema Attributes M

This topic lists the Errata found in the MS-ADA2 document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

[MS-ADA3]: Active Directory Schema Attributes N-Z

This topic lists the Errata found in the MS-ADA3 document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-ADDM]: Active Directory Web Services: Data Model and Common Elements

This topic lists the Errata found in [MS-ADDM] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

Errata below are for Protocol Document Version V15.0 – 2018/09/12.

Errata Published*	Description
2018/12/17	In Section 1.2.1, Normative References, the following reference has been deleted: [RFC4346] Dierks, T., and Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.1", RFC 4346, April 2006, http://www.ietf.org/rfc/rfc4346.txt In Section 2.1, Endpoints, changed from: The ADWS protocol set uses two types of authentication. Each endpoint (except for the "mex" endpoint) supports one or the other. The forms of authentication are: • Windows Integrated: These endpoints use Transport Layer Security (TLS) [RFC4346] to protect the TCP transport. Integrated Windows authentication using the .Net Negotiate Stream protocol [MS-NNS] is used to authenticate the client to the server at the transport layer and to negotiate the session key used for TLS. Changed to: The ADWS protocol set uses two types of authentication. Each endpoint (except for the "mex" endpoint) supports one or the other. The forms of authentication are: • Windows Integrated: These endpoints use integrated Windows authentication with the .Net Negotiate Stream protocol [MS-NNS] to authenticate the client and provide message security at the transport layer.

* Date format: YYYY/MM/DD

[MS-ADFSOAL]: Active Directory Federation Services OAuth Authorization Code Lookup Protocol

This topic lists the Errata found in [MS-ADFSOAL] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-ADFSPIP]: Active Directory Federation Services and Proxy Integration Protocol

This topic lists the Errata found in the MS-ADFSPIP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 30, 2015 - Download

July 18, 2016 - Download

September 15, 2017 - Download

September 12, 2018 - Download

Errata below are for Protocol Document Version V7.0 – 2018/09/12.

Errata Published*	Description
2019/05/27	In the sections listed below, the enum Certificate Type values have been changed from string to integer: Section 2.2.2.12, Port Type Section 2.2.2.14, TLS Query Behavior Section 2.2.2.15, Certificate Validation Section 2.2.2.16, Certificate Type Section 2.2.2.17, Error Type Section 3.10.5.1.1.3, Processing Details Section 3.10.5.1.1.3, Processing Details Section 3.11.5, Message Processing Events and Sequencing Rules Section 3.11.5.1, End-user X509 Certificate Success Processing Section 3.11.5.2, End-user X509 Certificate Common Processing Section 6, Appendix A: Full JSON Schema For details on the above changes, see the PDF doc here.
2019/05/27	In Section 3.10.5.1.1.2, Response Body, changed from: No response body is returned. Changed to: The response from the server MUST be returned to the client.

*Date format: YYYY/MM/DD

[MS-ADFSWAP]: Active Directory Federation Service (AD FS) Web Agent Protocol

This topic lists the Errata found in [MS-ADFSWAP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

Errata below are for Protocol Document Version V11.0 – 2018/09/12.

Errata Published*	Description
2019/11/25	In Section 3.1.4.1.1.3, GetFsTrustInformationSoapOut, and Section 6, Appendix: Full WSDL, the value of minOccurs was changed from 1 to 0. Changed from: `<s:complexType name="VersionInformation"> <s:sequence> <s:element minOccurs="1" maxOccurs="1" name="SoftwareVersion" type="s:long" /> <s:element minOccurs="1" maxOccurs="1" name="Guid" type="s1:guid" /> <s:element minOccurs="1" maxOccurs="1" name="Version" type="s:long" /> </s:sequence>` Changed to: `<s:complexType name="VersionInformation"> <s:sequence> <s:element minOccurs="0" maxOccurs="1" name="SoftwareVersion" type="s:long" /> <s:element minOccurs="0" maxOccurs="1" name="Guid" type="s1:guid" /> <s:element minOccurs="0" maxOccurs="1" name="Version" type="s:long" /> </s:sequence>`

*Date format: YYYY/MM/DD

[MS-ADLS]: Active Directory Lightweight Directory Services Schema

This topic lists the Errata found in the MS-ADLS document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-ADSC]: Active Directory Schema Classes

This topic lists the Errata found in the MS-ADSC document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

Errata below are for Protocol Document Version V23.0 – 2018/03/16.

Errata Published*	Description

2019/09/16	In Section 2.243, Class samDomain, changed from:
	
	`(OA;CIOI;RPWP;3f78c3e5-f79a-46bd-a0b8-9d18116ddc79;;PS)`
	`S:(AU;SA;WDWOWP;;;WD)(AU;SA;CR;;;BA)(AU;SA;CR;;;DU)`
	
	Changed to:
	
	`(OA;CIOI;RPWP;3f78c3e5-f79a-46bd-a0b8-9d18116ddc79;;PS)`
	`(OA;CIIO;SW;9b026da6-0d3c-465c-8bee-5199d7165cba;bf967a86-0de6-11d0-a285-00aa003049e2;PS)`
	`(OA;CIIO;SW;9b026da6-0d3c-465c-8bee-5199d7165cba;bf967a86-0de6-11d0-a285-00aa003049e2;CO)`
	`S:(AU;SA;WDWOWP;;;WD)(AU;SA;CR;;;BA)(AU;SA;CR;;;DU)`

*Date format: YYYY/MM/DD

[MS-ADTS]: Active Directory Technical Specification

This topic lists the Errata found in the MS-ADTS document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

March 20, 2017 - Download

September 15, 2017 - Download

December 1, 2017 - Download

March 16, 2018 - Download

September 12, 2018 - Download

March 13, 2019 - Download

March 4, 2020 - Download

Errata below are for Protocol Document Version V51.0 – 2020/03/04.

Errata Published*	Description
2020/06/22	In Section 6.1.6.7.9 trustAttributes, in the bit flags description table, moved the position of the TAEC attribute to immediately follow the TANC attribute. Changed from: TANC (TRUST_ATTRIBUTE_CROSS_ORGANIZATION_NO_TGT_DELEGATION) 0x00000200 TAPT (TRUST_ATTRIBUTE_PIM_TRUST) 0x00000400 TAEC (TRUST_ATTRIBUTE_CROSS_ORGANIZATION_ENABLE_TGT_DELEGATION) 0x00000800

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>Changed to:

<table><tr><td>TANC
(TRUST_ATTRIBUTE_CROSS_ORGANIZATION_NO_TGT_DELEGATION)
0x00000200</td></tr><tr><td>TAEC
(TRUST_ATTRIBUTE_CROSS_ORGANIZATION_ENABLE_TGT_DELEGATION)
0x00000800</td></tr><tr><td>TAPT
(TRUST_ATTRIBUTE_PIM_TRUST)
0x00000400</td></tr></table></td></tr>
<tr><td>2020/06/08</td><td>In Section 3.1.1.3.1.3.1, Search Filters, changed from:

Filter clauses of the form (objectClass=*), (distinguishedName=*), (name=*), and (objectGUID=*) always evaluate to true for all objects.

Changed to:

Filter clauses of the form (objectClass=*), (distinguishedName=*), (name=*), and (objectGUID=*) always evaluate to true for all objects.
A filter can be constructed recursively such that the filter clause takes the form of another filter. The maximum recursion depth supported by Active Directory is hardcoded to 512.</td></tr>
<tr><td>2020/05/11</td><td>In Section 6.1.6.7.9, trustAttributes, the description for the TANC

(TRUST_ATTRIBUTE_CROSS_ORGANIZATION_NO_TGT_DELEGATION) flag has been changed from:

If this bit is set, tickets granted under this trust MUST be trusted for delegation. The behavior controlled by this bit is as specified in [MS-KILE] section 3.3.5.7.5.

Only supported on Windows Server 2012 and later after [MSKB-4490425] updates are installed.

Changed to:

If this bit is set, tickets granted under this trust MUST be trusted for delegation. The behavior controlled by this bit is as specified in [MS-KILE] section 3.3.5.7.5.

Only supported on Windows Server 2008 and later after [MSKB-4490425] updates are installed.

In that same section the description for the TAEC

(TRUST_ATTRIBUTE_CROSS_ORGANIZATION_ENABLE_TGT_DELEGATION) flag has been changed from:

If this bit is set, tickets granted under this trust MUST be trusted for delegation. The behavior controlled by this bit is as specified in [MS-KILE] section 3.3.5.7.5.</td></tr>
</table>

Errata Published*	Description
	Only supported on Windows Server 2012 and later after [MSKB-4490425] updates are installed. Changed to: If this bit is set, tickets granted under this trust MUST be trusted for delegation. The behavior controlled by this bit is as specified in [MS-KILE] section 3.3.5.7.5. Only supported on Windows Server 2008 and later after [MSKB-4490425] updates are installed.
2020/05/11	In Section 5.1.2.2, Using SSL/TLS, added Channel Binding information for SSL/TLS protected LDAP connections and added references to [RFC5929], [RFC5056], and [RFC4121]. Changed from: As indicated in the previous section, Active Directory does not permit SASL-layer message confidentiality/integrity protection mechanisms to be employed on an SSL/TLS-protected LDAP connection. Changed to: As indicated in the previous section, Active Directory does not permit SASL-layer message confidentiality/integrity protection mechanisms to be employed on an SSL/TLS-protected LDAP connection. Active Directory supports channel binding on an SSL/TLS-protected LDAPconnection, as specified in [RFC5929], [RFC5056], and [RFC4121]. Active Directory can be configured for channel binding in the following ways: ● To not use channel binding (the default). ● To use channel binding but refuse connections that do not meet channel binding requirements. ● To use channel binding and permit connections that do not meet channel binding requirements. The mechanism to specify such configurations is implementation-defined.
2020/04/27	In Section 2.2.20.6, KEYCREDENTIALLINK_ENTRY Identifiers, for the DeviceId(0x6) 'Identifier value', changed the 'Description of the data stored in the Value field'. Changed from: Must contain all zeros Changed to: Contains a device object identifier, or all zeros
2020/03/16	In Section 2.2.4, DS_REPL_OPW_BLOB, a reference to the opType field name has been corrected. Changed from: ulOptions (4 bytes): Zero or more bits from the Directory Replication Service (DRS) options defined in [MS-DRSR] section 5.41, the interpretation of which depends on the OpType.

Errata Published*	Description
	Changed to: ulOptions (4 bytes): Zero or more bits from the Directory Replication Service (DRS) options defined in [MS-DRSR] section 5.41, the interpretation of which depends on the opType.

*Date format: YYYY/MM/DD

[MS-AIPS]: Authenticated Internet Protocol

This topic lists the Errata found in the MS-AIPS document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

[MS-APDS]: Authentication Protocol Domain Support

This topic lists the Errata found in the MS-APDS document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-AZOD]: Authorization Protocols Overview

This topic lists the Errata found in the MS-AZOD document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-BKRP]: BackupKey Remote Protocol

This topic lists the Errata found in the MS-BKRP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-BKUP]: Microsoft NT Backup File Structure

This topic lists the Errata found in the MS-BKUP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

Errata below are for Protocol Document Version V9.0 – 2018/09/12.

Errata Published*	Description
2020/04/27	In Section 2.13.1, Creating an NT Backup File, a new paragraph was added to the end of the section. Added: If F has any ghosted extents, the NT backup file MUST generate one GHOSTED_EXTENT backup stream structure. During restore the GHOSTED_EXTENT backup stream structure is presented to the filesystem to recreate the file ghosted extent state. The following new sections were added: 2.12 FileSystem Ghosted Extents Functionality Hierarchical Storage Management (HSM) solutions on top of a filesystem remove cold data and move it to the next storage tier. This movement creates sparse holes in file system data, and HSM solutions have to maintain mappings between those sparse holes and location of the data in the new tier. An implementation of the Ghosted extents feature helps this process by maintaining some token in the sparse holes, on behalf of the HSM solution. This obviates the need for the HSM solution to maintain its own mappings. When a file with such tokens, hereby referred to as ghosted extents, are backed up, the backup process should store the tokens and their locations in the backup stream state. On restore those tokens should be reinserted in the data stream in exactly the same locations to recreate the original file state. The method to query the tokens, serialize the tokens, and restore the tokens is file system implementation specific. 2.12.1 Ghosted Extents Stream Structure A ghosted extent stream structure represents ghosted extents in the DATA backup stream. Ghosted extents are a kind of sparse extents, which store a GUID representing the owner of the extent and some variable-sized metadata. The structure of the data portion of this backup stream for a specific implementation is as follows: 0 1 2 3 4 5 6 7 8 9 1 0 1 2 3 4 5 6 7 8 9 2 0 1 2 3 4 5 6 7 8 9 3 0 1

Errata Published*	Description
	Count TotalCount Data (variable) ... Count (4 bytes): The number of extents in the Data portion. TotalCount (4 bytes): The total number of ghosted extents in the stream. Data (Variable): The data portion of the above structure contains a variable number of extents. The number of extents is given by Count. The structure of each Extent is described below: 0 1 2 3 4 5 6 7 8 9 1 0 1 2 3 4 5 6 7 8 9 2 0 1 2 3 4 5 6 7 8 9 3 0 1 Offset ... Length ... Guid ... ... NextOffset Size Data (variable) ...

Errata Published*	Description
	Offset (8 bytes): The logical byte offset in the DATA backup stream where the ghosted extent starts. Length (8 bytes): The logical length of the ghosted extent. GUID (16 bytes): The GUID identifier of the owner for the ghosted extent. NextOffset (4 bytes): Offset to the next Extent structure. Size (4 bytes): Size of the metadata of the ghosted extent. Data (variable): Metadata of the ghosted extent.

*Date format: YYYY/MM/DD

[MS-CAPR]: Central Access Policy Identifier (ID) Retrieval Protocol

This topic lists the Errata found in the MS-CAPR document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-CDP]: Connected Devices Platform Protocol Version 3

This topic lists the Errata found in the MS-CDP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-CHAP]: Extensible Authentication Protocol Method for Microsoft Challenge Handshake Authentication Protocol (CHAP)

This topic lists the Errata found in the MS-CHAP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-CFB]: Compound File Binary File Format

This topic lists the Errata found in the MS-CFB document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

March 16, 2018 - Download

[MS-CIFS]: Common Internet File System (CIFS) Protocol

This topic lists the Errata found in the MS-CIFS document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

June 1, 2017 - Download

December 1, 2017 - Download

September 12, 2018 - Download

March 4, 2020 - Download

Errata below are for Protocol Document Version V29.0 – 2020/03/04

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td>2020/08/17</td><td>In Section 2.2.2.3.4 SET Information Level Codes the table was updated

Changed from:

<table>
<tr><th>Name</th><th>Code</th><th>Description</th><th>Dialect</th></tr>
<tr><td>SMB_INFO_STANDARD</td><td>0x0001</td><td>Set creation, access, and last write timestamps.</td><td>LANMAN 2.0</td></tr>
<tr><td>SMB_INFO_SET_EAS</td><td>0x0002</td><td>Set a specific list of extended attributes (EAs).</td><td>LANMAN 2.0</td></tr>
<tr><td>SMB_SET_FILE_BASIC_INFO</td><td>0x0101</td><td>Set 64-bit create, access, write, and change timestamps along with extended file attributes. Not supported for TRANS2_SET_PATH_INFORMATION (section 2.2.6.7).</td><td>NT LANMAN</td></tr>
<tr><td>SMB_SET_FILE_DISPOSITION_INFO</td><td>0x0102</td><td>Set whether or not the file is marked for deletion. Not supported for TRANS2_SET_PATH_INFORMATION (section 2.2.6.7).</td><td>NT LANMAN</td></tr>
<tr><td>SMB_SET_FILE_ALLOCATION_INFO</td><td>0x0103</td><td>Set file allocation size. Not supported for TRANS2_SET_PATH_INFORMATION (section 2.2.6.7).</td><td>NT LANMAN</td></tr>
<tr><td>SMB_SET_FILE_END_OF_FILE_INFO</td><td>0x0104</td><td>Set file EOF offset. Not supported for TRANS2_SET_PATH_INFORMATION (sectio</td><td>NT LANMAN</td></tr>
</table>
</td></tr>
</table>

Errata Published*	Description

		n 2.2.6.7).	

Changed to:

Name	Code	Description	Dialect
SMB_INFO_STANDARD	0x0001	Set creation, access, and last write timestamps.	LANMAN2.0
SMB_INFO_SET_EAS	0x0002	Set a specific list of extended attributes (EAs).	LANMAN2.0
SMB_SET_FILE_BASIC_INFO	0x0101	Set 64-bit create, access, write, and change timestamps along with extended file attributes.	NT LANMAN
SMB_SET_FILE_DISPOSITION_INFO	0x0102	Set whether or not the file is marked for deletion.	NT LANMAN
SMB_SET_FILE_ALLOCATION_INFO	0x0103	Set file allocation size.	NT LANMAN
SMB_SET_FILE_END_OF_FILE_INFO	0x0104	Set file EOF offset.	NT LANMAN

In Section 2.2.8.4.3 SMB_SET_FILE_BASIC_INFO, the following was changed from:

Changed from:

This information level structure is used in TRANS2_SET_PATH_INFORMATION (section 2.2.6.7) requests to set the following information for the file specified in the request.<182>

Changed to:

This information level structure is used in TRANS2_SET_PATH_INFORMATION (section 2.2.6.7) and TRANS2_SET_FILE_INFORMATION (section 2.2.6.9) requests to set the following information for the file specified in the request.<182>

In Section 2.2.8.4.4 SMB_SET_FILE_DISPOSITION_INFO, the following was changed from:

Changed from:

This information level structure is used in TRANS2_SET_PATH_INFORMATION (section 2.2.6.7) requests to mark or unmark the file specified in the request for deletion.<183>

Changed to:

This information level structure is used in TRANS2_SET_PATH_INFORMATION (section 2.2.6.7) and TRANS2_SET_FILE_INFORMATION (section 2.2.6.9) requests to mark or unmark the file specified in

Errata Published*	Description
	the request for deletion.<183> In Section 2.2.8.4.5 SMB_SET_FILE_ALLOCATION_INFO, the following was changed from: Changed from: This information level structure is used in TRANS2_SET_PATH_INFORMATION (section 2.2.6.7) requests to set allocation size information for the file specified in the request.<184> Changed to: This information level structure is used in TRANS2_SET_PATH_INFORMATION (section 2.2.6.7) and TRANS2_SET_FILE_INFORMATION (section 2.2.6.9) requests to set allocation size information for the file specified in the request.<184> In Section 2.2.8.4.6 SMB_SET_FILE_END_OF_FILE_INFO, the following was changed from: Changed from: This information level structure is used in TRANS2_SET_PATH_INFORMATION (section 2.2.6.7) requests to set end-of-file information for the file specified in the request.<185> Changed to: This information level structure is used in TRANS2_SET_PATH_INFORMATION (section 2.2.6.7) and TRANS2_SET_FILE_INFORMATION (section 2.2.6.9) requests to set end-of-file information for the file specified in the request.<185> In Section 3.2.4.13 Application Requests Setting File Attributes, the following was added: When the Information Level is SMB_SET_FILE_ALLOCATION_INFO, the application provides: ● The allocation size of the file in bytes. When the Information Level is SMB_SET_FILE_END_OF_FILE_INFO, the application provides: ● The absolute new end-of-file position as a byte offset from the start of the file.
2020/06/22	In Section 3.1.4.1.1, Command Sequence Requirements, removed SMB_COM_SESSION_SETUP. Changed from: ● Unless otherwise noted, following a successful Protocol Negotiation an SMB_COM_SESSION_SETUP or SMB_COM_SESSION_SETUP_ANDX (section 2.2.4.53) command MUST be used to establish an SMB session before any other SMB commands are sent. Multiple SMB sessions can be set up per SMB connection. Changed to: ● Unless otherwise noted, following a successful Protocol Negotiation an SMB_COM_SESSION_SETUP_ANDX (section 2.2.4.53) command MUST be used to establish an SMB session before any other SMB commands are sent. Multiple SMB sessions can be set up per SMB

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>connection.</td></tr>
<tr><td>2020/06/22</td><td>In Section 2.2.2.3.3, QUERY Information Level Codes, revised description for SMB_QUERY_FILE_ALL_INFO changing SMB_FILE_QUERY_STANDARD_INFO to SMB_QUERY_FILE_STANDARD_INFO & SMB_FILE_EA_INFO to SMB_QUERY_FILE_EA_INFO.

Changed from:
<table><tr><td>SMB_QUERY_FILE_ALL_INFO</td><td>0x0107</td><td>Query the SMB_QUERY_FILE_BASIC_INFO, SMB_FILE_QUERY_STANDARD_INFO, SMB_FILE_EA_INFO, and SMB_QUERY_FILE_NAME_INFO data as well as access flags, access mode, and alignment information in a single request.</td><td>NT LANMAN</td></tr></table>
Changed to:
<table><tr><td>SMB_QUERY_FILE_ALL_INFO</td><td>0x0107</td><td>Query the SMB_QUERY_FILE_BASIC_INFO, SMB_QUERY_FILE_STANDARD_INFO, SMB_QUERY_FILE_EA_INFO, and SMB_QUERY_FILE_NAME_INFO data as well as access flags, access mode, and alignment information in a single request.</td><td>NT LANMAN</td></tr></table>
In Section 2.2.4.21, SMB_CON_WRITE_AND_UNLOCK, revised response field from ByteCountWritten to CountOfBytesWritten.

Changed from:
The write and unlock command has the effect of writing to a range of bytes and then unlocking them. This command is usually associated with an earlier usage of SMB_COM_LOCK_AND_READ (section 2.2.4.20) on the same range of bytes. The server's response field ByteCountWritten indicates the number of bytes actually written.

Changed to:
The write and unlock command has the effect of writing to a range of bytes and then unlocking them. This command is usually associated with an earlier usage of SMB_COM_LOCK_AND_READ (section 2.2.4.20) on the same range of bytes. The server's response field CountOfBytesWritten indicates the number of bytes actually written.

In Section 2.2.4.23.2, Response, revised description of DataLength changing SMB_COM_SESSION_SETUP_AND_X to SMB_COM_SESSION_SETUP_ANDX.

Changed from:
DataLength (2 bytes): This field is the number of bytes read and included in the response. The value of this field MUST NOT cause the message to exceed the client's maximum buffer size as specified in MaxBufferSize of the SMB_COM_SESSION_SETUP_AND_X (section 2.2.4.53) client request.

Changed to:
DataLength (2 bytes): This field is the number of bytes read and included in the response. The value of this field MUST NOT cause the message to exceed the client's maximum buffer size as specified in MaxBufferSize of the SMB_COM_SESSION_SETUP_ANDX (section 2.2.4.53) client request.</td></tr>
</table>

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
In Section 2.2.4.33.2, Response, corrected typo – SMB_COM_TRANSACTION.

Changed from:
The SMB_COM_TRANSACTON response has two possible formats.

Changed to:
The SMB_COM_TRANSACTION response has two possible formats.

In Section 2.2.4.64.1, Request, revised description of FILE_OPEN_BY_FILE_ID changing FileId to FID.

Changed from:
<table>
<tr><td>FILE_OPEN_BY_FILE_ID
0x00002000</td><td>Opens a file based on the FileId. If this option is set, the server MUST fail the request with STATUS_NOT_SUPPORTED in the Status field of the SMB Header in the server response.</td></tr>
</table>

Changed to:
<table>
<tr><td>FILE_OPEN_BY_FILE_ID
0x00002000</td><td>Opens a file based on the FID. If this option is set, the server MUST fail the request with STATUS_NOT_SUPPORTED in the Status field of the SMB Header in the server response.</td></tr>
</table>

In Section 2.2.7.1.1, Request, revised description of FILE_OPEN_BY_FILE_ID changing FileId to FID.

Changed from:
<table>
<tr><td>FILE_OPEN_BY_FILE_ID
0x00002000</td><td>Opens a file based on the FileId. If this option is set, the server MUST fail the request with STATUS_NOT_SUPPORTED in the Status field of the SMB Header in the server response.</td></tr>
</table>

Changed to:
<table>
<tr><td>FILE_OPEN_BY_FILE_ID
0x00002000</td><td>Opens a file based on the FID. If this option is set, the server MUST fail the request with STATUS_NOT_SUPPORTED in the Status field of the SMB Header in the server response.</td></tr>
</table>

In Section 2.2.8.1.7, SMB_FIND_FILE_BOTH_DIRECTORY_INFO, revised section 2.2.8.1.5 name from SMB_FILE_FULL_DIRECTORY_INFO to SMB_FIND_FILE_FULL_DIRECTORY_INFO.

Changed from:
This information level structure is used in TRANS2_FIND_FIRST2 (section 2.2.6.2) and TRANS2_FIND_NEXT2 (section 2.2.6.3) responses to return a combination of the SMB_FILE_FULL_DIRECTORY_INFO and SMB_FIND_FILE_NAMES_INFO (section 2.2.8.1.6) data for all files that match the request's search criteria.
</td></tr>
</table>

Errata Published*	Description
	Changed to: This information level structure is used in TRANS2_FIND_FIRST2 (section 2.2.6.2) and TRANS2_FIND_NEXT2 (section 2.2.6.3) responses to return a combination of the SMB_FIND_FILE_FULL_DIRECTORY_INFO (section 2.2.8.1.5) and SMB_FIND_FILE_NAMES_INFO (section 2.2.8.1.6) data for all files that match the request's search criteria. In Section 2.2.8.3.10, SMB_QUERY_FILE_ALL_INFO, revised packet names SMB_QUERY_FILE_STANDARD_INFO and SMB_QUERY_FILE_EA_IINFO. Changed from: This information level structure is used in TRANS2_QUERY_PATH_INFORMATION (section 2.2.6.6) and TRANS2_QUERY_FILE_INFORMATION (section 2.2.6.8) responses to return the SMB_QUERY_FILE_BASIC_INFO, SMB_FILE_QUERY_STANDARD_INFO, SMB_FILE_EA_INFO, and SMB_QUERY_FILE_NAME_INFO data as well as access flags, access mode, and alignment information in a single request for the file specified in the request. Changed to: This information level structure is used in TRANS2_QUERY_PATH_INFORMATION (section 2.2.6.6) and TRANS2_QUERY_FILE_INFORMATION (section 2.2.6.8) responses to return the SMB_QUERY_FILE_BASIC_INFO, SMB_QUERY_FILE_STANDARD_INFO, SMB_QUERY_FILE_EA_INFO, and SMB_QUERY_FILE_NAME_INFO data as well as access flags, access mode, and alignment information in a single request for the file specified in the request. In Section 3.2.1.4, Per Tree Connect, revised description of Client.TreeConnect.TreeID changing treeID to TreeID. Changed from: Client.TreeConnect.TreeID: The treeID (TID) that identifies this tree connect as returned by the server in the header of the SMB_COM_TREE_CONNECT Response (section 2.2.4.50.2) or the SMB_COM_TREE_CONNECT_ANDX Response (section 2.2.4.55.2). Changed to: Client.TreeConnect.TreeID: The TreeID (TID) that identifies this tree connect as returned by the server in the header of the SMB_COM_TREE_CONNECT Response (section 2.2.4.50.2) or the SMB_COM_TREE_CONNECT_ANDX Response (section 2.2.4.55.2). In Section 3.2.4.2.4, User Authentication, revised description changing UnicodePasswordLength to UnicodePasswordLen and OEMPasswordLength to OEMPasswordLen. Changed from: ● If the server supports Unicode (as indicated in Client.Connection.ServerCapabilities) the client MAY send the plaintext password in Unicode. The Unicode password is placed into the UnicodePassword field of the SMB_COM_SESSION_SETUP_ANDX Request as an array of bytes (not a null-terminated string). No alignment padding is used. The UnicodePasswordLength field is set to the length, in bytes, of the Unicode password. ● If neither the client nor the server supports Unicode, or the client sends the password in OEM character set format, the password is placed into the OEMPassword field of the SMB_COM_SESSION_SETUP_ANDX Request as an array of bytes (not a null-terminated string). The OEMPasswordLength field is set to the length, in bytes, of the password. ... The LAN Manager (LM) response and the LAN Manager version 2 (LMv2) response are mutually exclusive. The implementation MUST select either the LM or the LMv2 response and send it in the OEMPassword field of the SMB_COM_SESSION_SETUP_ANDX Request as an array of bytes (not a null-terminated string). The OEMPasswordLength field MUST be set to the length in bytes of the LM

Errata Published*	Description
	or LMv2 response. The NT LAN Manager (NTLM) response and the NT LAN Manager version 2 (NTLMv2) response are mutually exclusive. The implementation MUST select either the NTLM or the NTLMv2 response and send it in the UnicodePassword field of the SMB_COM_SESSION_SETUP_ANDX Request as an array of bytes (not a null-terminated string). The UnicodePasswordLength field MUST be set to the length, in bytes of the NTLM or NTLMv2 response. Changed to: ● If the server supports Unicode (as indicated in Client.Connection.ServerCapabilities) the client MAY send the plaintext password in Unicode. The Unicode password is placed into the UnicodePassword field of the SMB_COM_SESSION_SETUP_ANDX Request as an array of bytes (not a null-terminated string). No alignment padding is used. The UnicodePasswordLen field is set to the length, in bytes, of the Unicode password. ● If neither the client nor the server supports Unicode, or the client sends the password in OEM character set format, the password is placed into the OEMPassword field of the SMB_COM_SESSION_SETUP_ANDX Request as an array of bytes (not a null-terminated string). The OEMPasswordLen field is set to the length, in bytes, of the password. ... The LAN Manager (LM) response and the LAN Manager version 2 (LMv2) response are mutually exclusive. The implementation MUST select either the LM or the LMv2 response and send it in the OEMPassword field of the SMB_COM_SESSION_SETUP_ANDX Request as an array of bytes (not a null-terminated string). The OEMPasswordLen field MUST be set to the length in bytes of the LM or LMv2 response. The NT LAN Manager (NTLM) response and the NT LAN Manager version 2 (NTLMv2) response are mutually exclusive. The implementation MUST select either the NTLM or the NTLMv2 response and send it in the UnicodePassword field of the SMB_COM_SESSION_SETUP_ANDX Request as an array of bytes (not a null-terminated string). The UnicodePasswordLen field MUST be set to the length, in bytes of the NTLM or NTLMv2 response. In Section 3.2.5.1, Receiving Any Message, revised description changing SMB_COM_RAW_READ to SMB_COM_READ_RAW. Changed from: If an SMB_COM_RAW_READ is in progress and the message is a raw data transfer, the message MUST be handled as described in section 3.2.5.16. Changed to: If an SMB_COM_READ_RAW is in progress and the message is a raw data transfer, the message MUST be handled as described in section 3.2.5.16. In Section 3.3.5.10, Receiving, corrected typos – TypeofLock to TypeOfLock & NewOplockLevel to NewOpLockLevel. Changed from: ● If another process has the file open, and that process has an OpLock on the file, and the process has asked for extended notification (Batch OpLock), the rename request MUST block until the server has sent an OpLock break request to the owner of the OpLock, as specified in section 3.3.4.2, and either received a response or the OpLock break time-out has expired.<259> The server MUST have the OPLOCK_RELEASE flag set in the TypeofLock field of the request. The server MUST set the NewOplockLevel field of the request to 0x00. If the process holding the OpLock closes the file (thus freeing the OpLock) the rename takes place. If not, the rename MUST fail with STATUS_SHARING_VIOLATION.

Errata Published*	Description
	Changed to: ● If another process has the file open, and that process has an OpLock on the file, and the process has asked for extended notification (Batch OpLock), the rename request MUST block until the server has sent an OpLock break request to the owner of the OpLock, as specified in section 3.3.4.2, and either received a response or the OpLock break time-out has expired. The server MUST have the OPLOCK_RELEASE flag set in the TypeOfLock field of the request. The server MUST set the NewOpLockLevel field of the request to 0x00. If the process holding the OpLock closes the file (thus freeing the OpLock) the rename takes place. If not, the rename MUST fail with STATUS_SHARING_VIOLATION. In Section 3.3.5.24, Receiving an SMB_COM_READ_RAW, revised description changing BytesToReturn to MaxCountOfBytesToReturn. Changed from: ● The server MUST attempt to read from the underlying object store for the file indicated by the FID in the response. It MUST start reading from the file at the offset indicated by the Offset field in the request, or by the combination of Offset and OffsetHigh if CAP_LARGE_FILES was negotiated. The client MUST read BytesToReturn bytes or until EOF, whichever comes first. Changed to: ● The server MUST attempt to read from the underlying object store for the file indicated by the FID in the response. It MUST start reading from the file at the offset indicated by the Offset field in the request, or by the combination of Offset and OffsetHigh if CAP_LARGE_FILES was negotiated. The client MUST read MaxCountOfBytesToReturn bytes or until EOF, whichever comes first. In Section 3.3.5.43, Receiving an SMB_COM_SESSION_SETUP_ANDX Request, revised description changing IdleTime to Server.Connection.IdleTime. Changed from: ● The server MUST set CreationTime and IdleTime to be current time. Changed to: ● The server MUST set CreationTime and Server.Connection.IdleTime to be current time. In Section 3.3.5.53, Receiving an SMB_COM_NT_RENAME Request, revised description changing SMB_NT_RENAME_RENAME FILE to SMB_NT_RENAME_RENAME_FILE. Changed from: ● If the InformationLevel field value is neither SMB_NT_RENAME_RENAME FILE (0x104) nor SMB_NT_RENAME_SET_LINK_INFO (0x103), the server SHOULD fail the request with STATUS_INVALID_SMB (ERRSRV/ERRerror). Changed to: ● If the InformationLevel field value is neither SMB_NT_RENAME_RENAME_FILE (0x104) nor SMB_NT_RENAME_SET_LINK_INFO (0x103), the server SHOULD fail the request with STATUS_INVALID_SMB (ERRSRV/ERRerror).

*Date format: YYYY/MM/DD

[MS-CMRP]: Failover Cluster: Management API (ClusAPI) Protocol

This topic lists the Errata found in the MS-CMRP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

June 1, 2017 - Download

March 16, 2018 - Download

September 12, 2018 - Download

March 4, 2020 - Download

Errata below are for Protocol Document Version V37.0 – 2020/03/04.

Errata Published *	Description

2020/07/20	In Section 3.1.4.2.129, ApiOnlineGroupEx (Opnum 130), revised processing rules.
	Changed from:
	• If the CLUSAPI_GROUP_ONLINE_IGNORE_RESOURCE_STATUS flag is set in the dwOnlineFlags parameter, the server MUST ignore the locked mode value of the group designated by the hGroup parameter.
	• For each resource contained in the group designated by the hGroup parameter that is not in the ClusterResourceOnline state (section 3.1.4.2.13), the server MUST provide the buffer specified by the lpInBuffer parameter to the server implementation-specific object that controls the resource operation while bringing the resource online.
	The server MUST accept an ApiOnlineGroupEx request only if it is in the read/write state, as specified in section 3.1.1.
	The server MUST require that the access level associated with the hGroup parameter is "All" (section 3.1.4).
	
	`error_status_t ApiOnlineGroupEx(`
	`  [in] HGROUP_RPC hGroup,`
	`  [in] DWORD dwOnlineFlags,`
	`  [in, size_is(cbInBufferSize)] BYTE* lpInBuffer,`
	`  [in] DWORD cbInBufferSize,`
	`  [out] error_status_t *rpc_status`
	`);`
	
	hGroup: An HGROUP_RPC context handle that was obtained in a previous call to ApiOpenGroup (section 3.1.4.2.42), ApiOpenGroupEx (section 3.1.4.2.118), or ApiCreateGroup (section 3.1.4.2.43).
	dwOnlineFlags: Either CLUSAPI_GROUP_ONLINE_IGNORE_RESOURCE_STATUS (0x00000001), if the client needs the server to ignore the locked mode for the group specified by the hGroup

Errata Published *	Description
	parameter (section 3.1.1.1.4), or zero. lpInBuffer: A pointer to a buffer that the server will provide to implementation-specific objects that control the resource operations for each resource in the group. The client SHOULD set this parameter to a PROPERTY_LIST (section 2.2.3.10). For each value in this list, the client SHOULD set the property name to the name of the resource type of one of the resources in the group. The client MAY provide a buffer that does not have a property value corresponding to each resource type in the group, and the client MAY provide a buffer that has multiple property values for the same resource type. Except for the following property values, the server MUST treat all property values provided by the client identically.

Property Name	CLUSTER_PROPERTY_FORMAT	Value	Description
Virtual Machine	CLUSPROP_FORMAT_DWORD	0x00000004	Reserved for local use. cbInBufferSize: The size in bytes of the buffer pointed to by the lpInBuffer parameter.

rpc_status: A 32-bit integer used to indicate success or failure. The RPC runtime MUST indicate, by writing to this parameter, whether it succeeded in executing this method on the server. The encoding of the value passed in this parameter MUST conform to encoding for comm_status and fault_status, as specified in Appendix E of [C706].

Return Values: This method MUST return the same error codes as specified for ApiOnlineGroup (section 3.1.4.2.50).

Changed to:

• If the CLUSAPI_GROUP_ONLINE_IGNORE_RESOURCE_STATUS flag is set in the dwOnlineFlags parameter, the server MUST ignore the locked mode value of the group designated by the hGroup parameter.

• If the CLUSAPI_GROUP_ONLINE_SYNCHRONOUS flag is set in the dwOnlineFlags parameter, the server MUST perform the operation synchronously to bring the group designated by the hGroup parameter online.

• If the CLUSAPI_GROUP_ONLINE_BEST_POSSIBLE_NODE flag is set in the dwOnlineFlags parameter, the server MUST determine the best possible node that will host the group designated by the hGroup parameter.

• If the CLUSAPI_GROUP_ONLINE_IGNORE_AFFINITY_RULE flag is set in the dwOnlineFlags parameter, the server MUST ignore the affinity rule of the group designated by the hGroup parameter.

• For each resource contained in the group designated by the hGroup parameter that is not in the ClusterResourceOnline state (section 3.1.4.2.13), the server MUST provide the buffer specified by the lpInBuffer parameter to the server implementation-specific object that controls the resource operation while bringing the resource online.

The server MUST accept an ApiOnlineGroupEx request only if it is in the read/write state, as specified in section 3.1.1.

The server MUST require that the access level associated with the hGroup parameter is "All" (section 3.1.4).

```
error_status_t ApiOnlineGroupEx(
  [in] HGROUP_RPC hGroup,
  [in] DWORD dwOnlineFlags,
  [in, size_is(cbInBufferSize)] BYTE* lpInBuffer,
  [in] DWORD cbInBufferSize,
  [out] error_status_t *rpc_status
```

Errata Published *	Description
	); hGroup: An HGROUP_RPC context handle that was obtained in a previous call to ApiOpenGroup (section 3.1.4.2.42), ApiOpenGroupEx (section 3.1.4.2.118), or ApiCreateGroup (section 3.1.4.2.43). dwOnlineFlags: A bitwise-OR of zero or more of the following flags.

Value	Description
0x00000001 CLUSAPI_GROUP_ONLINE_IGNORE_RESOURCE_STATUS	The server MUST ignore the locked mode of the group as specified in section 3.1.1.1.4.
0x00000002 CLUSAPI_GROUP_ONLINE_SYNCHRONOUS	The server MUST perform the operation synchronously to bring the group online.<114>
0x00000004 CLUSAPI_GROUP_ONLINE_BEST_POSSIBLE_NODE	The server MUST determine the best possible node that will host the group when it is brought online.<115>
0x00000008 CLUSAPI_GROUP_ONLINE_IGNORE_AFFINITY_RULE	The server MUST ignore the affinity rule of the group.<116> lpInBuffer: A pointer to a buffer that the server will provide to implementation-specific objects that control the resource operations for each resource in the group. The client SHOULD set this parameter to a PROPERTY_LIST (section 2.2.3.10) . For each value in this list, the client SHOULD set the property name to the name of the resource type of one of the resources in the group. The client MAY provide a buffer that does not have a property value corresponding to each resource type in the group, and the client MAY provide a buffer that has multiple property values for the same resource type. Except for the following property values, the server MUST treat all property values provided by the client identically.

Property Name	CLUSTER_PROPERTY_FORMAT	Value	Description
Virtual Machine	CLUSPROP_FORMAT_DWORD	0x00000004	Reserved for local use. cbInBufferSize: The size in bytes of the buffer pointed to by the lpInBuffer parameter.

<table>
<tr><th>Errata Published *</th><th>Description</th></tr>
<tr><td></td><td>
rpc_status: A 32-bit integer used to indicate success or failure. The RPC runtime MUST indicate, by writing to this parameter, whether it succeeded in executing this method on the server. The encoding of the value passed in this parameter MUST conform to encoding for comm_status and fault_status, as specified in Appendix E of [C706].
Return Values: This method MUST return the same error codes as specified for ApiOnlineGroup (section 3.1.4.2.50), in addition to the following return value.
<table>
<tr><th>Return value/code</th><th>Description</th></tr>
<tr><td>0x00000057
ERROR_INVALID_PARAMETER</td><td>The dwOnlineFlags parameter is not one of the specified values.</td></tr>
</table>

In Section 3.1.4.2.131, ApiMoveGroupEx (Opnum 132), revised processing rules, adding additional value to the dwMoveFlags field.

Changed from:
● If the CLUSAPI_GROUP_MOVE_HIGH_PRIORITY_START flag is set in the dwMoveFlags parameter, then on the destination node when bringing the group to its persistent state, the server SHOULD bring this group to its persistent state as soon as possible, regardless of other implementation-specific policies that govern the ordering and/or prioritization of bringing groups to their persistent states.
● For each resource contained in the group designated by hGroup that is in the state ClusterResourceOnline (section 3.1.4.2.13), the server MUST provide the buffer designated by the lpInBuffer parameter to the server implementation-specific object that controls the resource operation while bringing the resource offline on the current node and when bringing the resource online on the destination node. How the server provides this buffer is implementation-specific.
&
<table>
<tr><td>0x00000008
CLUSAPI_GROUP_MOVE_HIGH_PRIORITY_START</td><td>When bringing the group to its persistent state on the destination node, the server SHOULD bring this group to its persistent state as soon as possible without regard to implementation-specific policies that govern the ordering and/or prioritization of bringing groups to their persistent states.</td></tr>
</table>

Changed to:
● If the CLUSAPI_GROUP_MOVE_HIGH_PRIORITY_START flag is set in the dwMoveFlags parameter, then on the destination node when bringing the group to its persistent state, the server SHOULD bring this group to its persistent state as soon as possible, regardless of other implementation-specific policies that govern the ordering and/or prioritization of bringing groups to their persistent states.
● If the CLUSAPI_GROUP_MOVE_FAILBACK flag is set in the dwMoveFlags parameter, and if move group operation fails, the server MUST perform failback operation.
● If the CLUSAPI_GROUP_MOVE_IGNORE_AFFINITY_RULE flag is set in the dwMoveFlags parameter, the server MUST ignore the affinity rule of the group designated by the hGroup parameter.
● For each resource contained in the group designated by hGroup that is in the state ClusterResourceOnline (section 3.1.4.2.13), the server MUST provide the buffer designated by the lpInBuffer parameter to the server implementation-specific object that controls the resource operation while bringing the resource offline on the current node and when bringing the resource online on the destination node. How the server provides this buffer is implementation-specific.
</td></tr>
</table>

Errata Published *	Description
	& 0x00000008 CLUSAPI_GROUP_MOVE_HIGH_PRIORITY_START — When bringing the group to its persistent state on the destination node, the server SHOULD bring this group to its persistent state as soon as possible without regard to implementation-specific policies that govern the ordering and/or prioritization of bringing groups to their persistent states. CLUSAPI_GROUP_MOVE_FAILBACK 0x00000010 — If move group operation fails, the server MUST perform failback operation. CLUSAPI_GROUP_MOVE_IGNORE_AFFINITY_RULE 0x00000020 — The server MUST ignore the affinity rule while performing move group operation.<119> In Section 3.1.4.2.135, ApiOfflineResourceEx (Opnum 136), revised processing rules, adding dwOfflineFlags field values and a new return value. Changed from: ● If the CLUSAPI_RESOURCE_OFFLINE_IGNORE_RESOURCE_STATUS flag is set in the dwOfflineFlags parameter, the server MUST ignore the locked mode value of the resource designated by the hResource parameter as well as the locked mode value of any of its dependent resources as specified in section 3.1.1.1.2. ● If the resource designated by the hResource parameter is in the ClusterResourceOnline state (section 3.1.4.2.13), then the server MUST provide the buffer designated by the lpInBuffer parameter to the server implementation-specific object that controls the resource operation while bringing the resource offline. The server MUST also provide this buffer to the server implementation-specific objects for any of the designated resource's dependent resources that are also in the ClusterResourceOnline state. How the server provides this buffer is implementation-specific. & dwOfflineFlags: The value CLUSAPI_RESOURCE_OFFLINE_IGNORE_RESOURCE_STATUS (0x00000001), if the client needs the server to ignore the resource locked mode as described in 3.1.1.1.1, or zero. & Return Values: This method MUST return the same error codes returned by the ApiOfflineResource (section 3.1.4.2.19) method. Changed to: ● If the CLUSAPI_RESOURCE_OFFLINE_IGNORE_RESOURCE_STATUS flag is set in the dwOfflineFlags parameter, the server MUST ignore the locked mode value of the resource designated by the hResource parameter as well as the locked mode value of any of its dependent resources as specified in section 3.1.1.1.2. ● If the CLUSAPI_RESOURCE_OFFLINE_FORCE_WITH_TERMINATION flag is set in the dwOfflineFlags parameter, the server MUST shut down the resource designated by the hResource parameter. ● If the CLUSAPI_RESOURCE_OFFLINE_DO_NOT_UPDATE_PERSISTENT_STATE flag is set in the dwOfflineFlags parameter, the server MUST not update the persistent state of the resource designated by the hResource parameter when it is brought offline. ● If the resource designated by the hResource parameter is in the ClusterResourceOnline state (section 3.1.4.2.13), then the server MUST provide the buffer designated by the lpInBuffer

<table>
<tr><th>Errata Published *</th><th>Description</th></tr>
<tr><td></td><td>
parameter to the server implementation-specific object that controls the resource operation while bringing the resource offline. The server MUST also provide this buffer to the server implementation-specific objects for any of the designated resource's dependent resources that are also in the ClusterResourceOnline state. How the server provides this buffer is implementation-specific.
&
dwOfflineFlags: A bitwise-OR of zero or more of the following flags.
<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>0x00000001
CLUSAPI_RESOURCE_OFFLINE_IGNORE_RESOURCE_STATUS</td><td>The server MUST ignore the locked mode value of the resource as well as the locked mode value of any of its dependent resources as specified in section 3.1.1.1.2.</td></tr>
<tr><td>0x00000002
CLUSAPI_RESOURCE_OFFLINE_FORCE_WITH_TERMINATION</td><td>The server MUST shut down the resource.</td></tr>
<tr><td>0x00000004
CLUSAPI_RESOURCE_OFFLINE_DO_NOT_UPDATE_PERSISTENT_STATE</td><td>The server MUST not update the persistent state of the resource when it is brought offline.<125>
&</td></tr>
</table>
Return Values: This method MUST return the same error codes returned by the ApiOfflineResource (section 3.1.4.2.19) method, in addition to the following return value.
<table>
<tr><th>Return value/code</th><th>Description</th></tr>
<tr><td>0x00000057
ERROR_INVALID_PARAMETER</td><td>The dwOfflineFlags parameter is not one of the specified values.</td></tr>
</table>
In Section 7, Appendix B: Product Behavior, added version exceptions.

Changed from:
<112> Section 3.1.4.2.128: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1).
<113> Section 3.1.4.2.129: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1).
<114> Section 3.1.4.2.130: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1).
<115> Section 3.1.4.2.131: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1).
<116> Section 3.1.4.2.132: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1).
</td></tr>
</table>

Errata Published *	Description
	<117> Section 3.1.4.2.133: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1). <118> Section 3.1.4.2.134: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1). <119> Section 3.1.4.2.134: Windows Server v1909 and earlier operating systems do not support this value. <120> Section 3.1.4.2.135: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1). <121> Section 3.1.4.2.137: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1). Changed to: <112> Section 3.1.4.2.128: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1). <113> Section 3.1.4.2.129: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1). <114> Section 3.1.4.2.129: Windows Server 2012 R2 operating system and earlier operating systems do not support this value. <115> Section 3.1.4.2.129: Windows Server 2012 R2 and earlier operating systems do not support this value. <116> Section 3.1.4.2.129: Windows Server v1909 and earlier operating systems do not support this value. <117> Section 3.1.4.2.130: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1). <118> Section 3.1.4.2.131: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1). <119> Section 3.1.4.2.131: Windows Server v1909 and earlier operating systems do not support this value. <120> Section 3.1.4.2.132: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1). <121> Section 3.1.4.2.133: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1). <122> Section 3.1.4.2.134: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1). <123> Section 3.1.4.2.134: Windows Server v1909 and earlier operating systems do not support this value. <124> Section 3.1.4.2.135: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1). <125> Section 3.1.4.2.135: Windows Server 2016 and earlier operating systems do not support this value. <126> Section 3.1.4.2.137: Windows Server 2008 and Windows Server 2008 R2 do not support this method and fail calls with RPC_S_PROCNUM_OUT_OF_RANGE (0x000006D1).
2020/06/22	In Section 3.1.4.2.134, ApiOnlineResourceEx (Opnum 135), added processing rules, revised dwOnlineFlags field adding a value table, and added a return table to the Return Values field. Changed from: The server MUST handle this method in the same manner as ApiOnlineResource (section 3.1.4.2.18) except as follows: ● If the CLUSAPI_RESOURCE_ONLINE_IGNORE_RESOURCE_STATUS flag is set in the dwOnlineFlags parameter, the server MUST ignore the locked mode value of the resource designated by the hResource parameter as well as the locked mode value of any of its provider resources as specified in section 3.1.1.1.2.

<table>
<tr><th>Errata Published *</th><th>Description</th></tr>
<tr><td></td><td>
• If the resource designated by hResource is not already in the ClusterResourceOnline state (section 3.1.4.2.13), the server MUST provide the buffer designated by the lpInBuffer parameter to the server implementation-specific object that controls the resource operation while bringing the resource online and MUST provide this buffer to the server implementation-specific objects for any of the designated resource's provider resources that are not already in the ClusterResourceOnline state. How the server provides this buffer is implementation-specific.
The server accepts an ApiOnlineResourceEx request only if it is in the read/write state, as specified in section 3.1.1.

Changed to:
The server MUST handle this method in the same manner as ApiOnlineResource (section 3.1.4.2.18) except as follows:
• If the CLUSAPI_RESOURCE_ONLINE_IGNORE_RESOURCE_STATUS flag is set in the dwOnlineFlags parameter, the server MUST ignore the locked mode value of the resource designated by the hResource parameter as well as the locked mode value of any of its provider resources as specified in section 3.1.1.1.2.
• If the CLUSAPI_RESOURCE_ONLINE_DO_NOT_UPDATE_PERSISTENT_STATE flag is set in the dwOnlineFlags parameter, the server MUST not update the persistent state of the resource designated by the hResource parameter.
• If the CLUSAPI_RESOURCE_ONLINE_NECESSARY_FOR_QUORUM flag is set in the dwOnlineFlags parameter, the server MUST bring the resource designated by the hResource parameter to online to maintain a quorum.
• If the CLUSAPI_RESOURCE_ONLINE_BEST_POSSIBLE_NODE flag is set in the dwOnlineFlags parameter, the server MUST determine the best possible node that will host the resource designated by the hResource parameter.
• If the CLUSAPI_RESOURCE_ONLINE_IGNORE_AFFINITY_RULE flag is set in the dwOnlineFlags parameter, the server MUST ignore the affinity rule of the resource designated by the hResource parameter.
• If the resource designated by hResource is not already in the ClusterResourceOnline state (section 3.1.4.2.13), the server MUST provide the buffer designated by the lpInBuffer parameter to the server implementation-specific object that controls the resource operation while bringing the resource online and MUST provide this buffer to the server implementation-specific objects for any of the designated resource's provider resources that are not already in the ClusterResourceOnline state. How the server provides this buffer is implementation-specific.
The server accepts an ApiOnlineResourceEx request only if it is in the read/write state, as specified in section 3.1.1.

In this same section:

Changed from:
dwOnlineFlags: The value CLUSAPI_RESOURCE_ONLINE_IGNORE_RESOURCE_STATUS, if the client needs the server to ignore the Resource locked mode as described in 3.1.1.1.1, or zero.

Changed to:
dwOnlineFlags: A bitwise-OR of zero or more of the following flags.
<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>0x00000001
CLUSAPI_RESOURCE_ONLINE_IGNORE_RESOURCE_STATUS</td><td>The server MUST ignore the resource locked mode as specified in section 3.1.1.1.1.</td></tr>
<tr><td>0x00000002</td><td>The server MUST</td></tr>
</table>
</td></tr>
</table>

Errata Published *	Description

CLUSAPI_RESOURCE_ONLINE_DO_NOT_UPDATE_PERSISTENT_STATE	not update the persistent state of the resource.
0x00000004 CLUSAPI_RESOURCE_ONLINE_NECESSARY_FOR_QUORUM	The server MUST bring the resource to online to maintain a quorum.
0x00000008 CLUSAPI_RESOURCE_ONLINE_BEST_POSSIBLE_NODE	The server MUST determine the best possible node that will host the resource.
0x00000020 CLUSAPI_RESOURCE_ONLINE_IGNORE_AFFINITY_RULE	The server MUST ignore the affinity rule of the resource.<119>

In this same section:

Changed from:

Return Values: This method MUST return the same error codes as returned by the ApiOnlineResource (section 3.1.4.2.18) method.

Changed to

Return Values: This method MUST return the same error codes as returned by the ApiOnlineResource (section 3.1.4.2.18) method, except for the following additional return value.

Return value/code	Description
0x00000057 ERROR_INVALID_PARAMETER	The dwOnlineFlags parameter is not one of the specified values.

In Section 3.1.4.2.143, ApiGetNotifyAsync (Opnum 147), revised the value description for the ERROR_INVLAID_FUNCTION value.

Changed from:

Return Values: This method MUST return one of the following values.

Return value/code	Description
0x00000000 ERROR_SUCCESS	The method completed successfully.
0x00000006 ERROR_INVALID_HANDLE	The data that is pointed to by the hNotify parameter does not represent a valid HNOTIFY_RPC context handle.
0x00000103	The notification port represented by the hNotify parameter has

Errata Published *	Description

ERROR_NO_MORE_ITEMS	been closed.
0x00000001 ERROR_INVALID_FUNCTION	Either the ApiUnblockedGetNotificationCall (section 3.1.4.2.107) method or the ApiCloseNotify (section 3.1.4.2.57) method has been called in another thread. The client SHOULD terminate the notification thread.

Changed to:

Return Values: This method MUST return one of the following values.

Return value/code	Description
0x00000000 ERROR_SUCCESS	The method completed successfully.
0x00000006 ERROR_INVALID_HANDLE	The data that is pointed to by the hNotify parameter does not represent a valid HNOTIFY_RPC context handle.
0x00000103 ERROR_NO_MORE_ITEMS	The notification port represented by the hNotify parameter has been closed.
0x00000001 ERROR_INVALID_FUNCTION	Either the ApiUnblockGetNotifyCall (section 3.1.4.2.107) method or the ApiCloseNotify (section 3.1.4.2.57) method has been called in another thread. The client SHOULD terminate the notification thread.

In Section 3.1.4.2.163, ApiCreateNetInterfaceEnum (Opnum 181), revised ApiCreateNetInterfaceEnums to ApiCreateNetInterfaceEnum.

Changed from:

```
error_status_t
ApiCreateNetInterfaceEnums {
  [ in ] HCLUSTER_RPC hCluster,
  [ in, string ] LPCWSTR lpszNodeName,
  [ in, string ] LPCWSTR lpszNetworkName,
  [ out ] PENUM LIST *ReturnEnum,
  [ out ] error_status_t *rpc_status
};
```

Changed to:

```
error_status_t
ApiCreateNetInterfaceEnum {
  [ in ] HCLUSTER RPC hCluster,
  [ in, string ] LPCWSTR lpszNodeName,
  [ in, string ] LPCWSTR lpszNetworkName,
  [ out ] PENUM_LIST *ReturnEnum,
```

Errata Published *	Description
	[out] error_status_t *rpc_status }; In Section 3.1.4.3.1.63, CLUSCTL_RESOURCE_NETNAME_SET_PWD_INFOEX, added space between CLUS_NETNAME_PWD_INFOEX and structure. Changed from: If nInBufferSize is less than the size of CLUS_NETNAME_PWD_INFOEXstructure, the server MUST fail the request with ERROR_INVALID_PARAMETER. If the length of the new password in Password field in CLUS_NETNAME_PWD_INFOEXstructure is greater than 127, the server MUST fail the request with ERROR_PASSWORD_RESTRICTION. Changed to: If nInBufferSize is less than the size of CLUS_NETNAME_PWD_INFOEX structure, the server MUST fail the request with ERROR_INVALID_PARAMETER. If the length of the new password in Password field in CLUS_NETNAME_PWD_INFOEX structure is greater than 127, the server MUST fail the request with ERROR_PASSWORD_RESTRICTION. In Section 3.1.4.3.7.19, CLUSCTL_CLUSTER_CLEAR_UPGRADE_IN_PROGRESS, revised CLUSCTL_CLUSTER_CLEAR UPGRADE_IN_PROGRESS to CLUSCTL_CLUSTER_CLEAR_UPGRADE_IN_PROGRESS. Changed from: The CLUSCTL_CLUSTER_CLEAR UPGRADE_IN_PROGRESS control code SHOULD<206> be used to indicate that the current upgrade to the cluster operational version is no longer in progress. Changed to: The CLUSCTL_CLUSTER_CLEAR_UPGRADE_IN_PROGRESS control code SHOULD<207> be used to indicate that the current upgrade to the cluster operational version is no longer in progress. In Section 3.1.4.3.7.22, CLUSCTL_CLUSTER_SET_DNS_DOMAIN, revised CLUSTER_SET_DNS_DOMAIN to CLUSCTL_CLUSTER_SET_DNS_DOMAIN. Changed from: The server MUST accept a CLUSCTL_CLUSTER_SET_DNS_DOMAIN cluster control code request if its protocol server state is in the read/write state, as specified in section 3.1.1. The server MUST not accept a CLUSTER_SET_DNS_DOMAIN cluster control code request if its protocol server state is in the read only state. Changed to: The server MUST accept a CLUSCTL_CLUSTER_SET_DNS_DOMAIN cluster control code request if its protocol server state is in the read/write state, as specified in section 3.1.1. The server MUST not accept a CLUSCTL_CLUSTER_SET_DNS_DOMAIN cluster control code request if its protocol server state is in the read only state. In Section 3.2.4.4, ClusterNodes, Cluster Networks, and Cluster Network Interfaces, revised ApiCreaeNetInterfaceEnumEx to ApiCreateNetInterfaceEnum. Changed from:

<table>
<tr><th>Errata Published *</th><th>Description</th></tr>
<tr><td></td><td>• Enumerate the cluster network interfaces associated with this cluster network: ApiCreateNetworkEnum (section 3.1.4.1.85 for protocol version 2 or 3.1.4.2.85 for protocol version 3), or ApiCreateNetInterfaceEnumEx (section 3.1.4.2.163) for protocol version 3.

Changed to:
• Enumerate the cluster network interfaces associated with this cluster network: ApiCreateNetworkEnum (section 3.1.4.1.85 for protocol version 2 or 3.1.4.2.85 for protocol version 3), or ApiCreateNetInterfaceEnum (section 3.1.4.2.163) for protocol version 3.</td></tr>
<tr><td>2020/06/08</td><td>In Section 3.1.4.2.143, ApiGetNotifyAsync (Opnum 147), in the table for Return Values, changed from:

<table><tr><td>0x00000001
ERROR_INVALID_FUNCTION</td><td>Either the ApiUnblockedGetNotificationCall (section 3.1.4.2.107) method or the ApiCloseNotify (section 3.1.4.2.57) method has been called in another thread. The client SHOULD terminate the notification thread.</td></tr></table>
Changed to:

<table><tr><td>0x00000001
ERROR_INVALID_FUNCTION</td><td>Either the ApiUnblockGetNotifyCall (section 3.1.4.2.107) method or the ApiCloseNotify (section 3.1.4.2.57) method has been called in another thread. The client SHOULD terminate the notification thread.</td></tr></table>

In Section 3.1.4.3.7.22, CLUSCTL_CLUSTER_SET_DNS_DOMAIN, changed from:

The server MUST accept a CLUSCTL_CLUSTER_SET_DNS_DOMAIN cluster control code request if its protocol server state is in the read/write state, as specified in section 3.1.1. The server MUST not accept a CLUSTER_SET_DNS_DOMAIN cluster control code request if its protocol server state is in the read only state.

Changed to:

The server MUST accept a CLUSCTL_CLUSTER_SET_DNS_DOMAIN cluster control code request if its protocol server state is in the read/write state, as specified in section 3.1.1. The server MUST not accept a CLUSCTL_CLUSTER_SET_DNS_DOMAIN cluster control code request if its protocol server state is in the read only state.</td></tr>
</table>

*Date format: YYYY/MM/DD

[MS-COMA]: Component Object Model Plus (COMplus) Remote Administration Protocol

This topic lists the Errata found in the MS-COMA document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-CRTD]: Certificate Templates Structure

This topic lists the Errata found in [MS-CRTD] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

Errata below are for Protocol Document Version V24.0 – 2018/09/12.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td>2019/12/16</td><td>In Section 2.26, msPKI-Enrollment-Flag Attribute, added missing 'CT_FLAG_SKIP_AUTO_RENEWAL' flag and description to the enrollment flags table.

Changed from:

<table>
<tr><td>0x00020000
CT_FLAG_ISSUANCE_POLICIES_FROM_REQUEST</td><td>This flag indicates that the certificate issuance policies to be included in the issued certificate come from the request rather than from the template. The template contains a list of all of the issuance policies that the request is allowed to specify; if the request contains policies that are not listed in the template, then the request is rejected. For the processing rules of this flag, see [MS-WCCE] section 3.2.2.6.2.1.4.5.8.<33></td></tr>
</table>

Changed to:

<table>
<tr><td>0x00020000
CT_FLAG_ISSUANCE_POLICIES_FROM_REQUEST</td><td>This flag indicates that the certificate issuance policies to be included in the issued certificate come from the request rather than from the template. The template contains a list of all of the issuance policies that the request is allowed to specify; if the request contains policies that are not listed in the template, then the request is rejected. For the processing rules of this flag, see [MS-WCCE] section 3.2.2.6.2.1.4.5.8.<33></td></tr>
<tr><td>0x00040000
CT_FLAG_SKIP_AUTO_RENEWAL</td><td>This flag indicates that the certificate should not be auto-renewed, although it has a valid template.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
In Section 2.27, msPKI-Private-Key-Flag Attribute, added missing 'CT_FLAG_HELLO_LOGON_KEY' flag and description to the private key flags table.

Changed from:

<table>
<tr><td>0x00000800 *
CT_FLAG_EK_VALIDATE_KEY</td><td>This flag indicates that attestation based on the hardware key of the TPM is to be performed. For more details, see [MS-WCCE] section 3.2.2.6.2.1.4.5.7.</td></tr>
</table>

Changed to:

<table>
<tr><td>0x00000800 *
CT_FLAG_EK_VALIDATE_KEY</td><td>This flag indicates that attestation based on the hardware key of the TPM is to be performed. For more details, see [MS-WCCE] section 3.2.2.6.2.1.4.5.7.</td></tr>
<tr><td>0x00200000 *
CT_FLAG_HELLO_LOGON_KEY</td><td>This flag indicates that the key is used for Windows Hello logon. For more details, see [MS-WCCE] section 3.2.2.6.2.1.4.5.7.</td></tr>
</table>
</td></tr>
</table>

*Date format: YYYY/MM/DD

[MS-CSRA]: Certificate Services Remote Administration Protocol

This topic lists the Errata found in the MS-CSRA document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

September 12, 2018 - Download

Errata below are for Protocol Document Version V38.0 – 2018/09/12.

Errata Published*	Description
2020/08/17	In Section 3.1.4.2.14 ICertAdminD2::GetConfigEntry (Opnum 44), revised the processing rules for the second and sixth Input Parameters in the table of this section to clarify that the value of the Authority parameter determines whether one of two registry keys is loaded and subsequently whether a Get or Set operation is performed on the relevant loaded key. Changed from: pwszNodePath is EMPTY and pwszEntry is "SetupStatus" pwszNodePath is EMPTY and pwszEntry is "Version" Changed to: pwszAuthority is EMPTY and pwszNodePath is EMPTY and pwszEntry is "SetupStatus". pwszAuthority is EMPTY and pwszNodePath is EMPTY and pwszEntry is "Version" ***** In 3.1.4.2.15 ICertAdminD2::SetConfigEntry (Opnum 45), revised the processing rules for the second and sixth Input Parameters in the table of this section for the SetConfigEntry method, to clarify that the value of the Authority parameter determines whether one of two registry keys is loaded and subsequently whether a Get or Set operation is performed on the relevant loaded key. Changed from: pwszNodePath is EMPTY and pwszEntry is "SetupStatus"

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>pwszNodePath is EMPTY and pwszEntry is "Version"

Changed to:

pwszAuthority is EMPTY and pwszNodePath is EMPTY and pwszEntry is "SetupStatus".

pwszAuthority is EMPTY andpwszNodePath is EMPTY and pwszEntry is "Version"</td></tr>
<tr><td>2020/08/17</td><td>In Section 3.1.4.2.14 ICertAdminD2::GetConfigEntry (Opnum 44), removed the row containing 'pwszNodePath is EMPTY and pwszEntry is OCSPURLs' in two tables, as OCSPURLs is an invalid entry.

Changed from:

<table>
<tr><th>Input Parameters</th><th>Processing rule for pVariant</th></tr>
<tr><td>pwszNodePath is EMPTY and pwszEntry is "CACertPublicationURLs"</td><td>The CA MUST use the values of the following ADM elements to create the VARIANT returned:
OnNextRestart_Config_CA_AIA_Include_In_Cert
OnNextRestart_Config_CA_CACert_Publish_To
The vt member of the VARIANT MUST be set to VT_ARRAY | VT_BSTR and the pArray member MUST reference a single dimension safearray.
The number of elements of the safearray referenced by pArray MUST be equal to the number of URLs. For each URL, there MUST be an element in the safearray referenced by pArray containing the BSTR for the Unicode string value of the URI.
Each URI is of the format "NumericPrefix:URI", where NumericPrefix is the decimal value corresponding to the combination of following flags:
0x00000001 – The CA must publish the CAcertificate (1)(s) to the URI (OnNextRestart_Config_CA_CACert_Publish_To).
0x00000002 – The URI is to be added in the AIA extension of the certificates (1) issued by the CA (OnNextRestart_Config_CA_AIA_Include_In_Cert).</td></tr>
<tr><td>pwszNodePath is EMPTY and pwszEntry is "OCSPURLs"</td><td>The CA MUST return the value of the OnNextRestart_Config_CA_OCSP_Include_In_Cert ADM element as a VARIANT.
The vt member of the VARIANT MUST be set to VT_ARRAY | VT_BSTR and the pArray member MUST reference a single dimension safearray.</td></tr>
</table>
</td></tr>
</table>

Errata Published*	Description

	The number of elements of the safearray referenced by pArray MUST be equal to the number of machines running Online Responder Service with the same configuration information. For each machine, there MUST be an element in the safearray referenced by pArray containing the BSTR for the Unicode string value of the URI of the machine.
pwszNodePath is "PolicyModules\CertificateAuthority_MicrosoftDefault.Policy" and pwszEntry is "RequestDisposition"	The CA MUST return the value of the OnNextRestart_Config_CA_Requests_Disposition as a VARIANT. The vt member of VARIANT MUST be set to VT_I4 and the lVal member MUST be the value of the OnNextRestart_Config_CA_Requests_Disposition ADM element. The value of this ADM element determines whether the CA sets all requests to pending, accepts all requests, or denies all requests.

Changed to:

Input Parameters	Processing rule for pVariant
pwszNodePath is EMPTY and pwszEntry is "CACertPublicationURLs"	The CA MUST use the values of the following ADM elements to create the VARIANT returned: OnNextRestart_Config_CA_AIA_Include_In_Cert OnNextRestart_Config_CA_CACert_Publish_To The vt member of the VARIANT MUST be set to VT_ARRAY | VT_BSTR and the pArray member MUST reference a single dimension safearray. The number of elements of the safearray referenced by pArray MUST be equal to the number of URLs. For each URL, there MUST be an element in the safearray referenced by pArray containing the BSTR for the Unicode string value of the URI. Each URI is of the format "NumericPrefix:URI", where NumericPrefix is the decimal value corresponding to the combination of following flags: 0x00000001 – The CA must publish the CAcertificate (1)(s) to the URI (OnNextRestart_Config_CA_CACert_Publish_To). 0x00000002 – The URI is to be added in the AIA extension of the certificates (1)

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><td></td><td>issued by the CA (OnNextRestart_Config_CA_AIA_Include_In_Cert).</td></tr>
<tr><td>pwszNodePath is "PolicyModules\CertificateAuthority_MicrosoftDefault.Policy" and pwszEntry is "RequestDisposition"</td><td>The CA MUST return the value of the OnNextRestart_Config_CA_Requests_Disposition as a VARIANT. The vt member of VARIANT MUST be set to VT_I4 and the lVal member MUST be the value of the OnNextRestart_Config_CA_Requests_Disposition ADM element. The value of this ADM element determines whether the CA sets all requests to pending, accepts all requests, or denies all requests.</td></tr>
</table>
In Section 3.1.4.2.15 ICertAdminD2::SetConfigEntry (Opnum 45), removed the row containing 'pwszNodePath is EMPTY and pwszEntry is OCSPURLs' in two tables, as OCSPURLs is an invalid entry.
Changed from:
<table>
<tr><th>Input</th><th>Store information as ADM element</th></tr>
<tr><td>pwszNodePath is EMPTY and pwszEntry is "CACertPublicationURLs"</td><td>OnNextRestart_Config_CA_AIA_Include_In_Cert
OnNextRestart_Config_CA_CACert_Publish_To</td></tr>
<tr><td>pwszNodePath is EMPTY and pwszEntry is "OCSPURLs"</td><td>OnNextRestart_Config_CA_OCSP_Include_In_Cert</td></tr>
<tr><td>pwszNodePath is EMPTY and pwszEntry is "CRLAttemptRepublish"</td><td>OnNextRestart_Config_CA_CRL_Attempt_Republish</td></tr>
</table>
Changed to:
<table>
<tr><th>Input</th><th>Store information as ADM element</th></tr>
<tr><td>pwszNodePath is EMPTY and pwszEntry is "CACertPublicationURLs"</td><td>OnNextRestart_Config_CA_AIA_Include_In_Cert
OnNextRestart_Config_CA_CACert_Publish_To</td></tr>
<tr><td>pwszNodePath is EMPTY and pwszEntry is "CRLAttemptRepublish"</td><td>OnNextRestart_Config_CA_CRL_Attempt_Republish</td></tr>
</table>
</td></tr>
<tr><td>2020/07/20</td><td>In Section 2.2.1.11, added the optional ACCESS_DENIED_CALLBACK_ACE type for all access control entries (ACEs) in the discretionary access control list (DACL), as a security descriptor property for Officer rights and Enrollment Agent rights.

Changed from:

"1. AceType 0x9 (ACCESS_ALLOWED_CALLBACK_ACE_TYPE for the ACCESS_ALLOWED_CALLBACK_ACE, [MS-DTYP] section 2.4.4.6)."</td></tr>
</table>

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>Changed to:"1. Either the AceType 0x9 (ACCESS_ALLOWED_CALLBACK_ACE_TYPE for the ACCESS_ALLOWED_CALLBACK_ACE, [MS-DTYP] section 2.4.4.6), or the AceType 0x0A (ACCESS_DENIED_CALLBACK_ACE_TYPE for the ACCESS_DENIED_CALLBACK_ACE, [MS-DTYP] section 2.4.4.7)."</td></tr>
<tr><td>2019/03/18</td><td>In this document, changed the default value of the CA for Windows Server 2019.

In Section 3.1.4.2.14, ICertAdminD2::GetConfigEntry (Opnum 44), changed from:

…
8. For each input in the left column of the table below, the CA MUST perform the processing rules in the corresponding cell in the right column.

<table>
<tr><th>Input Parameters</th><th>Processing rule for pVariant</th></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td>pwszNodePath is EMPTY and pwszEntry is "Version"</td><td>The CA MUST return the value of the OnNextRestart_Config_Product_Version ADM element as a VARIANT.
The vt member of the VARIANT MUST be set to VT_I4 and the lVal member MUST be set to the one of the following values:
0x00010001 – Server is Windows 2000 Server operating system
0x00020002 – Server is Windows Server 2003 operating system
0x00030001 – Server is Windows Server 2008 operating system
0x00040001 – Server is Windows Server 2008 R2 operating system
0x00050001 – Server is Windows Server 2012 operating system
0x00060001 – Server is Windows Server 2012 R2 operating system <72>
0x00070001 – Server is Windows Server 2016 operating system
0x00080001 – Server is Windows Server 2019 operating system</td></tr>
<tr><td>…</td><td>…</td></tr>
</table>

Changed to:

…
8. For each input in the left column of the table below, the CA MUST perform the processing rules in the corresponding cell in the right column.</td></tr>
</table>

Errata Published*	Description

Input Parameters	Processing rule for pVariant
…	…
pwszNodePath is EMPTY and pwszEntry is "Version"	The CA MUST return the value of the OnNextRestart_Config_Product_Version ADM element as a VARIANT. The vt member of the VARIANT MUST be set to VT_I4 and the lVal member MUST be set to the one of the following values: 0x00010001 – Server is Windows 2000 Server operating system 0x00020002 – Server is Windows Server 2003 operating system 0x00030001 – Server is Windows Server 2008 operating system 0x00040001 – Server is Windows Server 2008 R2 operating system 0x00050001 – Server is Windows Server 2012 operating system 0x00060001 – Server is Windows Server 2012 R2 operating system <72> 0x00070001 – Server is Windows Server 2016 operating system or Windows Server 2019 operating system
…	…

In Section 7, Appendix B: Product Behavior, changed from:

<11> Section 3.1.1.10: Microsoft CAs persist only a subset of the configuration data. They store the configuration data in the registry in the following locations:

…

Version

ADM Datum: Config_Product_Version and OnNextRestart_Config_Product_Version

Registry Value Type: REG_DWORD

Default Value: By default, the value depends on the Windows version:

…

0x00080001: Windows Server 2019

Changed to:

<11> Section 3.1.1.10: Microsoft CAs persist only a subset of the configuration data. They store the configuration data in the registry in the following locations:

…

Version

ADM Datum: Config_Product_Version and OnNextRestart_Config_Product_Version

Errata Published*	Description
	Registry Value Type: REG_DWORD Default Value: By default, the value depends on the Windows version: … 0x00070001: Windows Server 2016 or Windows Server 2019

*Date format: YYYY/MM/DD

[MS-CSSP]: Credential Security Support Provider (CredSSP) Protocol

This topic lists the Errata found in the MS-CSSP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

June 1, 2017 - Download

March 16, 2018 - Download

September 12, 2018 - Download

Errata below are for Protocol Document Version V17.0 – 2018/09/12.

Errata Published*	Description
2020/07/06	In 9 sections - Section 2, Message Syntax, through Section 2.2.1.2.3.1, TSRemoteGuardPackageCred - added behavior notes to indicate character encoding and each data type with ASN.1 data type in each field as unsigned integer encoded as ASN.1 INTEGER or ASN.1 OCTET STRING. Added the following Product Note in or after each introduction: <n> Where data is a text string, Windows uses a Unicode string defined by a UNICODE_STRING structure to encode to ASN.1 OCTET STRING format. For more information see [MSDOCS-UNICODE_STRING]. For a description of Octet String see [MS-DTYP] and [X690].
2020/07/06	In Section 2.2.1,TSRequest, added a product behavior note that TLS requires messages only be fragmented at TLS's maximum message length. Changed from: . . .The TSRequest message, section 2.2.1, is always sent over the TLS-encrypted channel between the client and server in a CredSSP Protocol exchange (see step 1 in section 3.1.5). Changed to: . . .The TSRequest message, section 2.2.1, is always sent over the TLS-encrypted channel between the client and server in a CredSSP Protocol exchange (see step 1 in section 3.1.5).<8> <8> Section 2.2.1: The CredSSP standard requires that a TLS encrypted message fragment contain an entire ASN.1 message. CredSSP expects that the entire first tag and length to fall in the initial block of decrypted data and for the client to encrypt TSRequest messages as single blocks subject only to fragmentation at TLS's maximum message length. The CredSSP server expects a TLS encryption of an entire TSRequest message without fragmentation. Otherwise, the server returns an error.
2020/07/06	In Section 3.1.5, Processing Events and Sequencing Rules, added to step 1 that TLS session resumption is not supported.

Errata Published*	Description
	Changed from: 1. The CredSSP client and CredSSP server first complete the TLS handshake, as specified in [RFC2246]. After the handshake is complete, all subsequent CredSSP Protocol messages are encrypted by the TLS channel. The CredSSP Protocol does not extend the TLS wire protocol. As part of the TLS handshake, the CredSSP server does not request the client's X.509 certificate (thus far, the client is anonymous). Changed to: 1. The CredSSP client and CredSSP server first complete the TLS handshake, as specified in [RFC2246]. After the handshake is complete, all subsequent CredSSP Protocol messages are encrypted by the TLS channel. The CredSSP Protocol does not extend the TLS wire protocol. TLS session resumption is not supported. As part of the TLS handshake, the CredSSP server does not request the client's X.509 certificate (thus far, the client is anonymous).
2020/06/08	In Section 2.2.1.2.3.1, TSRemoteGuardPackageCred, clarified data structures and processing in product note 12. Changed from: In Windows, logon credentials (in the logonCred field of TSRemoteGuardCreds) are required in the KERB_TICKET_LOGON structure where the KRB_CRED message ([RFC4120], section 5.8.1) in the TicketGrantingTicket member is using the KERB_RPC_ENCRYPTION_KEY ([MS-RDPEAR] section 2.2.1.2.1) for the EncryptionKey. Supplemental credentials (in the supplementalCreds field of TSRemoteGuardCreds) are required in the following structure: Changed to: In Windows, the logon credentials that are in the logonCred field of TSRemoteGuardCreds structure are required to be in a KERB_TICKET_LOGON structure ([KERB-TICKET-LOGON]). The TicketGrantingTicket member within the KERB_TICKET_LOGON structure is an ASN.1-encoded KRB_CRED message ([RFC4120], section 5.8.1). The EncryptionKey in KrbCredInfo ([RFC4120], section 5.8.1) is required to be in a KERB_RPC_ENCRYPTION_KEY structure ([MS-RDPEAR] section 2.2.1.2.1). The ServiceTicket member within the KERB_TICKET_LOGON structure is a ticket to the computer account. Windows CredSSP clients will use Kerberos User to User tickets ([RFC4120], section 2.9.2) as the ServiceTicket, but the server does not enforce this. The session key of the ServiceTicket is used to encrypt the EncryptedData in the KRB_CRED message.

*Date format: YYYY/MM/DD

[MS-CSVP]: Failover Cluster: Setup and Validation Protocol (ClusPrep)

This topic lists the Errata found in the MS-CSVP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

Errata below are for Protocol Document Version V27.0 – 2019/03/13.

Errata Published*	Description

2019/08/05	In Section 3.2.4.4, CprepPrepareNodePhase2 (Opnum 6), more detail has been added to clarify server processing steps for this call.
	
	Changed from:
	...
	
	When processing this call, the server MUST do the following:
	• Determine the number of disks accessible to the system in an implementation-specific way.
	• For each disk:
	  • Create a ClusPrepDisk object.
	  • Initialize ClusPrepDisk.AttachedState to Not Attached.
	  • Initialize ClusPrepDisk.OnlineState to Not Online.
	  • Initialize ClusPrepDisk.OwnedState to Not Owned.
	  • Add the disk to ClusPrepDiskList.
	• Set pulNumDisks to that number.
	• Set the server Prepare State to Online.
	The server returns the following information to the client:
	• The number of disks attached to the system
	
	Changed to:
	...
	
	When processing this call, the server MUST do the following:
	• Determine the number of disks accessible to the system in an implementation-specific way.
	• If the Flags field includes ForceOfflineNonClusteredDisks but does not include SkipNonClusteredPools, detach spaces using non-clustered pools before including them in disks eligible for validation.
	• If the Flags field includes SkipNonClusteredPools, skip non-clustered pools for validation.
	• If the Flags field includes neither ForceOfflineNonClusteredDisks nor SkipNonClusteredPools, skip non-clustered pools with attached spaces for validation.

Errata Published*	Description

	• For each disk:
	  • Create a ClusPrepDisk object.
	  • Initialize ClusPrepDisk.AttachedState to Not Attached.
	  • Initialize ClusPrepDisk.OnlineState to Not Online.
	  • Initialize ClusPrepDisk.OwnedState to Not Owned.
	  • Add the disk to ClusPrepDiskList.
	• Set pulNumDisks to the number of disks in ClusPrepDiskList.
	• Set the server Prepare State to Online.
	The server returns the following information to the client:
	• pulNumDisks

*Date format: YYYY/MM/DD

[MS-DCOM]: Distributed Component Object Model (DCOM) Remote Protocol

This topic lists the Errata found in the MS-DCOM document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-DFSC]: Distributed File System (DFS) Referral Protocol

This topic lists the Errata found in [MS-DFSC] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

September 12, 2018 - Download

Errata below are for Protocol Document Version V28.0 – 2018/09/12.

Errata Published*	Description
2018/10/29	In Section 3.1.4.2, Sending a DFS Referral Request to the Server, the following has been changed from: The client MUST query the DFS referral, as specified in [MS-CIFS] section 3.4.4.9, by passing ClientGenericContext, HostName, UserCredentials, MaxOutputSize, the REQ_GET_DFS_REFERRAL_EX or REQ_GET_DFS_REFERRAL structure as the input buffer, and the FSCTL code set to FSCTL_DFS_GET_REFERRALS or FSCTL_DFS_GET_REFERRALS_EX based on the input buffer. Changed to: The client MUST query the DFS referral, as specified in [MS-CIFS] section 3.4.4.9, by passing ClientGenericContext, HostName, UserCredentials, MaxOutputSize, the REQ_GET_DFS_REFERRAL_EX or REQ_GET_DFS_REFERRAL structure as the input buffer, and the FSCTL code set to FSCTL_DFS_GET_REFERRALS, if the input buffer is an REQ_GET_DFS_REFERRAL, or FSCTL_DFS_GET_REFERRALS_EX, if the input buffer is an REQ_GET_DFS_REFERRAL_EX.

*Date format: YYYY/MM/DD

[MS-DHCPE]: Dynamic Host Configuration Protocol (DHCP) Extensions

This topic lists the Errata found in [MS-DHCPE] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

Errata below are for Protocol Document Version V24.0 – 2018/09/12.

Errata Published*	Description
2019/12/16	In Section 3.1.5.2, Receiving a DHCPACK, added alternate processing for none or two route options. Changed from: If it contains a Microsoft Classless Static Route Option, the client MUST first check whether the option conforms to the syntax specified in section 2.2.8. If any of the parameters in this DHCPv4 option are invalid or incomplete, the DHCPv4 client MUST silently discard the complete DHCPv4 message and start the initialization process again. Otherwise, the specified routes MUST be inserted into the routing table in the TCP/IP stack. Changed to: If it contains a Microsoft Classless Static Route Option, the client MUST first check whether the option conforms to the syntax specified in section 2.2.8. If any of the parameters in this DHCPv4 option are invalid or incomplete, the DHCPv4 client MUST silently discard the complete DHCPv4 message and start the initialization process again. Otherwise, if the DHCPACK does not contain a Classless Static Route Option (121), the specified routes MUST be inserted into the routing table in the TCP/IP stack. If it contains both a Microsoft Classless Static Route Option (249) and a Classless Static Route Option (121) then the client MUST select either (in any implementation-specific way[27]) set of routes as the routes to be added into the routing table in the TCP/IP stack. <27> Section 3.1.5.2: All versions of Windows Vista and Windows Server 2008 and later will insert the last option in the message.

*Date format: YYYY/MM/DD

[MS-DHCPM]: Microsoft Dynamic Host Configuration Protocol (DHCP) Server Management Protocol

This topic lists the Errata found in [MS-DHCPM] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

September 15, 2017 - Download

[MS-DNSP]: Domain Name Service (DNS) Server Management Protocol

This topic lists the Errata found in the MS-DNSP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

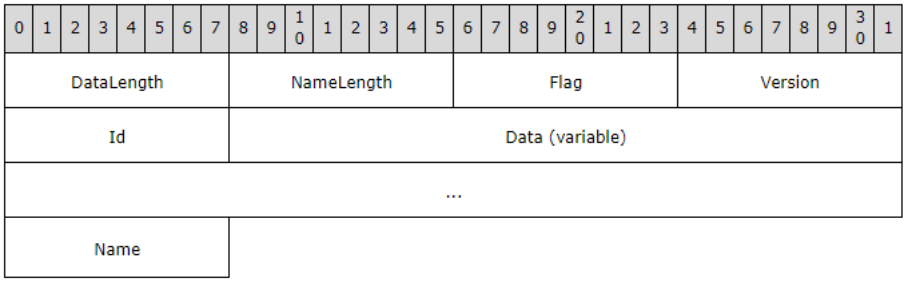
June 30, 2015 - Download

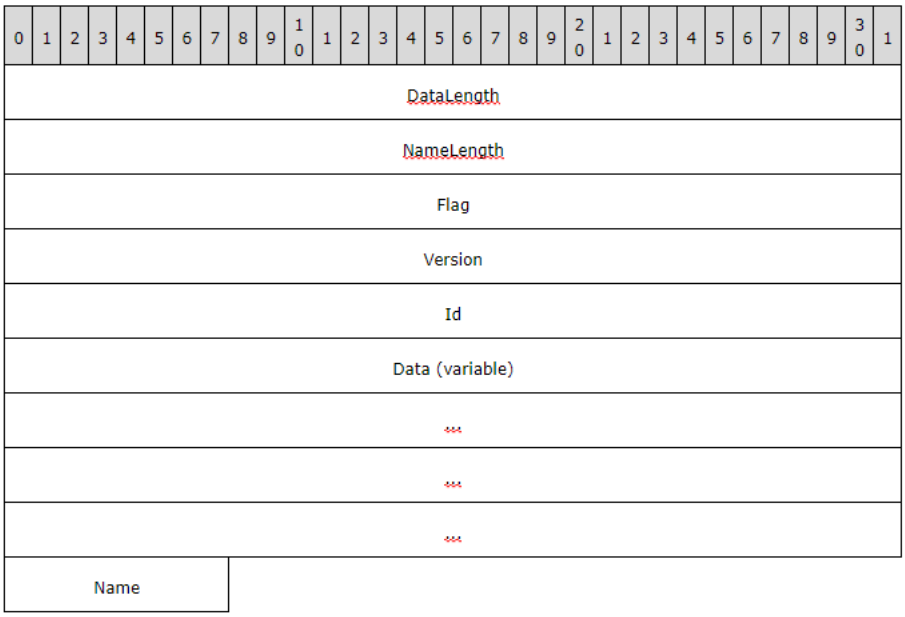
July 18, 2016 - Download

June 1, 2017 - Download

September 12, 2018 - Download

Errata below are for Protocol Document Version V35.0 – 2019/03/13.

Errata Published*	Description
2020/07/20	In Section 2.3.2.1, dnsProperty, updated bit table and field sizes. Changed from:  DataLength (1 byte): An unsigned binary integer containing the length, in bytes, of the Data field. If this value is 0, default values are assigned to Data. See Property Id section 2.3.2.1.1. NameLength (1 byte): Not Used. The value MUST be ignored and assumed to be 0x00000001. Flag (1 byte): This field is reserved for future use. The value MUST be 0x00000000. Version (1 byte): The version number associated with the property attribute. The value MUST be 0x00000001. Id (1 byte): The property attribute's type. See Property Id (section 2.3.2.1.1). Data (variable): The data associated with an Id. See Property Id (section 2.3.2.1.1). Name (1 byte): Not used. The value MUST be of length 1 byte, and MUST be ignored Changed to:

Errata Published*	Description
	 DataLength (4 bytes): An unsigned binary integer containing the length, in bytes, of the Data field. If this value is 0, default values are assigned to Data. See Property Id section 2.3.2.1.1. NameLength (4 bytes): Not Used. The value MUST be ignored and assumed to be 0x00000001. Flag (4 bytes): This field is reserved for future use. The value MUST be 0x00000000. Version (4 bytes): The version number associated with the property attribute. The value MUST be 0x00000001. Id (4 bytes): The property attribute's type. See Property Id (section 2.3.2.1.1). Data (variable): The data associated with an Id. See Property Id (section 2.3.2.1.1).Name (1 byte): Not used. The value MUST be of length 1 byte, and MUST be ignored.
2020/07/06	In Section 1.5, Prerequisites/Preconditions, added a product behavior note about DNS data validation. Changed from: …Consistency of DNS data stored in the local directory server is not guaranteed, since complete or partial updates to the LDAP directory can be replicated to the local directory server at any time…. Changed to: …Consistency of DNS data stored in the local directory server is not guaranteed, since complete or partial updates to the LDAP directory can be replicated to the local directory server at any time.<1> … <1>Active Directory does not perform extensive validation on the DNS data written to the directory, it is the responsibility of the entity that consumes the directory data to validate in. In case of Windows DNS, the data read from the directory is validated. In Section 2.3.2.1, dnsProperty, added reference to Property Id table for default values if the DataLength field is 0. Changed from:

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
DataLength (1 byte): An unsigned binary integer containing the length, in bytes, of the Data field.

Changed to:
DataLength (1 byte): An unsigned binary integer containing the length, in bytes, of the Data field. If this value is 0, default values are assigned to Data. See Property Id (section 2.3.2.1.1).

In Section 2.3.2.1.1, Property Id, added to introduction the case for when DataLength field is zero default values are used. Added default values in the ID table.

Changed from:
The Id specifies the type of data in a dnsProperty's Data field.
<table>
<tr><th>Constant/value</th><th>Description</th></tr>
<tr><td>DSPROPERTY_ZONE_TYPE
0x00000001</td><td>The zone type. See dwZoneType (section 2.2.5.2.4.1). Default: DNS_ZONE_TYPE_PRIMARY</td></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td>DSPROPERTY_ZONE_SECURE_TIME
0x00000008</td><td>The time at which the zone became secure. …</td></tr>
<tr><td>DSPROPERTY_ZONE_NOREFRESH_INTERVAL
0x00000010</td><td>The zone no refresh interval. …</td></tr>
<tr><td>DSPROPERTY_ZONE_REFRESH_INTERVAL
0x00000020</td><td>The zone refresh interval. …</td></tr>
<tr><td>DSPROPERTY_ZONE_AGING_STATE
0x00000040</td><td>Whether aging is enabled. …</td></tr>
<tr><td>DSPROPERTY_ZONE_SCAVENGING_SERVERS
0x00000011</td><td>A list of DNS servers that will perform scavenging. …</td></tr>
<tr><td>DSPROPERTY_ZONE_AGING_ENABLED_TIME
0x00000012</td><td>The time interval before the next scavenging cycle. …</td></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td>DSPROPERTY_ZONE_MASTER_SERVERS
0x00000081</td><td>A list of DNS servers that will perform zone transfers. …</td></tr>
<tr><td>DSPROPERTY_ZONE_AUTO_NS_SERVERS
0x00000082</td><td>A list of servers which MAY autocreate a delegation. …</td></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td>DSPROPERTY_ZONE_SCAVENGING_SERVERS_DA
0x00000090</td><td>A list of DNS servers that will perform scavenging. ...</td></tr>
<tr><td>DSPROPERTY_ZONE_MASTER_SERVERS_DA
0x00000091</td><td>A list of DNS servers that will perform zone transfers. ...</td></tr>
<tr><td>DSPROPERTY_ZONE_AUTO_NS_SERVERS_DA
0x00000092</td><td>A list of servers which MAY autocreate a delegation. ...</td></tr>
</table>
</td></tr>
</table>

Errata Published*	Description

…	…

Changed to:

The Id specifies the type of data in a dnsPropertyData field.<100> Each property Id in the table has a default value that is assigned if DataLength field of the dnsProperty is 0, irrespective of what is in the Data (variable) field.

Constant/value	Description
DSPROPERTY_ZONE_TYPE 0x00000001	The zone type. See dwZoneType (section 2.2.5.2.4.1). Default: DNS_ZONE_TYPE_PRIMARY
…	…
DSPROPERTY_ZONE_SECURE_TIME 0x00000008	The time at which the zone became secure. … Default: 0.
DSPROPERTY_ZONE_NOREFRESH_INTERVAL 0x00000010	The zone no refresh interval. … Default: 168 hours/7 days.
DSPROPERTY_ZONE_REFRESH_INTERVAL 0x00000020	The zone refresh interval. … Default: 168 hours/7 days.
DSPROPERTY_ZONE_AGING_STATE 0x00000040	Whether aging is enabled. … Default: 0.
DSPROPERTY_ZONE_SCAVENGING_SERVERS 0x00000011	A list of DNS servers that will perform scavenging. … Default: Empty Array.
DSPROPERTY_ZONE_AGING_ENABLED_TIME 0x00000012	The time interval before the next scavenging cycle. … Default: 0.
…	…
DSPROPERTY_ZONE_MASTER_SERVERS 0x00000081	A list of DNS servers that will perform zone transfers. … Default: Empty Array.
DSPROPERTY_ZONE_AUTO_NS_SERVERS 0x00000082	A list of servers which MAY autocreate a delegation. … Default: Empty Array.
…	…
DSPROPERTY_ZONE_SCAVENGING_SERVERS_DA 0x00000090	A list of DNS servers that will perform scavenging. … Default: Empty Array.
DSPROPERTY_ZONE_MASTER_SERVERS_DA 0x00000091	A list of DNS servers that will perform zone transfers. … Default: Empty Array.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><td>DSPROPERTY_ZONE_AUTO_NS_SERVERS_DA
0x00000092</td><td>A list of servers which MAY autocreate a delegation. ...
Default: Empty Array.</td></tr>
<tr><td>…</td><td>…</td></tr>
</table>
</td></tr>
<tr><td>2019/04/29</td><td>In Section 2.2.15.1.1.6, DNS_RPC_CRITERIA_ENUM, a product behavior note has been updated to indicate a change in the product versions that support the DnsPolicyCriteriaEDNSSubnet policy.

Changed from:

DnsPolicyCriteriaEDNSSubnet: Usage of this enum constant will fail the request with Win32 Error-9974 (DNS_ERROR_POLICY_INVALID_SETTINGS).<90>
<90> Section 2.2.15.1.1.6: DnsPolicyCriteriaEDNSSubnet is not implemented in Windows Server v1809 operating system and earlier.

Changed to:

DnsPolicyCriteriaEDNSSubnet: Usage of this enum constant will fail the request with Win32 Error-9974 (DNS_ERROR_POLICY_INVALID_SETTINGS).<90>
<90> Section 2.2.15.1.1.6: DnsPolicyCriteriaEDNSSubnet is implemented in Windows Server v1809 operating system with [MSKB-4497934] and later.

Also, in Section 1.2.1, Normative References, the following reference has been added:

[MSKB-4497934] Microsoft Corporation, "May 20, 2019 - KB4497934", https://support.microsoft.com/en-us/help/4497934</td></tr>
</table>

*Date format: YYYY/MM/DD

[MS-DPWSSN]: Devices Profile for Web Services (DPWS) Size Negotiation Extension

This topic lists the Errata found in [MS-DPWSSN] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

June 1, 2017 - Download

[MS-DRSR]: Directory Replication Service (DRS) Remote Protocol

This topic lists the Errata found in the MS-DRSR document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

September 15, 2017 - Download

December 1, 2017 - Download

Errata below are for Protocol Document Version V40.0 – 2019/09/12.

Errata Published*	Description

2019/10/16	In Section 4.1.1.2.3, CreateNtdsData, the pseudocode for creating an nTDSDSA object has been updated by including a check if attributes meet correct order for creating the NtdsDsa object, and if not, setting the ERROR_DS_NO_CROSSREF_FOR_NC error and returning 'false'.
	
	Changed from:
	
	.. if not accessAllowed then
	SetErrorData(SV_PROBLEM_DIR_ERROR, serviceError,
	ERROR_ACCESS_DENIED, pmsgOut, ver)
	return false
	endif
	
	/* Check for the functional level compliance. The functional level.."
	
	Changed to:
	.. if not accessAllowed then
	SetErrorData(SV_PROBLEM_DIR_ERROR, serviceError,
	ERROR_ACCESS_DENIED, pmsgOut, ver)
	return false
	endif
	
	correctOrder := DoAttributesSatisfyPreCheckForCreateNtdsDsa (entList)
	
	if not correctOrder then

Errata Published*	Description
	SetErrorData(SV_PROBLEM_DIR_ERROR, serviceError, ERROR_DS_NO_CROSSREF_FOR_NC, pmsgOut, ver) return false endif /* Check for the functional level compliance. The functional level.." Also, in Section 4.1.1.2.11, DoAttributesSatisfyPreCheckForCreateNtdsDsa, new content has been added to describe the new procedure above added in section 4.1.1.2.3.

*Date format: YYYY/MM/DD

[MS-DTCO]: MSDTC Connection Manager: OleTx Transaction Protocol

This topic lists the Errata found in the MS-DTCO document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

December 1, 2017 - Download

[MS-DSCPM]: Desired State Configuration Pull Model Protocol

This topic lists the Errata found in the MS-DSCPM document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

December 1, 2017 - Download

September 12, 2018 - Download

[MS-DTYP]: Windows Data Types

This topic lists the Errata found in the MS-DTYP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

September 15, 2017 - Download

December 1, 2017 - Download

September 12, 2018 - Download

Errata below are for Protocol Document Version V35.0 – 2018/09/12

Errata Published*	Description
2020/03/02	In Section 2.5.3.2, Access Check Algorithms Pseudocode, the pseudocode confirming that the object owner is always granted READ_CONTROL and WRITE_DAC has been corrected as follows: Changed from: Set GrantedAccess to GrantedAccess or READ_CONTROL or WRITE_OWNER Changed to: Set GrantedAccess to GrantedAccess or READ_CONTROL or WRITE_DAC
2019/11/11	In Section 2.4.2.4, Well-Known SID Structures, the description of the table entry for AUTHENTICATED_USERS has been updated for clarity, and an associated behavior note added: Changed from: A group that includes all users whose identities were authenticated when they logged on. Changed to: A group that includes all users whose identities were authenticated when they logged on. Users authenticated as Guest or Anonymous are not members of this group.<11>

Errata Published*	Description
	<11> Windows server versions earlier than Windows Server 2003 and client versions earlier than Windows XP SP2 included the Guest account in the Authenticated Users group.

*Date format: YYYY/MM/DD

[MS-DVRD]: Device Registration Discovery Protocol

This topic lists the Errata found in [MS-DVRD] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-DVRE]: Device Registration Enrollment Protocol

This topic lists the Errata found in the MS-DVRE document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-DVRJ]: Device Registration Join Protocol

This topic lists the Errata found in the MS-DVRJ document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-ECS]: Enterprise Client Synchronization Protocol

This topic lists the Errata found in the MS-ECS document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

Errata below are for Protocol Document Version V9.0 – 2018/09/12.

Errata Published*	Description
2020/03/02	In Section 3.6.5.1, Upload Scenario, the following was changed from: For each UploadFile in UploadFileList, the client MUST update UploadFile.CommitStatus to the Status entry returned in the FILE_STATUS structure. Changed to: For each UploadFile in UploadFileList, the client MUST update UploadFile.CommitStatus to the Status entry returned in the FILE_STATUS_ENTRY structure.
2019/09/30	In this document, numerous editorial fixes have been made, e.g., changed instances of "ID" to "Id" or instances of "Id" to "ID"; changed instances of "FileMetaDataTable" to "FileMetadataTable"; and removed whitespaces. Sections updated: 2.2.1 2.2.1.5 2.2.2.5 2.2.2.17 2.2.2.18 3.2.5.1.1 3.2.5.1.1.3 3.3.5.1.1 3.3.5.1.1.2 3.3.5.1.1.3 3.4.5.1.1 3.4.5.1.1.1 3.4.5.1.1.3 3.4.5.2.1

Errata Published*	Description
	3.4.5.2.2 3.4.5.2.2.3 3.4.5.3.1 3.4.5.3.1.3 3.4.5.4.1 3.4.5.4.1.3 3.4.5.5.1 3.4.5.6.1 3.5.5.1.1 3.5.5.2.1 3.6.1.1 3.6.3 3.6.5.1 3.6.5.2 4.1 For details on the above changes, see the PDF doc here.
2019/09/16	In Section 3.4.5.3.1.3, Processing Details, the following was changed from: Otherwise, if FileMetadata.RemoteStreamId is not equal to StreamId, and FileSize is greater than the space available for a user, the server MUST set ProtocolType to 0x00 and MUST set PrepareResult to ERROR_DISK_FULL, as specified in [MS-ERREF] section 2.1.1. Changed to: Otherwise, if FileMetadata.FileStreamId is not equal to FILE_INFO_INPUT_ENTRY.StreamId, and FILE_INFO_INPUT_ENTRY.FileSize is greater than the space available for a user, the server MUST set ProtocolType to 0x00 and MUST set PrepareResult to ERROR_DISK_FULL, as specified in [MS-ERREF] section 2.1.1.

*Date format: YYYY/MM/DD

[MS-EFSR]: Encrypting File System Remote (EFSRPC) Protocol

This topic lists the Errata found in the MS-EFSR document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-EMF]: Enhanced Metafile Format

This topic lists the Errata found in the MS-EMF document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

June 1, 2017 - Download

[MS-EMFPLUS]: Enhanced Metafile Format Plus Extensions

This topic lists the Errata found in the MS-EMFPLUS document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

March 4, 2020 - Download

Errata below are for Protocol Document Version V17.0 – 2020/03/04.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td>2020/07/06</td><td>In Section 2.2.3.4, ColorCurveEffect Object, added midtone adjustment range to AdjustmentIntensity field.

Changed from:

Shadow adjustment range:
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>-100 ≤ value < 0</td><td>As the value decreases, the dark areas of the image SHOULD appear darker.</td></tr>
<tr><td>0</td><td>A value of 0 specifies that the shadow MUST NOT change.</td></tr>
<tr><td>0 < value ≤ 100</td><td>As the value increases, the dark areas of the image SHOULD appear lighter.</td></tr>
</table>
Changed to:

Shadow adjustment range:
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>-100 ≤ value < 0</td><td>As the value decreases, the dark areas of the image SHOULD appear darker.</td></tr>
<tr><td>0</td><td>A value of 0 specifies that the shadow MUST NOT change.</td></tr>
<tr><td>0 < value ≤ 100</td><td>As the value increases, the dark areas of the image SHOULD appear lighter.</td></tr>
</table>
Midtone adjustment range:</td></tr>
</table>

Errata Published*	Description

Value	Meaning
- 100 ≤ value < 0	As the value decreases, the midtones of the image SHOULD appear darker.
0	A value of 0 specifies that the midtone MUST NOT change.
0 < value ≤ 100	As the value increases, the midtones of the image SHOULD appear lighter.

*Date format: YYYY/MM/DD

[MS-ERREF]: Windows Error Codes

This topic lists the Errata found in the MS-ERREF document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

June 1, 2017 - Download

September 15, 2017 - Download

Errata below are for Protocol Document Version V19.0 – 2018/09/12.

Errata Published*	Description
2019/08/05	In the Section 1.1, Glossary, the entry for the term message identifier, which is at odds with the definition in Section 2.2, has been removed.

*Date format: YYYY/MM/DD

[MS-EVEN]: EventLog Remoting Protocol

This topic lists the Errata found in the MS-EVEN document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

Errata below are for Protocol Document Version V22.0 – 2018/09/12.

Errata Published*	Description
2019/09/02	In Section 2.2.6, Handles, the section name has been changed to reflect the name of the type it describes. Changed from: 2.2.6 Handles Changed to: 2.2.6 IELF_HANDLE In Section 3.1.4.7, ElfrReadELW (Opnum 10), the name of the EVENTLOG_BACKWARDS_READ flag contained a misspelling in one place. Changed from: … If neither of the two flags are set, the server will treat it as if the EVENTLOG_BACKWORDS_READ flag is set. … Changed to: … If neither of the two flags are set, the server will treat it as if the EVENTLOG_BACKWARDS_READ flag is set. …

*Date format: YYYY/MM/DD

[MS-EVEN6]: EventLog Remoting Protocol Version 6.0

This topic lists the Errata found in the MS-EVEN6 document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 12, 2018 - Download

[MS-FASP]: Firewall and Advanced Security Protocol

This topic lists the Errata found in the MS-FASP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

March 13, 2019 - Download

[MS-FAX]: Fax Server and Client Remote Protocol

This topic lists the Errata found in the MS-FAX document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 12, 2018 - Download

[MS-FRS2]: Distributed File System Replication Protocol

This topic lists the Errata found in the MS-FRS2 document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

Errata below are for Protocol Document Version V28.0 – 2018/09/12.

Errata Published*	Description
2019/02/19	In Section 1.2.1, Normative References, the following reference has been added: [MS-XCA] Microsoft Corporation, "Xpress Compression Algorithm". In Section 2.2.1.4.15, XPRESS Block, the Block Data field has been changed from: If the value of the Block Compressed Size field is less than the value of the Block Uncompressed Size field, then the data has been compressed. For more information about decompressing compressed data, see section 3.1.1.1.3.9. Changed to: If the value of the Block Compressed Size field is less than the value of the Block Uncompressed Size field, then the data has been compressed. For more information about decompressing compressed data, see section 3.1.1.2. In Section 3.1.1.1, Compression, the following was changed from: Many of the FrsTransport methods use compression to reduce the amount of data that is returned to the client. This section describes algorithms and a conceptual model of possible data organization that an implementation maintains in order to decompress compressed data. The described organization is provided to facilitate the explanation of how the algorithm behaves. Error checking and handling has been omitted from all algorithms in the interests of clarity. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with what is described in this document. Changed to: Many of the FrsTransport methods use the LZ77+Huffman Compression algorithm, specified in [MS-XCA] section 2.1, to compress data. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with what is described in this document.

Errata Published*	Description
	The following sections have been removed and replaced with links to MS-XCA: 3.1.1.1.1 Pseudocode Conventions 3.1.1.1.2 Data Structures 3.1.1.1.2.1 PREFIX_CODE_NODE 3.1.1.1.2.2 PREFIX_CODE_SYMBOL 3.1.1.1.2.3 BITSTRING 3.1.1.1.3 Procedures 3.1.1.1.3.1 PrefixCodeTreeRebuild 3.1.1.1.3.2 PrefixCodeTreeAddLeaf 3.1.1.1.3.3 SortSymbols 3.1.1.1.3.4 CompareSymbols 3.1.1.1.3.5 BitstringInit 3.1.1.1.3.6 BitstringLookup 3.1.1.1.3.7 BitstreamSkip 3.1.1.1.3.8 PrefixCodeTreeDecodeSymbol A new section, 3.1.1.2, Decompression, has been added: FrsTransport methods that compress data will always return information specifying the size of the original data. It is the caller's responsibility to determine whether the returned data is compressed. If the size of the compressed data buffer that is returned by the server in bytes is equal to the size in bytes of the original uncompressed data, then the buffer returned by the server contains uncompressed data. In Section 3.2.4.1.7, RequestRecords (Opnum 6), the description of the compressedRecords field has been changed from: compressedRecords: The data records, compressed using the DFS-R compression algorithm specified in section 3.1.1.1. The compressedRecords bytes correspond to an array of FRS_ID_GVSN entries. DFS-R uses custom marshaling in this RPC call to compress the set of transmitted records. The size of the FRS_ID_GVSN array is given by the numRecords parameter. The decompression algorithm specified in section 3.1.1.1.3.9 can be used to decompress the received data into a buffer of sizeof(FRS_ID_GVSN)*numRecords bytes, which can be re-interpreted as an array of FRS_ID_GVSN entries. Changed to: compressedRecords: The data records, compressed using the algorithm specified in section 3.1.1.1. The compressedRecords bytes correspond to an array of FRS_ID_GVSN entries. DFS-R uses custom marshaling in this RPC call to compress the set of transmitted records. The size of the FRS_ID_GVSN array is given by the numRecords parameter. The decompression algorithm specified in section 3.1.1.1 can be used to decompress the received data into a buffer of sizeof(FRS_ID_GVSN)*numRecords bytes, which can be re-interpreted as an array of FRS_ID_GVSN entries. In Section 3.2.4.1.14, InitializeFileTransferAsync (Opnum 13), changed from: 2. An encapsulation of the marshaled file data stream using the compressed data

Errata Published*	Description
	format (as specified in section 3.2.4.1.14.2) generated by the DFS-R compression algorithm specified in section 3.1.1.1. Even if the marshaled file data stream is not compressed by the server, it is still encapsulated using the compressed data format. Changed to: 2. An encapsulation of the marshaled file data stream using the compressed data format (as specified in section 3.2.4.1.14.2) generated by the compression algorithm specified in section 3.1.1.1. Even if the marshaled file data stream is not compressed by the server, it is still encapsulated using the compressed data format.

*Date format: YYYY/MM/DD

[MS-FSA]: File System Algorithms

This topic lists the Errata found in the MS-FSA document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

March 2, 2016 - Download

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

September 15, 2017 - Download

December 1, 2017 - Download

September 12, 2018 - Download

June 24, 2019 – Download

March 4, 2020 - Download

Errata below are for Protocol Document Version V30.0 – 2020/03/04.

Errata Published*	Description
2020/07/06	In Section 2.1.5.9.22, FSCTL_QUERY_FILE_REGIONS, added a new behavior note. Changed from: • Set InputRegion.DesiredUsage = FILE_REGION_USAGE_VALID_CACHED_DATA for NTFS or Set InputRegion.DesiredUsage = FILE_REGION_USAGE_VALID_NONCACHED_DATA for ReFS Changed to: • Set InputRegion.DesiredUsage = FILE_REGION_USAGE_VALID_CACHED_DATA for NTFS or Set InputRegion.DesiredUsage = FILE_REGION_USAGE_VALID_NONCACHED_DATA for ReFS<99> <99> Section 2.1.5.9.22: In Windows Server 2012 R2, InputRegion.DesiredUsage is set to FILE_REGION_USAGE_VALID_CACHED_DATA for ReFS.
2020/06/22	In Section 2.1.1.1, Per Volume, added definition for the NumberOfDataCopies field. Changed from:

Errata Published*	Description
	• FreeSpareBlocks: A 32-bit unsigned integer indicating the available number of spare blocks. The following fields are specific to the ReFS object store: Changed to: • FreeSpareBlocks: A 32-bit unsigned integer indicating the available number of spare blocks. • NumberOfDataCopies: A 32-bit unsigned integer indicating the number of copies of redundant data that are available on this volume. A volume with redundant copies of data MUST set this to 2 or greater. A volume without redundancy MUST have a value of 1. For example, a 2-way mirrored volume would have 2 copies and a 3-way mirrored volume would have 3 copies. Volumes configured with RAID should have a value of 2 or larger depending on which raid configuration is used. The following fields are specific to the ReFS object store: In Section 2.1.1.6, Per Open, added definition for the ReadCopyNumber field. Changed from: • UserSetAccessTime: A Boolean that is TRUE if a user has explicitly set File.LastAccessTime through this Open. • NextEaEntry: Contains a reference to the next FILE_FULL_EA_INFORMATION entry in File.ExtendedAttributes to be returned the next time FileFullEaInformation is called using this Open as defined in section 2.1.5.11.12.<39> Changed to: • UserSetAccessTime: A Boolean that is TRUE if a user has explicitly set File.LastAccessTime through this Open. • ReadCopyNumber: A 32-bit unsigned integer which is initialized to a value of 0XFFFFFFFF. Identifies which copy of data should be read from a volume with redundant data (where Volume.NumberOfDataCopies > 1). The CopyNumber is zero based, meaning zero reads the 1st copy, 1 reads the 2nd copy, etc. • NextEaEntry: Contains a reference to the next FILE_FULL_EA_INFORMATION entry in File.ExtendedAttributes to be returned the next time FileFullEaInformation is called using this Open as defined in section 2.1.5.11.12.<39> In Section 2.1.5.2, Server Requests a Read, revised/added processing for the BytesToRead. Changed from: • Set BytesToRead to BlockAlign(ByteCount,Open.File.Volume.LogicalBytesPerSector). • Read BytesToRead bytes from the disk at offset ByteOffset for this stream into OutputBuffer. If the read from the disk failed, the operation MUST be failed with the same error status. • If RequestedByteCount > ByteCount, zero out OutputBuffer between ByteCount and RequestedByteCount. Changed to: • Set BytesToRead to BlockAlign(ByteCount,Open.File.Volume.LogicalBytesPerSector). • Read BytesToRead bytes from the disk at offset ByteOffset for this stream into OutputBuffer. If Open. ReadCopyNumber != 0XFFFFFFFF then include this information in the read request to the disk to indicate which copy the data should be read from. If the read from the disk failed, the operation MUST be failed with the same error status. • If RequestedByteCount > ByteCount, zero out OutputBuffer between ByteCount and RequestedByteCount.

Errata Published*	Description
	Added a new Section 2.1.5.9.17, FSCTL_MARK_HANDLE. 2.1.5.9.17 FSCTL_MARK_HANDLE The server provides: ● Open: An Open of a DataFile. ● InputBufferSize: The byte count of the InputBuffer. ● InputBuffer: A buffer of type MARK_HANDLE_INFO as defined in [MS-FSCC] section 2.3.31. Upon completion, the object store MUST return: ● Status: An NTSTATUS code that specifies the result. Support for this operation is optional. If the object store does not implement this functionality, the operation MUST be failed with STATUS_INVALID_DEVICE_REQUEST. <93> Pseudocode for the operation is as follows: ● If InputBufferSize is less than the size of the MARK_HANDLE_INFO structure, the operation MUST be failed with STATUS_BUFFER_TOO_SMALL. ● If Open.Stream.StreamType == DirectoryStream, the operation MUST be failed with STATUS_DIRECTORY_NOT_SUPPORTED. ● STATUS_INVALID_PARAMETER is returned if: ● InputBuffer.HandleInfo contains any flag other than one and only one of either MARK_HANDLE_READ_COPY or MARK_HANDLE_NOT_READ_COPY. ● Open.Mode.FILE_NO_INTERMEDIATE_BUFFERING was not specified at open time, meaning the file was opened for cached IO operations. ● If InputBuffer.CopyNumber > (Open.File.Volume.NumberOfDataCopies – 1). ● If Open.Stream.StreamType != DataStream. ● If InputBuffer.HandleInfo has MARK_HANDLE_READ_COPY set: ● If Open.File.Volume.NumberOfDataCopies < 2, the operation MUST be failed with STATUS_NOT_REDUNDANT_STORAGE. ● If Open.Stream.IsCompressed is TRUE, the operation MUST be failed with STATUS_COMPRESSED_FILE_NOT_SUPPORTED. ● Set Open.ReadCopyNumber = InputBuffer.CopyNumber. ● Else If InputBuffer.HandleInfo has MARK_HANDLE_NOT_READ_COPY set: ● Set Open.ReadCopyNumber = 0xffffffff. ● EndIf Upon successful completion of the operation, the object store MUST return: ● Status set to STATUS_SUCCESS. In Section 6, Appendix A: Product Behavior, added product behavior note to support new section 2.1.5.9.17. <93> Section 2.1.5.9.17: This operation is only supported on the NTFS and ReFS file systems. This feature is supported in Windows Server 2019 and later.
2020/04/27	In Section 2.1.5.1, Server Requests an Open of a File, the following was changed from: ● If Link.File.FileType is not DirectoryFile, the operation MUST be failed with STATUS_NOT_A_DIRECTORY. Changed to: ● If Link.File.FileType is not DirectoryFile, the operation MUST be failed with

Errata Published*	Description
	STATUS_OBJECT_PATH_NOT_FOUND.
2020/04/27	In Section 2.1.4.12, Algorithm to Check for an Oplock Break, the following has been added: ● OPERATION_MASK – a constant that MUST contain the following value: ● (LEVEL_ONE_OPLOCK|LEVEL_TWO_OPLOCK|BATCH_OPLOCK|READ_CACHING|WRITE_CACHING|HANDLE_CACHING) The following was changed from: ● If OpParams.DesiredAccess contains no flags other than FILE_READ_ATTRIBUTES, FILE_WRITE_ATTRIBUTES, or SYNCHRONIZE, the algorithm returns at this point. Changed to: ● If (((OpParams.DesiredAccess contains no flags other than FILE_READ_ATTRIBUTES, FILE_WRITE_ATTRIBUTES, READ_CONTROL, or SYNCHRONIZE) and (Oplock.State anded with OPERATION_MASK) contains no flags other than READ_CACHING, WRITE_CACHING, or HANDLE_CACHING)) or ((OpParams.DesiredAccess contains no flags other than FILE_READ_ATTRIBUTES, FILE_WRITE_ATTRIBUTES or SYNCHRONIZE) and (Oplock.State anded with OPERATION_MASK) contains no flags other than LEVEL_TWO_OPLOCK, LEVEL_ONE_OPLOCK or BATCH_OPLOCK))), the algorithm returns at this point.
2020/04/27	In Section 2.1.5.9.21, FSCTL_QUERY_FILE_REGIONS, the following was added: Support for this operation is optional. If the object store does not implement this functionality, this operation MUST be failed with STATUS_INVALID_DEVICE_REQUEST.<97> <97> Section 2.1.5.9.21: This operation is only supported by the NTFS and ReFS file systems.
2020/03/30	Section 2.1.5.9.33 FSCTL_SET_SHORT_NAME_BEHAVIOR has been removed from the document. Removed: 2.1.5.9.33 FSCTL_SET_SHORT_NAME_BEHAVIOR This control code is reserved for the WinPE<118> environment; the object store MUST return STATUS _INVALID_DEVICE_REQUEST. <118>WinPE stands for the windows Preinstallation Environment. For more information please see [MSFT-WinPE].

*Date format: YYYY/MM/DD

[MS-FSCC]: File System Control Codes

This topic lists the Errata found in the MS-FSCC document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

September 15, 2017 - Download

December 1, 2017 - Download

September 12, 2018 - Download

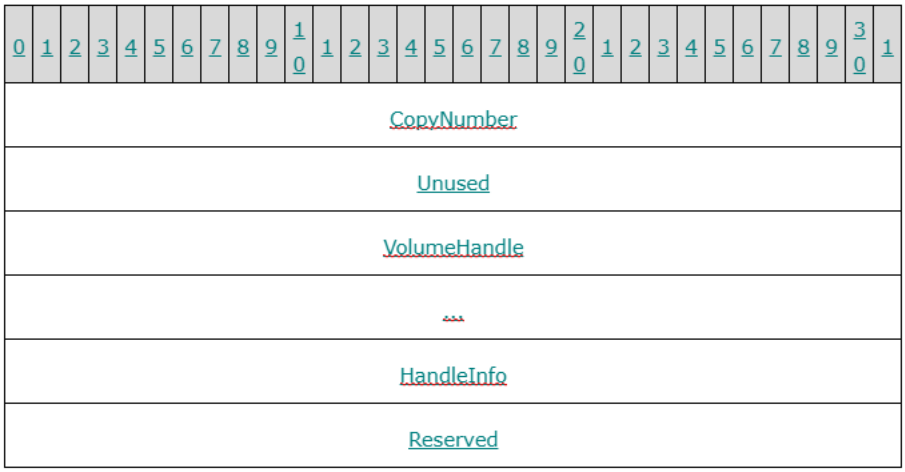
September 23, 2019 - Download

March 4, 2020 - Download

Errata below are for Protocol Document Version V47.0 – 2020/03/04.

Errata Published*	Description
2020/06/22	Added a new Section 2.3.31, FSCTL_MARK_HANDLE Request. 2.3.31 FSCTL_MARK_HANDLE Request The FSCTL_MARK_HANDLE request is used to set specific operational state on the given file handle. This state is lost once the handle is closed.<29> The MARK_HANDLE_INFO element is as follows:

Errata Published*	Description



CopyNumber (4 bytes): A 32-bit unsigned integer that identifies, when reading from a file which resides on redundant media, which copy to read.

Unused (4 bytes): Reserved for alignment. This field can contain any value and MUST be ignored.

VolumeHandle (8 bytes): A 64-bit HANDLE that is not used and MUST be set to zero.

HandleInfo (4 bytes): A 32-bit unsigned integer containing flags to identify the request. Only one of the following values can be set:

Value	Meaning
MARK_HANDLE_READ_COPY 0x00000080	When a file resides on redundant media (ex: mirrored or RAID) this tells the file system that read operations on this handle should only come from the specified copy of data. When this state is not set a file system will return data from any copy available as it sees fit. This operation is typically used by scrubber applications that want to validate the contents of all copies of data for a given file.
MARK_HANDLE_NOT_READ_COPY 0x00000100	When a file resides on redundant media (ex: mirrored or RAID) this tells the file system that read operations on this handle may come from any copy of the data as the file system sees fit. This turns off reading from a specific copy.

Reserved (4 Bytes): A 32-bit field. This field is reserved. This field SHOULD be set to 0, and MUST be ignored.

Added a new Section 2.3.32, FSCTL_MARK_HANDLE.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
2.3.32 FSCTL_MARK_HANDLE Reply
This message returns the results of the FSCTL_MARK_HANDLE request.

The only data item this message returns is a status code, as specified in section 2.2. Upon success, the status code returned by the function that processes this FSCTL is STATUS_SUCCESS. The most common error codes are listed in the following table.

<table>
<tr><th>Error code</th><th>Meaning</th></tr>
<tr><td>STATUS_INVALID_PARAMETER
0xC000000D</td><td>This status is returned if:
▪ HandleInfo contains any flag other than one and only one of either MARK_HANDLE_READ_COPY or MARK_HANDLE_NOT_READ_COPY
▪ The file was opened for cached IO
▪ The specified copy number is greater than the number of available redundant copies</td></tr>
<tr><td>STATUS_DIRECTORY_NOT_SUPPORTED
0xC000047C</td><td>This operation is not supported on directory files.</td></tr>
<tr><td>STATUS_NOT_REDUNDANT_STORAGE
0xC0000479</td><td>This operation is only supported on redundant media.</td></tr>
<tr><td>STATUS_COMPRESSED_FILE_NOT_SUPPORTED
0xC000047B</td><td>This operation is not supported on compressed files.</td></tr>
</table>

In Section 6, Appendix B: Product Behavior, added a new product behavior note to support new section 2.3.31.

Added:

<29> Section 2.3.31: This operation is supported only by the NTFS and ReFS file systems.
</td></tr>
<tr><td>2020/04/27</td><td>
In Section 2.7.1, FILE_NOTIFY_INFORMATION, the following was changed from:

FILE_ACTION_REMOVED

0x00000002 The file was removed from the directory.

FILE_ACTION_MODIFIED

0x00000003 The file was modified. This can be a change to the data or attributes of the file.

FILE_ACTION_RENAMED_OLD_NAME
</td></tr>
</table>

Errata Published*	Description
	0x00000004 The file was renamed, and this is the old name. If the new name resides within the directory being monitored, the client also receives the FILE_ACTION_RENAMED_NEW_NAME bit value as described in the next list item. If the new name resides outside of the directory being monitored, the client will not receive the FILE_ACTION_RENAMED_NEW_NAME bit value. FILE_ACTION_RENAMED_NEW_NAME 0x00000005 The file was renamed, and this is the new name. If the old name resides within the directory being monitored, the client will also receive the FILE_ACTION_RENAME_OLD_NAME bit value. If the old name resides outside of the directory being monitored, the client will not receive the FILE_ACTION_RENAME_OLD_NAME bit value. Changed to: FILE_ACTION_REMOVED 0x00000002 The file was removed from the directory. When a file is renamed to a different directory the client will receive this notification along with FILE_ACTION_MODIFIED. FILE_ACTION_MODIFIED 0x00000003 The file was modified. This can be a change to the data or attributes of the file. When a file is renamed to a different directory the client will receive this notification along with FILE_ACTION_REMOVED. FILE_ACTION_RENAMED_OLD_NAME 0x00000004 The file was renamed, and this is the old name. This notification is only sent when the rename operation does not change the directory the file resides in. The client will also receive a FILE_ACTION_RENAMED_NEW_NAME notification. This notification will not be received if the file is renamed to a different directory. FILE_ACTION_RENAMED_NEW_NAME 0x00000005 The file was renamed, and this is the new name. This notification is only sent when the rename operation does not change the directory the file resides in. The client will also receive a FILE_ACTION_RENAME_OLD_NAME notification. This notification will not be received if the file is renamed to a different directory.
2020/04/27	In Section 6, Appendix B: Product Behavior, the following was changed in behavior note <10> Section 2.1.9 from: NTFS computes the 64-bit file ID as follows: 48 bits are the index of the file's primary record in the master file table (MFT), and the other 16 bits are a sequence number. Therefore, it is possible that a different file can have the same 64-bit file ID as a file on that volume had in the past. Changed to NTFS computes the 64-bit file ID as follows: the low 48 bits are the index of the file's primary record in the master file table (MFT); the remaining 16 bits are a sequence number. Therefore, it is possible, though rare, that a different file can have the same 64-bit file ID as a file on that volume had in the past.

Errata Published*	Description
	ReFS maps a subset of the possible 128-bit file ID values to a 64-bit value using a reversible algorithm; for values outside of this subset, ReFS sets the 64-bit file ID to -1. The following was added to behavior note <11> in Section 2.1.10: NTFS computes the 128-bit file ID as follows: the low 48 bits are the index of the file's primary record in the master file table (MFT), the next 16 bits are a sequence number, and the high 64 bits MUST be zero. Therefore, it is possible, though rare, that a different file can have the same 128-bit file ID as a file on that volume had in the past. ReFS computes the 128-bit file ID as follows: the low 64 bits consists of an index uniquely identifying the file's parent directory on the volume. The high 64-bits consists of an index uniquely identifying the file within that directory.

*Date format: YYYY/MM/DD

[MS-FSRVP]: File Server Remote VSS Protocol

This topic lists the Errata found in the MS-FSRVP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

[MS-FSVCA]: File Set Version Comparison Algorithms

This topic lists the Errata found in the MS-FSVCA document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-GPPREF]: Group Policy: Preferences Extension Data Structure

This topic lists the Errata found in [MS-GPPREF] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

September 12, 2018 - Download

[MS-GPSB]: Group Policy: Security Protocol Extension

This topic lists the Errata found in [MS-GPSB] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

September 12, 2018 - Download

[MS-GPOL]: Group Policy: Core Protocol

This topic lists the Errata found in [MS-GPOL] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

June 1, 2017 - Download

Errata below are for Protocol Document Version V36.1 – 2019/03/15.

Errata Published*	Description
2019/05/27	In Section 2.2.4, GPO Search, changed from: The gpt.ini file MUST be encoded in UTF-8 and is described with the following Augmented Backus-Naur Form (ABNF), as specified in [RFC4234]. Changed to: The gpt.ini file MUST be encoded in ANSI and is described with the following Augmented Backus-Naur Form (ABNF), as specified in [RFC4234].

*Date format: YYYY/MM/DD

[MS-GPWL]: Group Policy: Wireless/Wired Protocol Extension

This topic lists the Errata found in [MS-GPWL] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

Errata below are for Protocol Document Version V27.0 – 2020/03/04.

Errata Published*	Description
2020/06/22	In Section 2.2.1.2.1, Message Syntax for XML-Based Wireless Profiles, added values and enumerations to phyType. Changed from: autoSwitch: If the connection to a more preferred network is attempted when already connected to a network. A more preferred network is one that is ordered higher in a list of preferred wireless networks. phyType: The IEEE 802.11 physical type that a domain client uses while connected to this wireless network. authentication: The type of 802.11 authentication the domain clients uses for connecting to the WLAN. This value MUST be one of the following: … Changed to: autoSwitch: If the connection to a more preferred network is attempted when already connected to a network. A more preferred network is one that is ordered higher in a list of preferred wireless networks. phyType: The IEEE 802.11 physical type that a domain client uses while connected to this wireless network. This value MUST be one of the following: ▪ a: refers to LAN protocol IEEE 802.11a-1999 ▪ b: refers to LAN protocol IEEE 802.11b-1999 ▪ g: refers to LAN protocol IEEE 802.11g-2003 ▪ n: refers to LAN protocol IEEE 802.11n-2009 ▪ ac: refers to LAN protocol IEEE 802.11ac-2013 ▪ ax: refers to LAN protocol IEEE 802.11ax authentication: The type of 802.11 authentication the domain clients uses for connecting to the

Errata Published*	Description

	WLAN. This value MUST be one of the following:
	…
	
	In Section 6.3.1, Wireless LAN Profile v1 Schema, added values to phyType.
	
	Changed from:
	
	`<xs:element name="phyType" minOccurs="0" maxOccurs="4">`
	`<xs:simpleType>`
	`<xs:restriction base="xs:string">`
	`<xs:enumeration value="a" />`
	`<xs:enumeration value="b" />`
	`<xs:enumeration value="g" />`
	`<!-- this value is reserved for future use -->`
	`<xs:enumeration value="n" />`
	`<xs:enumeration value="ac" />`
	`</xs:restriction>`
	`</xs:simpleType>`
	`</xs:element>`
	
	Changed to:
	
	`<xs:element name="phyType" minOccurs="0" maxOccurs="6">`
	`<xs:simpleType>`
	`<xs:restriction base="xs:string">`
	`<xs:enumeration value="a" />`
	`\<xs:enumeration value="b" />`
	`\<xs:enumeration value="g" />`
	`<xs:enumeration value="n" />`
	`<xs:enumeration value="ac" />`
	`<xs:enumeration value="ax" />`
	`</xs:restriction>`

Errata Published*	Description
	</xs:simpleType> </xs:element>

*Date format: YYYY/MM/DD

[MS-GSSA]: Generic Security Service Algorithm for Secret Key Transaction Authentication for DNS (GSS-TSIG) Protocol Extension

This topic lists the Errata found in the MS-GSSA document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-HGSA]: Host Guardian Service: Attestation Protocol

This topic lists the Errata found in the MS-HGSA document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

September 15, 2017 - Download

September 12, 2018 - Download

June 24, 2019 – Download

[MS-HTTPE]: Hypertext Transfer Protocol (HTTP) Extensions

This topic lists the Errata found in [MS-HTTPE] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-HVRS]: Hyper-V Remote Storage Profile

This topic lists the Errata found in [MS-HVRS] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

March 20, 2017 - Download

[MS-ICPR]: ICertPassage Remote Protocol

This topic lists the Errata found in the MS-ICPR document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-IKEE]: Internet Key Exchange Protocol Extensions

This topic lists the Errata found in the MS-IKEE document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

September 15, 2017 - Download

Errata below are for Protocol Document Version V27.0 – 2018/09/12.

Errata Published*	Description
2019/10/28	In Section 2.2.8, Configuration Attribute (IKEv2) Packet, changed from: Length (2 bytes): The length of the data in the value field. Changed to: Length (2 bytes): The length of the data in the Value field. In Section 2.2.11.2, Encrypted Fragment Payload, changed from: Next_Payload (1 byte): In the very first fragment (with Fragment Number equal to 1), this field MUST be set to the payload type of the first inner payload. In the remainder of the Fragment messages (with Fragment Number greater than 1), this field MUST be set to zero. Changed to: Next_Payload (1 byte): In the very first fragment (with Fragment_Number equal to 1), this field MUST be set to the payload type of the first inner payload. In the remainder of the Fragment messages (with Fragment_Number greater than 1), this field MUST be set to zero. In Section 3.3.1, Abstract Data Model, references have been added o disambiguate which fields in section 2.2.3.1 set the values of the ADM elements: Fragment ID, Fragment Number, Flag, and Fragment Data. Changed from: Fragment queue: A queue holding the fragments that correspond to incomplete IKE messages, indexed by the Fragment ID. Each entry in the queue MUST contain: -- The Fragment ID -- The Fragment Number -- A Flag that indicates whether this fragment is the last one (that is, the LAST_FRAGMENT

Errata Published*	Description
	bit is set in the Fragment payload). -- The Fragment Data For definitions of the previous values, see section 2.2.3.1. Flow state table: The following information MUST be maintained. Changed to: Fragment queue: A queue holding the fragments that correspond to incomplete IKE messages, indexed by the Fragment ID. Each entry in the queue MUST contain: -- The Fragment ID, which is set to the Fragment_ID field in section 2.2.3.1. -- The Fragment Number, which is set to the Fragment_Number field in section 2.2.3.1. -- A Flag that is set to the Flags field in section 2.2.3.1 to indicates whether this fragment is the last one (that is, the LAST_FRAGMENT bit is set in the Fragment payload). -- The Fragment Data, which is set to the Fragment_Data field in section 2.2.3.1. Flow state table: The following information MUST be maintained. In Section 3.3.2, Timers, the second bullet point has been changed from: When the fragmentation reassembly timer fires, the delay MUST NOT exceed 90 seconds.<17> Changed to: When the fragment reassembly timer fires, the delay MUST NOT exceed 90 seconds.<17> In Section 3.3.5.3, Receiving Other IKE Messages, the action taken by the Receiver upon receipt of an IKE message (to discard such a message when a Fragment payload is present and it is not the only payload in the message) has been clarified. Changed from: On receipt of an IKE message, the host MUST check if the message contains a Fragment payload. If a Fragment payload is present, this payload MUST be the only payload in the message. If not, the host MUST silently discard the message. Changed to: On receipt of an IKE message, the host MUST check if the message contains a Fragment payload. If a Fragment payload is present, and the payload is not the only payload in the message, the host MUST silently discard the message' In Section 3.3.5.3, Receiving Other IKE Messages, text has been changed to clarify from where to retrieve the Fragment ID. Changed from: Retrieve the Fragment ID from the Fragment payload.

Errata Published*	Description
	Changed to: Retrieve the Fragment ID from the Fragment_ID field in the Fragment payload. In Section 3.3.5.3, Receiving Other IKE Messages, text has been changed to clarify how fragments not of the same Fragment Number are added to the Fragment queue in the corresponding entry of the MMSAD. Changed from: If the queue for this Fragment ID already contains a fragment with the same Fragment Number, the host MUST silently discard the message. If not, the host MUST queue the Fragment payload's fields in the corresponding entry of the MMSAD, indexed by the Fragment Id Changed to: If the queue for this Fragment ID already contains a fragment with the same Fragment Number, the host MUST silently discard the message. If not, the host MUST add an entry to the Fragment queue in the corresponding entry of the MMSAD, with the queue entry fields initialized based on the associated fields of the Fragment payload. In Section 3.3.5.3, Receiving Other IKE Messages, changed from: The host MUST then check whether all Fragment payloads for this Fragment ID have been received (that is, whether Fragment payloads that have a Fragment number from 1 to n.. Changed to: The host MUST then check whether all Fragment payloads for this Fragment ID have been received (that is, whether Fragment payloads that have a Fragment Number from 1 to n.. In Section 3.3.5.3, Receiving Other IKE Messages, text has been changed to clarify the error condition where the host MUST discard all Fragment payloads for a specific Fragment ID. Changed from: A Fragment payload has been received with a Fragment number greater than the Fragment number of the fragment with the Flags field set to LAST_FRAGMENT.' Changed to: A Fragment payload has been received with a Fragment Number greater than the Fragment Number of an entry in the Fragment queue with the Flags field set to LAST_FRAGMENT. In Section 3.3.5.3, Receiving Other IKE Messages, changed from: Fragment payloads (without the Fragment payload header) in the order of their Fragment number. Changed to:

Errata Published*	Description
	Fragment payloads (without the Fragment payload header) in the order of their Fragment Number. In Section 3.15.1, Abstract Data Model, references have been added to disambiguate which fields in section 2.2.3.1 set the values of the ADM elements: Fragment ID, Fragment Number, and Fragment Data. Changed from: Fragment queue: A queue holding the fragments that correspond to incomplete IKE messages, indexed by the Fragment ID. Each entry in the queue MUST contain the following: Fragment ID, which is the Message ID Fragment Number Total Fragments Fragment Data Flow state table: The following information MUST be maintained. Changed to: Fragment queue: A queue holding the fragments that correspond to incomplete IKE messages, indexed by the Fragment ID. Each entry in the queue MUST contain the following: Fragment ID, which is the Message ID, is set to the Fragment_ID field in section 2.2.3.1. Fragment Number, which is set to the Fragment_Number field in section 2.2.3.1. Total Fragments Fragment Data, which is set to the Fragment_Data field in section 2.2.3.1. Flow state table: The following information MUST be maintained.

*Date format: YYYY/MM/DD

[MS-IPAMM2]: IP Address Management (IPAM) Management Protocol Version 2

This topic lists the Errata found in [MS-IPAMM2] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

June 1, 2017 - Download

[MS-IPHTTPS]: IP over HTTPS (IP-HTTPS) Tunneling Protocol

This topic lists the Errata found in the MS-IPHTTPS document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 12, 2018 - Download

[MS-IRP]: Internet Information Services (IIS) Inetinfo Remote Protocol

This topic lists the Errata found in [MS-IRP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-KILE]: Kerberos Protocol Extensions

This topic lists the Errata found in [MS-KILE] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

March 4, 2020 - Download

Errata below are for Protocol Document Version V35.0 – 2020/03/04.

Errata Published*	Description
2020/07/20	In Section 3.4.5.3, Processing Authorization Data, added processing for searching AD-IF-RELEVANT containers for authorization data. Changed from: The server MUST check if KERB-AD-RESTRICTION-ENTRY.Restriction.MachineID (section 2.2.6) is equal to Machine ID (section 3.1.1.4): Changed to: The server MUST search all AD-IF-RELEVANT containers for the KERB_AUTH_DATA_TOKEN_RESTRICTIONS and KERB_AUTH_DATA_LOOPBACK authorization data entries. The server MAY<76> search all AD-IF-RELEVANT containers for all other authorization data entries. The server MUST check if KERB-AD-RESTRICTION-ENTRY.Restriction.MachineID (section 2.2.6) is equal to machine ID (section 3.1.1.4): <76> Windows only searches the first AD-IF-RELEVANT container. In Section 3.2.5.8, AP Exchange, updated AD-AUTH-DATA-AP-OPTIONS is sent in the first AD-IF-RELEVANT element. Changed from: If ChannelBinding is set to TRUE, the client sends AD-AUTH-DATA-AP-OPTIONS data in an AD-IF-RELEVANT element ([RFC4120] section 5.2.6.1). Changed to: If ChannelBinding is set to TRUE, the client sends AD-AUTH-DATA-AP-OPTIONS data in the first AD-IF-RELEVANT element ([RFC4120] section 5.2.6.1).
2020/05/11	In Section 3.3.5.7.5, Cross-Domain Trust and Referrals, updated product support for the TRUST_ATTRIBUTE_CROSS_ORGANIZATION_ENABLE_TGT_DELEGATION flag.

Errata Published*	Description
	Changed from: <67> Section 3.3.5.7.5: The TRUST_ATTRIBUTE_CROSS_ORGANIZATION_ENABLE_TGT_DELEGATION flag is supported on Windows Server 2003 and later when [MSKB-4490425] is installed. Changed to:<67> Section 3.3.5.7.5: The TRUST_ATTRIBUTE_CROSS_ORGANIZATION_ENABLE_TGT_DELEGATION flag is supported on Windows Server 2008 and later when [MSKB-4490425] is installed.
2020/04/27	In Section 3.3.5.7.8, Key List Request, added reference to [RFC6806] to define EncKDCRepPart structure. Changed from: … the KDC SHOULD include the long-term secrets of the client for the requested encryption types in the KERB-KEY-LIST-REP [162] response message and insert it into the encrypted-pa-data of the EncKDCRepPart.<69> Changed to: … the KDC SHOULD include the long-term secrets of the client for the requested encryption types in the KERB-KEY-LIST-REP [162] response message and insert it into the encrypted-pa-data of the EncKDCRepPart structure, as defined in [RFC6806].<69>
2020/04/27	In Section 3.3.5.7.6, FORWARDED TGT etype, added PA data number to PA-SUPPORTED-ENCTYPES [165]. Changed from: … the client provides a PA-SUPPORTED-ENCTYPES structure (section 2.2.8) with encryption types (section 2.2.7) the KDC supports, then the KDC SHOULD<68> select the strongest encryption type that is both included in the PA-SUPPORTED-ENCTYPES structure (section 2.2.8) and supported by the KDC to generate the random session key. Changed to:… the client provides a PA-SUPPORTED-ENCTYPES [165] structure (section 2.2.8) with encryption types (section 2.2.7) the KDC supports, then the KDC SHOULD<68> select the strongest encryption type that is both included in the PA-SUPPORTED-ENCTYPES [165] structure (section 2.2.8) and supported by the KDC to generate the random session key
2020/04/27	In Section 3.3.5.3, PAC Generation, removed PA data number [128] as not part of KERB-PA-PAC-REQUEST Boolean structure. Changed from: The request to include a PAC is expressed through the use of a KERB-PA-PAC-REQUEST [128] (section 2.2.3) padata type that is set to TRUE: Changed to:The request to include a PAC is expressed through the use of a KERB-PA-PAC-REQUEST (section 2.2.3) padata type that is set to TRUE:
2020/04/27	In Section 1.3.2, Kerberos Network Authentication Service (V5) Synopsis, added product note for addition of PA-Data in the TGS-REQ and TGS-REP messages. Changed from: The Ticket-Granting Service (TGS) exchange ([RFC4120] section 3.3): Kerberos ticket-granting service (TGS) request message (KRB_TGS_REQ)…

Errata Published*	Description
	• Kerberos ticket-granting service (TGS) response message (KRB_TGS_REP)… Changed to: The Ticket-Granting Service (TGS) exchange ([RFC4120] section 3.3):<1> • Kerberos ticket-granting service (TGS) request message (KRB_TGS_REQ)… • Kerberos ticket-granting service (TGS) response message (KRB_TGS_REP)… <1>Added a PA-Data request in the TGS-REQ message and an encrypted PA-Data response in the TGS-REP message that includes the NTLM hash for the authenticated user in Windows 10 v1607 operating system client version and in Windows Server 2016 server version and later.
2020/04/13	In Section 3.1.5.4, Ticket Flag Details, the description of the transit policy enforcement has been clarified. Changed from: The TRANSITED-POLICY-CHECKED flag ([RFC4120] section 2.7): KILE does not check for transited domains on servers or a KDC. Application servers MUST ignore the TRANSITED-POLICY-CHECKED flag. Changed To: The TRANSITED-POLICY-CHECKED flag ([RFC4120] section 2.7): KILE does not check for transited domains on servers or a KDC. Application servers MUST ignore the TRANSITED-POLICY-CHECKED flag. For details on decoding a cross-realm TGTandcrealm filtering see [MS-PAC] section 4.1.2.3.

*Date format: YYYY/MM/DD

[MS-KPP]: Key Provisioning Protocol

This topic lists the Errata found in the MS-KPP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-KPS]: Key Protection Service Protocol

This topic lists the Errata found in the MS-KPP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

September 15, 2017 - Download

[MS-LCID]: Windows Language Code Identifier (LCID) Reference

This topic lists the Errata found in [MS-LCID] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 15, 2017 - Download

December 1, 2017 - Download

September 12, 2018 - Download

March 13, 2019 - Download

[MS-LSAD]: Local Security Authority (Domain Policy) Remote Protocol

This topic lists the Errata found in [MS-LSAD] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

March 16, 2018 - Download

Errata below are for Protocol Document Version V43.0 – 2019/09/12.

Errata Published*	Description
2019/10/16	In Section 2.2.4.4, LSAPR_POLICY_AUDIT_EVENTS_INFO: Changed from: MaximumAuditingEventCount Changed to: MaximumAuditEventCount

*Date format: YYYY/MM/DD

[MS-LSAT]: Local Security Authority (Translation Methods) Remote Protocol

This topic lists the Errata found in [MS-LSAT] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

June 1, 2017 - Download

June 1, 2017 - Download

September 15, 2017 - Download

[MS-MDE]: Mobile Device Enrollment Protocol

This topic lists the Errata found in [MS-MDE] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

September 15, 2017 - Download

[MS-MDE2]: Mobile Device Enrollment Protocol Version 2

This topic lists the Errata found in [MS-MDE2] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

September 15, 2017 - Download

December 1, 2017 - Download

September 12, 2018 - Download

[MS-MDM]: Mobile Device Management Protocol

This topic lists the Errata found in [MS-MDM] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

December 1, 2017 - Download

[MS-MICE]: Miracast over infrastructure Connection Establishment Protocol

This topic lists the Errata found in [MS-MICE] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 12, 2018 - Download

[MS-MSSOD]: Media Streaming Server Protocols Overview

This topic lists the Errata found in [MS-MSSOD] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

December 1, 2017 - Download

[MS-MWBE]: Microsoft Web Browser Federated Sign-On Protocol Extensions

This topic lists the Errata found in [MS-MWBE] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous version of this document, see the following ERRATA archive:

June 30, 2015 - Download

[MS-MWBF]: Microsoft Web Browser Federated Sign-On Protocol

This topic lists the Errata found in [MS-MWBF] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

June 1, 2017 - Download

September 12, 2018 - Download

[MS-NCNBI]: Network Controller Northbound Interface Specification

This topic lists the Errata found in the MS-NCNBI document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

March 16, 2018 - Download

September 12, 2018 - Download

Errata below are for Protocol Document Version V6.0 – 2018/09/12.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td>2018/12/17</td><td>
In several sections throughout this document, missing element Type designations have been added to existing element or header tables. For example, in Section 2.2.1.2, Request Headers, the text in bold has been added to the existing table as shown below.
<table>
<tr><th>Header</th><th>Section</th><th>Type</th><th>Description</th></tr>
<tr><td>Content-Type</td><td>2.2.1.1</td><td>Required or Optional Required for PUT, must be "application/json; charset=UTF-8". Optional for GET or Delete</td><td>The content type of the payload.</td></tr>
</table>
In the following sections, the added Type designations are shown in bold.

2.2.2, Common JSON Elements
<table>
<tr><td>resourceId</td><td>Optional or Required When optional for ancestor resource, then required for descendant resource. See section 2.2.3.</td></tr>
<tr><td>resourceRef</td><td>Read-only Optional or Required See section 1.3.3.2.</td></tr>
<tr><td>properties.etag</td><td>Read-only</td></tr>
</table>
</td></tr>
</table>

Errata Published*	Description

properties.provisioningState	**Read-only**

3.1.5.1 accessControlLists

configurationState.id	**Optional Read-only**
virtualNetworkInterfaceErrors	**Optional Read-only**

3.1.5.5.3 frontendIPConfigurations

configurationState.vipEndpointStates	**Read-only**
configurationState.vipEndpointStates.vipEndpoint	**Read-only**
configurationState.vipEndpointStates.dipEndpointStates	**Read-only**
configurationState.vipEndpointStates.dipEndpointStates.dipEndpoint	**Read-only**
configurationState.vipEndpointStates.dipEndpointStates.hostIPAddress	**Read-only**
configurationState.vipEndpointStates.dipEndpointStates.hostId	**Read-only**
configurationState.vipEndpointStates.dipEndpointStates.AdapterId	**Read-only**
configurationState.vipEndpointStates.dipEndpointStates.ProbeRule	**Read-only**

3.1.5.11 networkInterfaces

dnsSettings	**Optional**
dnsSettings.dnsServers	**Optional**
ipConfigurations	**Read-only**
isHostVirtualNetworkInterface	**Optional FALSE is default. Cannot be changed after creation.**
internalDnsNameLabel	**Optional**
isPrimary	**Optional TRUE is default.**
isMultitenantStack	**Optional**
privateMacAddress	**Optional**
privateMacAllocationMethod	**Required**
serviceInsertionElements	**Optional Read-only**

Errata Published*	Description
	3.1.5.14 publicIPAddresses dnsSettings — **Optional** 3.1.5.15 servers connections — **Required** connections.credential — **Required** connections.credentialType — **Required** connections.managementAddresses — **Required** certificate — **Optional or Required Required only if the certificate used by the server is self-signed.** 3.1.5.18 virtualNetworks configurationState.id — **Optional Read-only** configurationState.hostErrors — **Optional Read-only** 3.1.5.18.3 virtualNetworkPeerings remoteVirtualNetwork — **Required** 3.1.5.21 virtualServers connections.credential — **Optional** connections.credentialType — **Optional** connections.managementAddresses — **Optional** In Section 3.1.5.26, changed from: HTTP method — Description

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><td>PUT</td><td>Create a new virtualNetworkManager resource or update an existing VirtualGateways resource.</td></tr>
<tr><td>GET</td><td>Get one virtualNetworkManager resource</td></tr>
</table>

Changed to:
<table>
<tr><td>HTTP method</td><td>Description</td></tr>
<tr><td>PUT</td><td>Update the virtualNetworkManager singleton resource.</td></tr>
<tr><td>GET</td><td>Get the virtualNetworkManager resource.</td></tr>
</table>
</td></tr>
</table>

* Date format: YYYY/MM/DD

[MS-NCT]: Network Cost Transfer Protocol

This topic lists the Errata found in the MS-NCT document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-NFPB]: Near Field Proximity Bidirectional Services Protocol

This topic lists the Errata found in [MS-NFPB] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-NFPS]: Near Field Proximity Sharing Protocol

This topic lists the Errata found in [MS-NFPS] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-NKPU]: Network Key Protector Unlock Protocol

This topic lists the Errata found in [MS-NKPU] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

December 1, 2017 - Download

[MS-NLMP]: NT LAN Manager (NTLM) Authentication Protocol

This topic lists the Errata found in [MS-NLMP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

September 23, 2019 - Download

Errata below are for Protocol Document Version V31.0 – 2019/09/23.

Errata Published*	Description

2020/08/17	In section 3.1.5.1.2 Client Receives a CHALLENGE_MESSAGE from the Server, added NTLMSSP_NEGOTIATE_SIGN and NTLMSSP_NEGOTIATE_SEAL Flags.
	
	Changed from:
	
	`If (NTLMSSP_NEGOTIATE_KEY_EXCH bit is set in CHALLENGE_MESSAGE.NegotiateFlags )`
	`Set ExportedSessionKey to NONCE(16)`
	`Set AUTHENTICATE_MESSAGE.EncryptedRandomSessionKey to RC4K(KeyExchangeKey, ExportedSessionKey)`
	`Else`
	`Set ExportedSessionKey to KeyExchangeKey`
	`Set AUTHENTICATE_MESSAGE.EncryptedRandomSessionKey to NIL`
	`Endif`
	
	Changed to:
	
	`If (NTLMSSP_NEGOTIATE_KEY_EXCH bit is set in CHALLENGE MESSAGE.NegotiateFlags AND (NTLMSSP_NEGOTIATE_SIGN OR NTLMSSP_NEGOTIATE_SEAL are set in CHALLENGE MESSAGE.NegotiateFlags))`
	`Set ExportedSessionKey to NONCE(16)`
	`Set AUTHENTICATE_MESSAGE.EncryptedRandomSessionKey to RC4K(KeyExchangeKey, ExportedSessionKey)`
	`Else`
	`Set ExportedSessionKey to KeyExchangeKey`
	`Set AUTHENTICATE_MESSAGE.EncryptedRandomSessionKey to NIL`
	`Endif`

Errata Published*	Description
	In section 3.2.5.1.2 Server Receives an AUTHENTICATE_MESSAGE from the Client, added server behavior when NTLMSSP_NEGOTIATE_KEY_EXCH is set. Changed from: If GuestSession is TRUE, a SessionBaseKey with all-zeroes, Z(16), is used. If NTLM v2 authentication is used and channel binding is provided by the application, then the server MUST verify the channel binding: <66> Changed to: If GuestSession is TRUE, a SessionBaseKey with all-zeroes, Z(16), is used. If NTLMSSP_NEGOTIATE_KEY_EXCH is set, the server MUST check if client supplied a valid EncryptedRandomSessionKey in the AUTHENTICATE_MESSAGE (section 2.2.1.3); otherwise, the server MUST return SEC_E_INVALID_TOKEN. If NTLM v2 authentication is used and channel binding is provided by the application, then the server MUST verify the channel binding:<66>
2020/08/17	In section 3.4.4.3 Without NTLMSSP_NEGOTIATE_SIGN, added section. Changed from: 3.4.4.2 With Extended Session Security . . . 3.4.5 KXKEY, SIGNKEY, and SEALKEY Changed to: 3.4.4.2 With Extended Session Security . . . 3.4.4.3 Without NTLMSSP_NEGOTIATE_SIGN When NTLMSSP_ALWAYS_NEGOTIATE_SIGN is set and message integrity (NTLMSSP_NEGOTIATE_SIGN) is not negotiated, the message signature for NTLM is a 16-byte value that contains the following components, as specified by the NTLMSSP_MESSAGE_SIGNATURE structure (section 2.2.2.9): • Version: A 4-byte number value that is set to 0x00000001. • All other bytes set to zero. 3.4.5 KXKEY, SIGNKEY, and SEALKEY
2019/12/16	In Section 3.4, Session Security Details, added ANONYMOUS user with Guest user and section reference. Changed from: For the case of Guest user login, there is no session security. Changed to: For the cases of ANONYMOUS user and Guest user login, there is no session security (see section 3.2.5.1.2). In Section 5.1, Security Considerations for Implementers, added ANONYMOUS user, Guest user,

Errata Published*	Description
	and Guest log in case 2 of 3. Changed from: The use of NullSession results in a SessionBaseKey with all zeroes, which does not provide security. Therefore, applications are generally advised not to use NullSession. The Guest user account is disabled by default in Windows for security reasons. If the Guest user account is enabled, it is strongly recommended to set a password so that logon failures do not result in Guest logins (section 3.2.5.1.2). Changed to: The use of ANONYMOUS user NullSession results in a SessionBaseKey with all zeroes, which does not provide security. Therefore, applications are generally advised not to use NullSession. The use of Guest user GuestSession results in a SessionBaseKey with all zeroes, which does not provide security. The Guest user account is disabled by default in Windows for security reasons. If the Guest user account is enabled, it is strongly recommended to set a password so that logon failures do not result in Guest logins (section 3.2.5.1.2). If a password is set on the Guest account, then there is a guest fallback where logons will be tried with unknown usernames against the Guest password.

*Date format: YYYY/MM/DD

[MS-NMFMB]: .NET Message Framing MSMQ Binding Protocol

This topic lists the Errata found in [MS-NMFMB] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

December 1, 2017 - Download

[MS-NNS]: .NET NegotiateStream Protocol

This topic lists the Errata found in [MS-NNS] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

Errata below are for Protocol Document Version V7.0 – 2017/12/01.

Errata Published*	Description
2019/02/19	In Section 2.2.2, Data Message, the maximum size of the PayloadSize field has been changed from '0x0000FC00' to '0x0000FC30', to accommodate for both the application data size and the size increase that occurs when this protocol signs or encrypts the data to be transferred. Changed from: PayloadSize (4 bytes): The unsigned size, in bytes, of the Payload field. The maximum value for this field is 0x0000FC00 (that is, 63K, or 64,512). Changed to: PayloadSize (4 bytes): The unsigned size, in bytes, of the Payload field. The maximum value for this field is 0x0000FC30 (64,560).

*Date format: YYYY/MM/DD

[MS-NRBF]: .NET Remoting: Binary Format Data Structure

This topic lists the Errata found in [MS-NRBF] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

Errata below are for Protocol Document Version V12.0 – 2019/03/13.

Errata Published*	Description
2019/10/28	In Section 3.0, Structure Examples, in the logical Request message for dotNET_Framework 1.1, changed the BinaryMethodCall value from: BinaryMethodCall: RecordTypeEnum: BinaryMethodCall (0x21) MessageEnum: 00000014 Changed to: BinaryMethodCall: RecordTypeEnum: BinaryMethodCall (0x15) MessageEnum: 00000014

*Date format: YYYY/MM/DD

[MS-NRPC]: Netlogon Remote Protocol

This topic lists the Errata found in [MS-NRPC] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

September 23, 2019 - Download

Errata below are for Protocol Document Version V36.0 – 2019/09/23.

Errata Published*	Description
2020/08/17	In section 3.1.1 Abstract Data Model, added Netlogon server variable VulnerableChannelAllowList. Changed from: Implementations SHOULD<68> persistently store and retrieve the SealSecureChannel variable. Changed to: Implementations SHOULD<68> persistently store and retrieve the SealSecureChannel variable. The Netlogon server variable is as follows: VulnerableChannelAllowList: A setting expressed in Security Descriptor Description Language (SDDL) ([MS-DTYP] section 2.5.1) of Netlogon client allowed to not use secure bindings see Section 3.1.4.6.<69> <69> Section 3.1.1: VulnerableChannelAllowList is not supported by Windows NT, Windows 2000, Windows Server 2003, and Windows Server 2008. In section 3.1.4.1 Session-Key Negotiation, added session-key failure scenario in step 7. Changed from: 6. … If the comparison fails, the server MUST fail session-key negotiation without further processing of the following steps. 7. The server computes its server Netlogon credential by using the server challenge as input to the credential computation algorithm, as specified in section 3.1.4.4. … Changed to: 6. … If the comparison fails, the server MUST fail session-key negotiation without further

Errata Published*	Description
	processing of the following steps. 7. If none of the first 5 bytes of the client challenge is unique, the server MUST fail session-key negotiation without further processing of the following steps.<70> 8. The server computes its server Netlogon credential by using the server challenge as input to the credential computation algorithm, as specified in section 3.1.4.4. … <70> Section 3.1.4.1: Windows NT, Windows 2000, Windows Server 2003, and Windows Server 2008 allow the call to succeed. In section 3.1.4.2 Netlogon Negotiable Options, added in product note <73> option Y that Windows NT 4.0 SP4 does not support Secure RPC and secure bind. Changed from: Y is not supported in Windows NT prior to Windows NT 4.0 operating system Service Pack 2 (SP2). Changed to: Y is not supported in Windows NT prior to Windows NT 4.0 operating system Service Pack 2 (SP2). Windows NT 4.0 operating system Service Pack 4 (SP4) does not support Secure RPC and does not perform a secure bind. In section 3.1.4.6 Calling Methods Requiring Session-Key Establishment, Added product note for server security enforcement in earlier versions. Moved product note after MUST in step 1 to section 3.4.1.2 product note <73>. Added steps for server processing of secure bind and session-key. Changed from: The client follows this sequence of steps. The client SHOULD<72> bind to the RPC server using TCP/IP. The client and server SHOULD<73> utilize a secure bind. If a secure bind is used, the client instructs the RPC runtime to use the Netlogon SSP ([MS-RPCE] section 2.2.1.1.7) for privacy/integrity of the RPC messages. If the SealSecureChannel setting is TRUE, the client requests the Privacy authentication level from the RPC runtime. If the SealSecureChannel setting is FALSE, then the authentication level requested is Integrity. If the call to be made uses Netlogon authenticators, the client MUST compute the Netlogon authenticator to be passed as a parameter to the RPC method, as specified in section 3.1.4.5. The client calls the method on the server. If the RPC server denies access, the client attempts to re-establish the session key with the target server if the difference between the current time and value of ServerSessionInfo.LastAuthenticationTry (indexed by the name of the target server) is greater than 45 seconds. The server MUST verify the authenticator, if used, and compute the return authenticator, as specified in section 3.1.4.5. The client MUST validate the returned authenticator, if used. The client MAY unbind from the server, but it SHOULD<74> reuse the binding for multiple RPC calls. <72> Section 3.1.4.6: For Windows, the client binds to the RPC server using TCP (except for Windows NT, in which the client binds to the RPC server using the named pipe "\PIPE\NETLOGON",). If RPC returns an error indicating that the protocol sequence is not

Errata Published*	Description
	supported, the client binds to the RPC server using named pipes. <73> Section 3.1.4.6: Windows NT 4.0 operating system Service Pack 4 (SP4) does not support Secure RPC and does not perform a secure bind. Changed to: The client and server follow this sequence of steps.<74> The client SHOULD<75> bind to the RPC server using TCP/IP. The client and server MUST utilize a secure bind. If a secure bind is used, the client instructs the RPC runtime to use the Netlogon SSP ([MS-RPCE] section 2.2.1.1.7) for privacy/integrity of the RPC messages. If the SealSecureChannel setting is TRUE, the client requests the Privacy authentication level from the RPC runtime. If the SealSecureChannel setting is FALSE, then the authentication level requested is Integrity. If the call to be made uses Netlogon authenticators, the client MUST compute the Netlogon authenticator to be passed as a parameter to the RPC method, as specified in section 3.1.4.5. The client calls the method on the server. If the RPC server denies access, the client attempts to re-establish the session key with the target server if the difference between the current time and value of ServerSessionInfo.LastAuthenticationTry (indexed by the name of the target server) is greater than 45 seconds. If secure bind is not used, the server MUST deny the request unless client is in the VulnerableChannelAllowList setting.<76> The server MUST verify the authenticator, if used, and compute the return authenticator, as specified in section 3.1.4.5. If none of the first 5 bytes of the ClientStoredCredential computation result (step 1, section 3.1.4.5) is unique, the server MUST fail session-key negotiation without further processing of the following steps.<77> The client MUST validate the returned authenticator, if used. The client MAY unbind from the server, but it SHOULD<78> reuse the binding for multiple RPC calls. <74> Section 3.1.4.6: Whenever a Windows 7 client or later creates a secure channel with a Windows Server 2008 server or later, the server will enforce that clients are using RPC Integrity and Confidentiality to secure the connection. <75> Section 3.1.4.6: For Windows, the client binds to the RPC server using TCP (except for Windows NT, in which the client binds to the RPC server using the named pipe "\PIPE\NETLOGON"). If RPC returns an error indicating that the protocol sequence is not supported, the client binds to the RPC server using named pipes. <76> Section 3.1.4.6: Windows NT, Windows 2000, Windows Server 2003, and Windows Server 2008 allow the call to succeed. <77> Section 3.1.4.6: Windows NT, Windows 2000, Windows Server 2003, and Windows Server 2008 allow the call to succeed.
2020/05/11	In Section 2.2.1.2.1, DOMAIN_CONTROLLER_INFOW, added 'T' bit flag to indicate the DC supports Kerberos key list requests. Changed from:

Errata Published*	Description

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
O	N	M	0	0	0	0	0	0	0	0	0	0	0	0	S	R	Q	P	L	K	J	I	H	G	F	E	D	C	B	0	A

Value	Description
...	...
R	The DC has a functional level of DS_BEHAVIOR_WIN2012R2 or later.
S	The DC has a functional level of DS_BEHAVIOR_WIN2016 or later.

Changed to:

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
O	N	M	0	0	0	0	0	0	0	0	0	0	0	T	S	R	Q	P	L	K	J	I	H	G	F	E	D	C	B	0	A

Value	Description
...	...
R	The DC has a functional level of DS_BEHAVIOR_WIN2012R2 or later.
S	The DC has a functional level of DS_BEHAVIOR_WIN2016 or later.
T	The DC supports **key list requests**, as specified in [MS-KILE] section 2.2.11. If this bit is set, bit S and bit E must also be set.

*Date format: YYYY/MM/DD

[MS-NSPI]: Name Service Provider Interface (NSPI) Protocol

This topic lists the Errata found in [MS-NSPI] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-OAPX]: OAuth 2.0 Protocol Extensions

This topic lists the Errata found in [MS-OAPX] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

June 1, 2017 - Download

September 15, 2017 - Download

September 12, 2018 - Download

[MS-OAPXBC]: OAuth 2.0 Protocol Extensions for Broker Clients

This topic lists the Errata found in [MS-OAPXBC] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

September 26, 2016 - Download

June 1, 2017 - Download

September 15, 2017 - Download

September 12, 2018 - Download

[MS-OCSPA]: Microsoft OCSP Administration Protocol

This topic lists the Errata found in [MS-OCSPA] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

Errata below are for Protocol Document Version V11.0 – 2018/09/12.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td>2020/07/06</td><td>In Section 3.2.4.1.3 GetCAConfigInformation (Opnum 5), added 'ReminderDuration' and 'RefreshTimeout' properties along with associated processing rules to first and second tables in the indicated section, respectively.

Changed from:

<table>
<tr><th>Property name</th><th>Processing rule</th></tr>
<tr><td>SigningCertificate</td><td>The vt member of the VARIANT MUST be set to VT_ARRAY|VT_UI1, and the pArray member MUST reference a single dimension safearray. The number of elements of the safearray referenced by pArray MUST be equal to the length in bytes of the ASN.1 DER encoding of the signing certificate used by the responder to sign OCSP responses for this revocation configuration.</td></tr>
</table>

Changed to:

<table>
<tr><th>Property name</th><th>Processing rule</th></tr>
<tr><td>ReminderDuration</td><td>The vt member of the VARIANT MUST be set to VT_I4, and the lVal member MUST be a DWORD value denoting the percentage of the signing certificate’s lifetime, after which, if the signing certificate is not renewed, a warning event will be logged.</td></tr>
<tr><td>SigningCertificate</td><td>The vt member of the VARIANT MUST be set to VT_ARRAY|VT_UI1, and the pArray member MUST reference a single dimension safearray. The number of elements of the safearray referenced by pArray MUST be equal to the length in bytes of the ASN.1 DER encoding of the signing certificate used by the responder to sign OCSP responses for this revocation configuration.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>Changed from:

<table><tr><th>Property name</th><th>Processing rules</th></tr><tr><td>RevocationErrorCode</td><td>The vt member of the VARIANT MUST be set to VT_I4, and the lVal member MUST be the HRESULT DWORD value denoting the status of this revocation provider. A value of 0 means that the revocation provider can provide certificate revocation status for certificates issued by the certificate authority configured for the revocation configuration. See [MS-ERREF] for a list of the possible error codes.</td></tr></table>
Changed to:

<table><tr><th>Property name</th><th>Processing rules</th></tr><tr><td>RefreshTimeout</td><td>The vt member of the VARIANT MUST be set to VT_I4, and the lVal member MUST be the time-out value in milliseconds associated with refreshing the CRL information.</td></tr><tr><td>RevocationErrorCode</td><td>The vt member of the VARIANT MUST be set to VT_I4, and the lVal member MUST be the HRESULT DWORD value denoting the status of this revocation provider. A value of 0 means that the revocation provider can provide certificate revocation status for certificates issued by the certificate authority configured for the revocation configuration. See [MS-ERREF] for a list of the possible error codes.</td></tr></table></td></tr>
</table>

*Date format: YYYY/MM/DD

[MS-OIDCE]: OpenID Connect 1.0 Protocol Extensions

This topic lists the Errata found in [MS-OIDCE] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

December 1, 2017 - Download

March 16, 2018 - Download

[MS-OLEDS]: Object Linking and Embedding (OLE) Data Structures

This topic lists the Errata found in [MS-OLEDS] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

December 1, 2017 - Download

September 12, 2018 - Download

[MS-OTPCE]: One-Time Password Certificate Enrollment Protocol

This topic lists the Errata found in [MS-OTPCE] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

RSS
Atom

Errata are subject to the same terms as the Open Specifications documentation referenced.

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-PAC]: Privilege Attribute Certificate Data Structure

This topic lists the Errata found in [MS-PAC] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

Errata below are for Protocol Document Version V19.0 – 2018/09/12.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td>2020/06/08</td><td>In Section 4.1.2.2, SID Filtering and Claims Transformation, addedMember trust boundary type.

Changed from:

<table>
<tr><th>Trust boundary type</th><th>Description</th></tr>
<tr><td>WithinDomain</td><td>Within a domain, each domain controller trusts every other domain controller. Added Member trust boundary type</td></tr>
<tr><td>WithinForest</td><td>Within a forest, there are parent/child trust relationships and shortcut trust relationships between the domains in the forest. Each domain controller trusts every other domain controller within the forest.</td></tr>
</table>

Changed to:

<table>
<tr><th>Trust boundary type</th><th>Description</th></tr>
<tr><td>Member</td><td>The member trust boundary type member boundary filters SIDs that are in the AlwaysFilter group as well as anything that has the prefix of the member server.</td></tr>
<tr><td>WithinDomain</td><td>Within a domain, each domain controller trusts every other domain controller.</td></tr>
<tr><td>WithinForest</td><td>Within a forest, there are parent/child trust relationships and shortcut trust relationships between the domains in the forest. Each domain controller trusts every other domain controller within the forest.</td></tr>
</table>
</td></tr>
<tr><td>2019/09/02</td><td>In Section 3.1, Logon Authorization Information, the string format for two SIDs has been changed from:

S-1-5-397955417-626881126-188441444</td></tr>
</table>

Errata Published*	Description
	Changed to: S-1-5-21-397955417-626881126-188441444 Changed from: S-1-5-397955417-626881126-188441444-3392609 Changed to: S-1-5-21-397955417-626881126-188441444-3392609

*Date format: YYYY/MM/DD

[MS-PAR]: Print System Asynchronous Remote Protocol

This topic lists the Errata found in [MS-PAR] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

Errata below are for Protocol Document Version V15.0 – 2018/09/12.

Errata Published*	Description
2018/12/10	In Section 3.1.4.2.7, RpcAsyncInstallPrinterDriverFromPackage (Opnum 62), changed from: The print server SHOULD<10> perform the following additional validation steps: … • Validate that, if the printer driver specified by the client is a derived printer driver, either the class printer driver on which the derived printer driver depends is already installed on the print server, or a driver package containing the class printer driver is installed in the print server's driver store, or the print server can locate a driver package containing the class printer driver through some other implementation-specific mechanism;<11> otherwise, the server returns ERROR_UNKNOWN_PRINTER_DRIVER. Changed to: The print server SHOULD<10> perform the following additional validation steps: … • Validate that, if the printer driver specified by the client is a derived printer driver, either the class printer driver on which the derived printer driver depends is already installed on the print server, or a driver package containing the class printer driver is installed in the print server's driver store, or the print server can locate a driver package containing the class printer driver through some other implementation-specific mechanism;<11> otherwise, the server returns ERROR_UNKNOWN_PRINTER_DRIVER. This HRESULT error code is constructed by using the HRESULT From WIN32 Error Code Macro ([MS-ERREF] section 2.1.2) on the 16-bit Win32 value for this error ([MS-ERREF] section 2.2).

*Date format: YYYY/MM/DD

[MS-PEAP]: Protected Extensible Authentication Protocol (PEAP)

This topic lists the Errata found in [MS-PEAP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-PKAP]: Public Key Authentication Protocol

This topic lists the Errata found in the MS-PKAP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-PSRDP]: PowerShell Remote Debugging Protocol

This topic lists the Errata found in [MS-PSRDP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-PSRP]: PowerShell Remoting Protocol

This topic lists the Errata found in [MS-PSRP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

[MS-RA]: Remote Assistance Protocol

This topic lists the Errata found in [MS-RA] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-RAI]: Remote Assistance Initiation Protocol

This topic lists the Errata found in [MS-RAI] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

Errata below are for Protocol Document Version V9.0 – 2018/09/12.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td>2019/06/24</td><td>In Section 2.2.2, Remote Assistance Connection String 2, details for the URI attribute have been added for the Listener node.

Changed from:

...

3. The Transport Node has Listener child Nodes that give information about the Server IP and port. This Listener node <L> has the following attributes.

<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>P</td><td>Port: The dynamic port on which the Remote Assistance connection could happen.</td></tr><tr><td>N</td><td>Server Name: The name/IP address of the server, that is, the novice computer.</td></tr></table>
...

Changed to:

...

3. The Transport Node has Listener child Nodes that give information about the Server IP and port. This Listener node <L> has the following attributes.

<table><tr><th>Value</th><th>Meaning</th></tr></table></td></tr>
</table>

Errata Published*	Description
	P: Port: The dynamic port on which the Remote Assistance connection could happen. N: Server Name: The name/IP address of the server, that is, the novice computer. U: URI: The full URI if websocket listener is enabled. The U (URI) is used instead of the P (port) attribute. N (server name) attribute is still included. …

*Date format: YYYY/MM/DD

[MS-RDPADRV]: Remote Desktop Protocol Audio Level and Drive Letter Persistence Virtual Channel Extension

This topic lists the Errata found in [MS-RDPADRV] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 12, 2018 - Download

[MS-RDPBCGR]: Remote Desktop Protocol: Basic Connectivity and Graphics Remoting

This topic lists the Errata found in [MS-RDPBCGR] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

March 20, 2017 - Download

June 1, 2017 - Download

December 1, 2017 - Download

March 16, 2018 - Download

September 12, 2018 - Download

March 13, 2019 - Download

September 23, 2019 - Download

March 4, 2020 - Download

Errata below are for Protocol Document Version V8.0 – 2019/03/13.

Errata Published*	Description
2020/08/17	In Section 2.2.1.4.4 Server Network Data (TS_UD_SC_NET), we updated the channelIdArray description to remove inaccurate information. Changed from: channelIdArray (variable): A variable-length array of MCS channel IDs (each channel ID is a 16-bit, unsigned integer) which have been allocated (the number is given by the channelCount field). Each MCS channel ID corresponds in position to the channels requested in the Client Network Data structure. A channel value of 0 indicates that the channel was not allocated. Changed to: channelIdArray (variable): A variable-length array of MCS channel IDs (each channel ID is a 16-bit, unsigned integer) which have been allocated (the number is given by the channelCount field). Each MCS channel ID corresponds in position to the channels requested in the Client Network Data structure.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td>2020/08/17</td><td>In Section 2.2.1.3.4.1 Channel Definition Structure (CHANNEL_DEF), modified the meaning of the CHANNEL_OPTION_INITIALIZED flag to indicate that the flag is unused and MUST be ignored by the server.

Changed from:

CHANNEL_OPTION_INITIALIZED
0x80000000

Absence of this flag indicates that this channel is a placeholder and that the server MUST NOT set it up.

Changed to:

CHANNEL_OPTION_INITIALIZED
0x80000000
This flag is unused and its value MUST be ignored by the server.</td></tr>
<tr><td>2020/07/20</td><td>In Section 2.2.10.1.1.4.1.1, Logon Errors Info (TS_LOGON_ERRORS_INFO), added the LOGON_MSG_SESSION_BUSY_OPTIONS notification type.

Changed from:
ErrorNotificationType (4 bytes): A 32-bit, unsigned integer that specifies an NTSTATUS value (see [ERRTRANS] for information about translating NTSTATUS error codes to usable text strings), or one of the following values.
<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>LOGON_MSG_DISCONNECT_REFUSED
0xFFFFFFF9</td><td>The "Disconnection Refused" dialog is being displayed by Winlogon. The session identifier is specified by the ErrorNotificationData field.</td></tr></table>
Changed to:
ErrorNotificationType (4 bytes): A 32-bit, unsigned integer that specifies an NTSTATUS value (see [ERRTRANS] for information about translating NTSTATUS error codes to usable text strings), or one of the following values.
<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>LOGON_MSG_SESSION_BUSY_OPTIONS
0xFFFFFFF8</td><td>The "Session is Busy" dialog is being displayed by Winlogon. The session identifier is specified by the ErrorNotificationData field.</td></tr><tr><td>LOGON_MSG_DISCONNECT_REFUSED
0xFFFFFFF9</td><td>The "Disconnection Refused" dialog is being displayed by Winlogon. The session identifier is specified by the ErrorNotificationData field.</td></tr></table></td></tr>
<tr><td>2020/07/06</td><td>In Section 2.2.17.4, RDSTLS Authentication Response PDU, revised ResultCode description to match definition – 16-bit to 32-bit unsigned integer.

Changed from:</td></tr>
</table>

Errata Published*	Description

	ResultCode (4 bytes): A 16-bit unsigned integer that specifies the user authentication result.
	
	Changed to:
	
	ResultCode (4 bytes): A 32-bit unsigned integer that specifies the user authentication result.
	
	In Section 4.1.3, Client MCS Connect Initial PDU with GCC Conference Create Request, revised comment for TS_UD_CS_CORE::connectionType.
	
	Changed from:
	
	`00 -> TS_UD_CS_CORE::connectionType = 0 (not used as RNS UD CS VALID CONNECTION TYPE not set)`
	`00 -> TS_UD_CS_CORE::pad1octet`
	
	Changed to:
	
	`00 -> TS UD CS CORE::connectionType = 0 (ignored as RNS UD CS VALID CONNECTION TYPE not set)`
	`00 -> TS_UD_CS_CORE::pad1octet`
	
	In Section 4.1.14, Client Synchronize PDU, revised annotated dump to match revision made in section 2.2.17.4.
	
	Changed from:
	
	`00 01 ->TS SYNCHRONIZE PDU::messageType = SYNCMSGTYPE SYNC (1)`
	`ea 03 ->TS_SYNCHRONIZE_PDU::targetUser = 0x03ea`
	
	Changed to:
	
	`01 00 ->TS SYNCHRONIZE PDU::messageType = SYNCMSGTYPE SYNC (1)`
	`ea 03 ->TS_SYNCHRONIZE_PDU::targetUser = 0x03ea`

2020/07/06	In Section 4.1.12, Server Demand Active PDU, revised annotated dump – pad1octet to drawingFlags & pad2octetsB to orderSupportExFlags.
	
	Changed from:
	
	`00 -> TS_BITMAP_CAPABILITYSET::highColorFlags = 0`
	`00 -> TS_BITMAP_CAPABILITYSET::pad1octet`
	`01 00 -> TS_BITMAP_CAPABILITYSET::multipleRectangleSupport = TRUE`

Errata Published*	Description
	00 00 -> TS BITMAP CAPABILITYSET::pad2octetsB Order Capability Set (88 bytes) Changed to: 00 -> TS_BITMAP_CAPABILITYSET::highColorFlags = 0 00 -> TS BITMAP CAPABILITYSET::drawingFlags 01 00 -> TS_BITMAP_CAPABILITYSET::multipleRectangleSupport = TRUE 00 00 -> TS_BITMAP_CAPABILITYSET::pad2octetsB Order Capability Set (88 bytes) In that same section, changed from: a1 06 -> TS_ORDER_CAPABILITYSET::textFlags = 0x06a1 00 00 -> TS ORDER CAPABILITYSET::pad2octetsB 40 42 0f 00 -> TS_ORDER_CAPABILITYSET::pad4octetsB 40 42 0f 00 -> TS_ORDER_CAPABILITYSET::desktopSaveSize = 0xf4240 = 1000000 Changed to: a1 06 -> TS_ORDER_CAPABILITYSET::textFlags = 0x06a1 00 00 -> TS ORDER CAPABILITYSET::orderSupportExFlags 40 42 0f 00 -> TS_ORDER_CAPABILITYSET::pad4octetsB 40 42 0f 00 -> TS_ORDER_CAPABILITYSET::desktopSaveSize = 0xf4240 = 1000000 In Section 4.1.13, Client Confirm Active PDU, revised annotated dump – pad2octets to pad2Octets & pad1octet to drawingFlags & pad2octetsB to orderSupportExFlags. Changed from: 4d 53 54 53 43 00 -> TS CONFIRM ACTIVE PDU::sourceDescriptor = "MSTSC" 12 00 -> TS_CONFIRM_ACTIVE_PDU::numberCapabilities = 18 00 00 -> TS_CONFIRM_ACTIVE_PDU::pad2octets General Capability Set (24 bytes) Changed to: 4d 53 54 53 43 00 -> TS_CONFIRM_ACTIVE_PDU::sourceDescriptor = "MSTSC"

Errata Published*	Description
	12 00 -> TS CONFIRM ACTIVE PDU::numberCapabilities = 18 00 00 -> TS_CONFIRM_ACTIVE_PDU::pad2Octets General Capability Set (24 bytes) In that same section, changed from: 00 -> TS BITMAP CAPABILITYSET::highColorFlags = 0 00 -> TS_BITMAP_CAPABILITYSET::pad1octet 01 00 -> TS_BITMAP_CAPABILITYSET::multipleRectangleSupport = TRUE 00 00 -> TS_BITMAP_CAPABILITYSET::pad2octetsB Order Capability Set (88 bytes) Changed to: 00 -> TS BITMAP CAPABILITYSET::highColorFlags = 0 00 -> TS_BITMAP_CAPABILITYSET::drawingFlags 01 00 -> TS_BITMAP_CAPABILITYSET::multipleRectangleSupport = TRUE 00 00 -> TS_BITMAP_CAPABILITYSET::pad2octetsB Order Capability Set (88 bytes) Changed from: TS_TEXTFLAGS_ALLOWDELTAXSIM | TS TEXTFLAGS CHECKFONTASPECT 00 00 -> TS_ORDER_CAPABILITYSET::pad2octetsB 00 00 00 00 -> TS_ORDER_CAPABILITYSET::pad4octetsB 00 84 03 00 -> TS_ORDER_CAPABILITYSET::desktopSaveSize = 0x38400 = 230400 Changed to: TS_TEXTFLAGS_CHECKFONTASPECT 00 00 -> TS_ORDER_CAPABILITYSET::orderSupportExFlags 00 00 00 00 -> TS_ORDER_CAPABILITYSET::pad4octetsB 00 84 03 00 -> TS_ORDER_CAPABILITYSET::desktopSaveSize = 0x38400 = 230400 In Section 4.1.18, Client Font List PDU, revised annotated dump – numberEntries to numberFonts & totalNumEntries to totalNumFonts & added a couple of lines. Changed from: 00 00 -> TS_SHAREDATAHEADER::compressedLength = 0

Errata Published*	Description
	00 00 -> TS FONT LIST PDU::numberEntries = 0 00 00 -> TS_FONT_LIST_PDU::totalNumEntries = 0 03 00 -> TS FONT LIST PDU::listFlags = 0x0003 = 0x0002 | 0x0001 = FONTLIST_LAST | FONTLIST_FIRST 32 00 -> TS_FONT_LIST_PDU::entrySize = 0x0032 = 50 bytes Changed to: 00 00 -> TS_SHAREDATAHEADER::compressedLength = 0 00 00 -> TS_FONT_LIST_PDU::numberFonts = 0 00 00 -> TS FONT LIST PDU::totalNumFonts = 0 03 00 -> TS FONT LIST PDU::listFlags = 0x0003 0x0003 = 0x0002 | 0x0001 FONTLIST_LAST | FONTLIST_FIRST 32 00 -> TS_FONT_LIST_PDU::entrySize = 0x0032 = 50 bytes In Section 4.4, Annotated Server-to-Client Virtual Channel PUD, revised HEADER to HEADER1. Changed from: 01 00 -> TS_SECURITY_HEADER::flagsHi - ignored as flags field does not contain SEC FLAGSHI VALID (0x8000) 47 bd eb cb 29 51 ae 0a -> TS SECURITY HEADER::dataSignature f6 07 33 ce fc a5 f7 09 de 67 4e a3 2a 2c 38 29 -> Encrypted static virtual channel data Changed to: 01 00 -> TS_SECURITY_HEADER::flagsHi - ignored as flags field does not contain SEC FLAGSHI VALID (0x8000) 47 bd eb cb 29 51 ae 0a -> TS_SECURITY_HEADER1::dataSignature f6 07 33 ce fc a5 f7 09 de 67 4e a3 2a 2c 38 29 -> Encrypted static virtual channel data In Section 4.5, Annotated Standard Security Server Redirection PDU, revised HEADER to HEADER1. Changed from: = SEC_SECURE_CHECKSUM | SEC_REDIRECTION_PKT 00 00 -> TS SECURITY HEADER::flagsHi - ignored as flags field does not contain RDP_SEC_FLAGSHI_VALID (0x8000)

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
<pre>
     58 dd 3f e5 f3 de 80 26 -> TS SECURITY HEADER::dataSignature

     c0 d6 3f 26 0e 2c b5 93 dd 26 d5 4b 84 a1 1d 2a
</pre>
Changed to:
<pre>
     = SEC_SECURE_CHECKSUM | SEC_REDIRECTION_PKT

     00 00 -> TS_SECURITY_HEADER::flagsHi - ignored as flags field does not contain
    RDP_SEC_FLAGSHI_VALID (0x8000)
     58 dd 3f e5 f3 de 80 26 -> TS_SECURITY_HEADER1::dataSignature

     c0 d6 3f 26 0e 2c b5 93 dd 26 d5 4b 84 a1 1d 2a
</pre>
</td></tr>
<tr><td>2020/07/06</td><td>
In Section 2.2.4.1, Server Auto-Reconnect Status PDU, revised Client Auto-Reconnection Packet to Client Auto-Reconnect Packet.

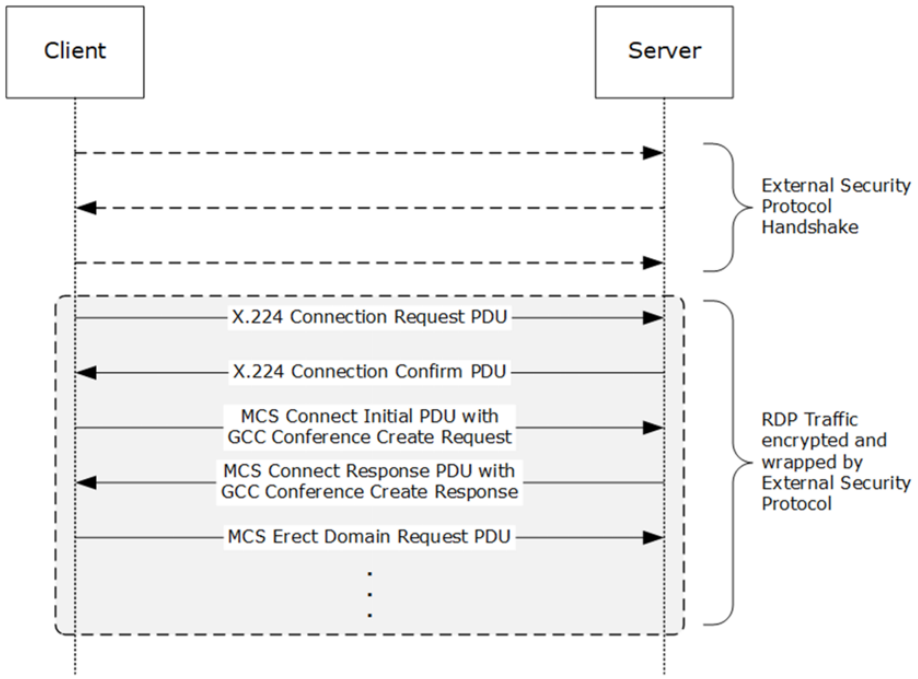
Changed from:
The Auto-Reconnect Status PDU is sent by the server to the client to indicate that automatic reconnection using the Client Auto-Reconnection Packet (section 2.2.4.3), sent as part of the extended information of the Client Info PDU (section 2.2.1.11.1), has failed.

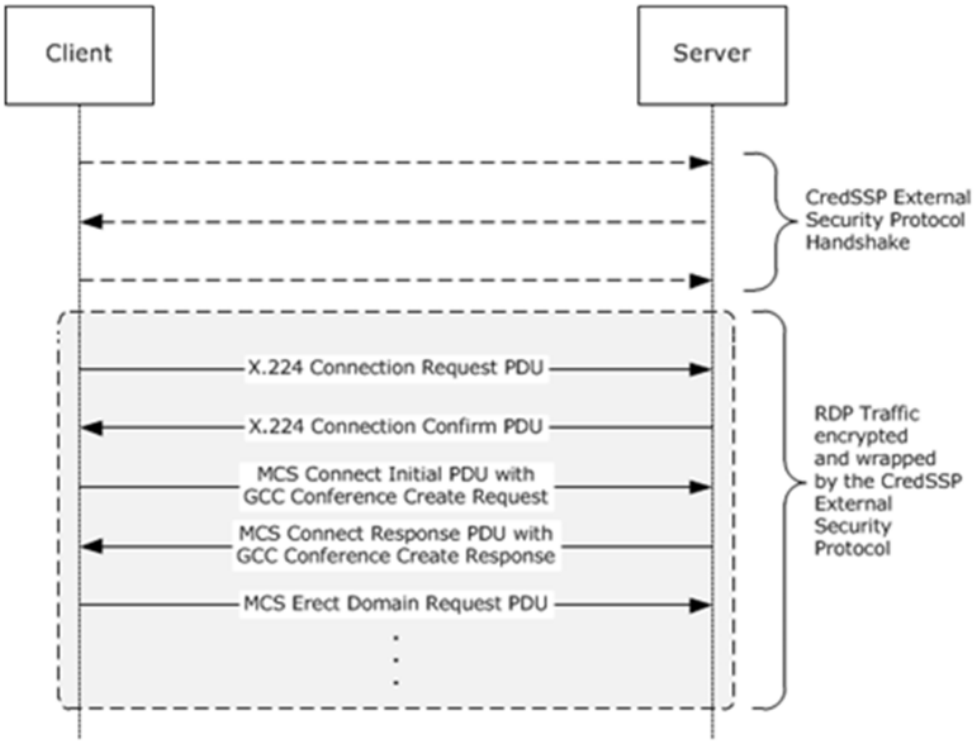
Changed to:
The Auto-Reconnect Status PDU is sent by the server to the client to indicate that automatic reconnection using the Client Auto-Reconnect Packet (section 2.2.4.3), sent as part of the extended information of the Client Info PDU (section 2.2.1.11.1), has failed.

In Section 2.2.5.1.1, Set Error Info PDU Data, revised description for ERRINFO_CONTROLPDUSEQUENCE.

Changed from:
<table>
<tr><td>ERRINFO_CONTROLPDUSEQUENCE
0x000010CD</td><td>An out-of-sequence Slow-Path Non-Data PDU (section 2.2.8.1.1.1.1) has been received.</td></tr>
</table>
Changed to:
<table>
<tr><td>ERRINFO_CONTROLPDUSEQUENCE
0x000010CD</td><td>An out-of-sequence Server Demand Active PDU (section 2.2.1.13.1), Client Confirm Active PDU (section 2.2.1.13.2), Server Deactivate All PDU (section 2.2.3.1) or Enhanced Security Server Redirection PDU (section 2.2.13.3.1) has been received.</td></tr>
</table>
In Section 2.2.8.1.1.3.1.1, Slow-Path Input Event, revised definition for slowPathInputData.

Changed from:
slowPathInputData (variable): TS_KEYBOARD_EVENT, TS_UNICODE_KEYBOARD_EVENT,
</td></tr>
</table>

Errata Published*	Description
	TS_POINTER_EVENT, TS_POINTERX_EVENT, or TS_SYNC_EVENT. The actual contents of the input event specified by the messageType field (sections 2.2.8.1.1.3.1.1.1 through 2.2.8.1.1.3.1.1.6). Changed to: slowPathInputData (variable): TS_KEYBOARD_EVENT, TS_UNICODE_KEYBOARD_EVENT, TS_POINTER_EVENT, TS_POINTERX_EVENT, TS_SYNC_EVENT, or TS_UNUSED_EVENT. The actual contents of the input event specified by the messageType field (sections 2.2.8.1.1.3.1.1.1 through 2.2.8.1.1.3.1.1.6).
2020/07/06	In Section 5.4.2.2, Direct Approach, revised the figure. Changed from:  Changed to:

Errata Published*	Description
	 In that same section, changed from: When using the Direct Approach, no negotiation of the security protocol takes place. The client and server are hard-coded to use the Credential Security Support Provider (CredSSP) Protocol (section 5.4.5) when a connection is initiated. Once the security protocol handshake has completed successfully, the RDP Connection Sequence begins, starting with (a) the X.224 messages which form the Connection Initiation phase (section 1.3.1.1); or (b) the Early User Authorization Result PDU (section 2.2.10.2) followed by the X.224 messages. From this point all RDP traffic is encrypted using the CredSSP External Security Protocol. The RDP Negotiation Request (section 2.2.1.1.1) will still be appended to the X.224 Connection Request PDU (section 2.2.1.1) and the requested protocol list will contain the identifier of the CredSSP protocol (section 2.2.1.1.1). If this is not the case, the server will append an RDP Negotiation Failure (section 2.2.1.2.2) to the X.224 Connection Confirm PDU (section 2.2.1.2) with a failure code of INCONSISTENT_FLAGS (0x04). Similarly, the server will indicate that the hard-coded security protocol is the selected protocol in the RDP Negotiation Response (section 2.2.1.2.1) which is appended to the X.224 Connection Confirm PDU. Changed to: When using the Direct Approach, no negotiation of the security protocol takes place. The client and server are hard-coded to use the Credential Security Support Provider (CredSSP) Protocol (section 5.4.5) when a connection is initiated. The Early User Authorization Result PDU (section 2.2.10.2) is not supported in the Direct Approach. Once the security protocol handshake has completed successfully, the RDP Connection Sequence begins, starting with the X.224 messages which form the Connection Initiation phase (section 1.3.1.1). From this point all RDP traffic is encrypted using the CredSSP External Security Protocol.

Errata Published*	Description
	The RDP Negotiation Request (section 2.2.1.1.1) MUST be appended to the X.224 Connection Request PDU (section 2.2.1.1) and the requested protocol list MUST contain the PROTOCOL_HYBRID (0x00000002) flag identifying the CredSSP protocol (section 2.2.1.1.1). If this is not the case, the server will append an RDP Negotiation Failure (section 2.2.1.2.2) to the X.224 Connection Confirm PDU (section 2.2.1.2) with a failure code of INCONSISTENT_FLAGS (0x04). Similarly, the server MUST indicate that CredSSP is the selected protocol in the RDP Negotiation Response (section 2.2.1.2.1) which is appended to the X.224 Connection Confirm PDU.

*Date format: YYYY/MM/DD

[MS-RDPEA]: Remote Desktop Protocol: Audio Output Virtual Channel Extension

This topic lists the Errata found in [MS-RDPEA] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-RDPEAR]: Remote Desktop Protocol Authentication Redirection Virtual Channel

This topic lists the Errata found in [MS-RDPEAR] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 15, 2017 - Download

Errata below are for Protocol Document Version V4.0 – 2018/09/12.

Errata Published*	Description
2020/08/03	In Section 2.2.2.1.4, CreateApReqAuthenticator, updated key description and source. Changed from: EncryptionKey: The authenticator encryption key. Changed to: EncryptionKey: The opaque structure associated with the key that the CredSSP server uses to build the authenticator. The exact format of this structure is CredSSP client dependent, as specified in section 2.2.1.2.1. The key comes from a previous UnpackKdcReplyBody output message. In Section 2.2.2.1.5, DecryptApReply, updated key description and source. Changed from: Key: The Kerberos key needed to decrypt EncryptedReply. Changed to: Key: The opaque structure associated with the key that the CredSSP server uses to decrypt EncryptedReply. The exact format of this structure is CredSSP client dependent, as specified in section 2.2.1.2.1. The key comes from a previous UnpackKdcReplyBody output message. In Section 2.2.2.1.6, UnpackKdcReplyBody, updated key description and source. Changed from: Key: The decryption key. Changed to:

Errata Published*	Description
	Key: The opaque structure associated with the decryption key that the CredSSP server uses. The exact format of this structure is CredSSP client dependent, as specified in section 2.2.1.2.1. The key comes from a previous UnpackKdcReplyBody output message or the CredSSP client.
2020/07/20	In Sections 2.2.1.1, RemoteGuardCallId Enumeration and 6.2 Appendix A.2: Kerberos.IDL, replaced FAST related names with Reserved names for FAST related structures that do not appear on the wire. Changed from: RemoteCallKerbBuildTicketArmorKey RemoteCallKerbBuildExplicitArmorKey RemoteCallKerbVerifyFastArmoredTgsReply RemoteCallKerbVerifyEncryptedChallengePaData RemoteCallKerbBuildFastArmoredKdcRequest RemoteCallKerbDecryptFastArmoredKerbError RemoteCallKerbDecryptFastArmoredAsReply Changed to: Reserved1 Reserved2 Reserved3 Reserved4 Reserved5 Reserved6 Reserved7 In Sections 2.2.1.2.2, KerbCredIsoRemoteInput and 2.2.1.2.3, KerbCredIsoRemoteOutput, removed the following FAST related structures that do not appear on the wire. RemoteCallKerbBuildTicketArmorKey RemoteCallKerbBuildExplicitArmorKey RemoteCallKerbVerifyFastArmoredTgsReply RemoteCallKerbVerifyEncryptedChallengePaData RemoteCallKerbBuildFastArmoredKdcRequest RemoteCallKerbDecryptFastArmoredKerbError RemoteCallKerbDecryptFastArmoredAsReply In Sections 6.2, Appendix A.2: Kerberos.IDL, removed the following FAST related structures that do not appear on the wire for both input and output structures. RemoteCallKerbBuildTicketArmorKey RemoteCallKerbBuildExplicitArmorKey RemoteCallKerbVerifyFastArmoredTgsReply RemoteCallKerbVerifyEncryptedChallengePaData RemoteCallKerbBuildFastArmoredKdcRequest RemoteCallKerbDecryptFastArmoredKerbError RemoteCallKerbDecryptFastArmoredAsReply Removed the following sections: 2.2.2.1.13 BuildTicketArmorKey 2.2.2.1.14 BuildExplicitArmorKey

Errata Published*	Description
	2.2.2.1.15 VerifyFastArmoredTgsReply 2.2.2.1.16 VerifyEncryptedChallengePaData 2.2.2.1.17 BuildFastArmoredKdcRequest 2.2.2.1.18 DecryptFastArmoredKerbError 2.2.2.1.19 DecryptFastArmoredAsReply 3.1.5.13 RemoteCallKerbBuildTicketArmorKey 3.1.5.14 RemoteCallKerbBuildExplicitArmorKey 3.1.5.15 RemoteCallKerbVerifyFastArmoredTgsReply 3.1.5.16 RemoteCallKerbVerifyEncryptedChallengePaData 3.1.5.17 RemoteCallKerbBuildFastArmoredKdcRequest 3.1.5.18 RemoteCallKerbDecryptFastArmoredKerbError3.1.5.19 RemoteCallKerbDecryptFastArmoredAsReply
2020/06/08	In Section 2.2, Message Syntax, specified the buffer byteconfiguration Changed from: buffer: The opaque (at this layer) security package callbuffer. This buffer is to be processed by the security package described by thepackageName field. Changed to: buffer: The opaque (at this layer) security package callbuffer. This buffer is to be processed by the security package described by thepackageName field. The buffer has a 16-byte header with the first 2 bytes setto 0x1 (unsigned). The other 14 bytes are set to 0. In Section 2.2.2, Package-Specific Messages, added sectionreference to Type Serialization Version 1 in MS-RPCE. Changed from: All package-specific messages are formatted by using theDistributed Computing Environment (DCE) data representation as specified in[C706], and as exposed by the type marshaling support in Remote Procedure Call(RPC) [MS-RPCE]. Changed to: All package-specific messages are formatted byusing the Distributed Computing Environment (DCE) data representation asspecified in [C706], and as exposed by the type marshaling support in RemoteProcedure Call (RPC), as specified in Type Serialization Version 1, [MS-RPCE]section 2.2.6.

*Date format: YYYY/MM/DD

[MS-RDPECLIP]: Remote Desktop Protocol: Clipboard Virtual Channel Extension

This topic lists the Errata found in [MS-RDPECLIP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

September 12, 2018 - Download

[MS-RDPECAM]: Remote Desktop Protocol: Video Capture Virtual Channel Extension

This topic lists the Errata found in [MS-RDPECAM] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

Errata below are for Protocol Document Version V1.0 – 2018/09/12.

Errata Published*	Description
2019/02/19	In Section 4.6.2, Property List Response, an annotated dump of a Property List Response (section 2.2.3.17) has been added. Added: The following is an annotated dump of a Property List Response (section 2.2.3.17). 00000000 02 15 01 02 03 00 00 00 00 fa 00 00 00 05 00 00 .... 00000010 00 00 00 00 00 02 02 01 00 00 00 00 ff 00 00 00 .... 00000020 01 00 00 00 80 00 00 00 .... 02->SHARED MSG HEADER::Version = 2 15->SHARED_MSG_HEADER::MessageId = PropertyListResponse(21) 01->PropertyDescription[0]::PropertySet = CameraControl(1) 02->PropertyDescription[0]::PropertyId = Focus(2) 03->PropertyDescription[0]::Capabilities = Manual and Auto(1 + 2) 00 00 00 00->PropertyDescription[0]::MinValue = 0 fa 00 00 00->PropertyDescription[0]::MaxValue = 250 05 00 00 00->PropertyDescription[0]::Step = 5 00 00 00 00->PropertyDescription[0]::DefaultValue = 0 02->PropertyDescription[1]::PropertySet = VideoProcAmp(2) 02->PropertyDescription[1]::PropertyId = Brightness(2) 01->PropertyDescription[1]::Capabilities = Manual(1) 00 00 00 00->PropertyDescription[1]::MinValue = 0 ff 00 00 00->PropertyDescription[1]::MaxValue = 255 01 00 00 00->PropertyDescription[1]::Step = 1 80 00 00 00->PropertyDescription[1]::DefaultValue = 128
2019/02/19	In Section 2.2.1, Shared Message Header (SHARED_MSG_HEADER), updated values to hexadecimal format for consistency in the MessageId field table. Changed from: ... MessageId (1 byte): An 8-bit unsigned integer that specifies the type of the message. Value / Meaning SuccessResponse 1 / A Success Response (section 2.2.3.1) message.

Errata Published*	Description	
	ErrorResponse 2	An Error Response (section 2.2.3.2) message.
	SelectVersionRequest 3	A Select Version Request (section 2.2.2.1) message.
	SelectVersionResponse 4	A Select Version Response (section 2.2.2.2) message.
	DeviceAddedNotification 5	A Device Added Notification (section 2.2.2.3) message.
	DeviceRemovedNotification 6	A Device Removed Notification (section 2.2.2.4) message.
	ActivateDeviceRequest 7	An Activate Device Request (section 2.2.3.3) message.
	DeactivateDeviceRequest 8	A Deactivate Device Request (section 2.2.3.4) message.
	StreamListRequest 9	A Stream List Request (section 2.2.3.5) message.
	StreamListResponse 10	A Stream List Response (section 2.2.3.6) message.
	MediaTypeListRequest 11	A Media Type List Request (section 2.2.3.7) message.
	MediaTypeListResponse 12	A Media Type List Response (section 2.2.3.8) message.
	CurrentMediaTypeRequest 13	A Current Media Type Request (section 2.2.3.9) message.
	CurrentMediaTypeResponse 14	A Current Media Type Response (section 2.2.3.10) message.
	StartStreamsRequest 15	A Start Streams Request (section 2.2.3.11) message.
	StopStreamsRequest 16	A Stop Streams Request (section 2.2.3.12) message.
	SampleRequest 17	A Sample Request (section 2.2.3.13) message.
	SampleResponse 18	A Sample Response (section 2.2.3.14) message.
	SampleErrorResponse 19	A Sample Error Response (section 2.2.3.15) message.
	PropertyListRequest 20	A Property List Request (section 2.2.3.16) message. This message is supported only by version 2 of the protocol.
	PropertyListResponse 21	A Property List Response (section 2.2.3.17) message. This message is supported only by version 2 of the

Errata Published*	Description

	protocol.
PropertyValueRequest 22	A Property Value Request (section 2.2.3.18) message. This message is supported only by version 2 of the protocol.
PropertyValueResponse 23	A Property Value Response (section 2.2.3.19) message. This message is supported only by version 2 of the protocol.
SetPropertyValueRequest 24	A Set Property Value Request (section 2.2.3.20) message. This message is supported only by version 2 of the protocol.

Changed to:

...

MessageId (1 byte): An 8-bit unsigned integer that specifies the type of the message.

Value	Meaning
SuccessResponse 0x01	A Success Response (section 2.2.3.1) message.
ErrorResponse 0x02	An Error Response (section 2.2.3.2) message.
SelectVersionRequest 0x03	A Select Version Request (section 2.2.2.1) message.
SelectVersionResponse 0x04	A Select Version Response (section 2.2.2.2) message.
DeviceAddedNotification 0x05	A Device Added Notification (section 2.2.2.3) message.
DeviceRemovedNotification 0x06	A Device Removed Notification (section 2.2.2.4) message.
ActivateDeviceRequest 0x07	An Activate Device Request (section 2.2.3.3) message.
DeactivateDeviceRequest 0x08	A Deactivate Device Request (section 2.2.3.4) message.
StreamListRequest 0x09	A Stream List Request (section 2.2.3.5) message.
StreamListResponse 0x0A	A Stream List Response (section 2.2.3.6) message.
MediaTypeListRequest 0x0B	A Media Type List Request (section 2.2.3.7) message.
MediaTypeListResponse 0x0C	A Media Type List Response (section 2.2.3.8) message.
CurrentMediaTypeRequest 0x0D	A Current Media Type Request

Errata Published*	Description

	(section 2.2.3.9) message.
CurrentMediaTypeResponse 0x0E	A Current Media Type Response (section 2.2.3.10) message.
StartStreamsRequest 0x0F	A Start Streams Request (section 2.2.3.11) message.
StopStreamsRequest 0x10	A Stop Streams Request (section 2.2.3.12) message.
SampleRequest 0x11	A Sample Request (section 2.2.3.13) message.
SampleResponse 0x12	A Sample Response (section 2.2.3.14) message.
SampleErrorResponse 0x13	A Sample Error Response (section 2.2.3.15) message.
PropertyListRequest 0x14	A Property List Request (section 2.2.3.16) message. This message is supported only by version 2 of the protocol.
PropertyListResponse 0x15	A Property List Response (section 2.2.3.17) message. This message is supported only by version 2 of the protocol.
PropertyValueRequest 0x16	A Property Value Request (section 2.2.3.18) message. This message is supported only by version 2 of the protocol.
PropertyValueResponse 0x17	A Property Value Response (section 2.2.3.19) message. This message is supported only by version 2 of the protocol.
SetPropertyValueRequest 0x18	A Set Property Value Request (section 2.2.3.20) message. This message is supported only by version 2 of the protocol.

In Section 2.2.3.2, Error Response, updated values to hexadecimal format for consistency in the ErrorCode field table.

Changed from:

...

ErrorCode (4 bytes): A 32-bit unsigned integer containing an error code.

Value	Meaning
UnexpectedError 1	An unexpected error occurred.
InvalidMessage 2	An invalid message was received. Either the message is malformed, or

Errata Published*	Description

	the protocol version or message type is unexpected.
NotInitialized 3	The object MUST be initialized before the requested operation can be carried out. This error could be returned, for example, when attempting to communicate with a deactivated camera device.
InvalidRequest 4	The request is invalid in the current state.
InvalidStreamNumber 5	The provided stream number was invalid.
InvalidMediaType 6	The data specified for the stream format is invalid, inconsistent, or not supported.
OutOfMemory 7	The client ran out of memory.
ItemNotFound 8	The device does not support the requested property. This error code is generated only by version 2 of the protocol.
SetNotFound 9	The device does not support the requested property set. This error code is generated only by version 2 of the protocol.
OperationNotSupported 10	The requested operation is not supported. This error code is generated only by version 2 of the protocol.

Changed to:

...

ErrorCode (4 bytes): A 32-bit unsigned integer containing an error code.

Value	Meaning
UnexpectedError 0x00000001	An unexpected error occurred.
InvalidMessage 0x00000002	An invalid message was received. Either the message is malformed, or the protocol version or message type is unexpected.
NotInitialized 0x00000003	The object MUST be initialized before the requested operation can be carried out. This error could be returned, for example, when attempting to communicate with a deactivated camera device.
InvalidRequest 0x00000004	The request is invalid in the current state.
InvalidStreamNumber 0x00000005	The provided stream number was

Errata Published*	Description

	invalid.
InvalidMediaType 0x00000006	The data specified for the stream format is invalid, inconsistent, or not supported.
OutOfMemory 0x00000007	The client ran out of memory.
ItemNotFound 0x00000008	The device does not support the requested property. This error code is generated only by version 2 of the protocol.
SetNotFound 0x00000009	The device does not support the requested property set. This error code is generated only by version 2 of the protocol.
OperationNotSupported 0x0000000A	The requested operation is not supported. This error code is generated only by version 2 of the protocol.

In Section 2.2.3.6.1, STREAM_DESCRIPTION, updated the value to hexadecimal format for consistency in the StreamCategory field table.

Changed from:

...

StreamCategory (1 byte): An 8-bit unsigned integer that specifies the category of the stream.

Value	Meaning
Capture 1	Capture category streams provide a stream of compressed or uncompressed digital video.

Changed to:

...

StreamCategory (1 byte): An 8-bit unsigned integer that specifies the category of the stream.

Value	Meaning
Capture 0x01	Capture category streams provide a stream of compressed or uncompressed digital video.

In Section 2.2.3.8.1, MEDIA_TYPE_DESCRIPTION, updated values to hexadecimal format for consistency in the Format field table.

Changed from:

...

Format (1 byte): An 8-bit unsigned integer that specifies the stream codec.

Errata Published*	Description

Value	Meaning
H264 1	H.264 video as described in [ITU-H.264-201704]. Media samples contain H.264 bitstream data with start codes and interleaved sequence parameter set/picture parameter set (SPS/PPS) packets. Each sample contains one complete picture, either one field or one frame.
MJPG 2	Motion JPEG. Motion JPEG is a video compression format in which each video frame of a digital video sequence is independently compressed as a JPEG image.
YUY2 3	YUY2 video as specified in [MSDN-YUVFormats].
NV12 4	NV12 video as described in [MSDN-YUVFormats].
I420 5	I420 video. Identical to YV12 as described in [MSDN-YUVFormats] except that the order of the U and V planes is reversed.
RGB24 6	RGB, 24 bits per pixel.
RGB32 7	RGB, 32 bits per pixel.

...

Changed to:

...

Format (1 byte): An 8-bit unsigned integer that specifies the stream codec.

Value	Meaning
H264 0x01	H.264 video as described in [ITU-H.264-201704]. Media samples contain H.264 bitstream data with start codes and interleaved sequence parameter set/picture parameter set (SPS/PPS) packets. Each sample contains one complete picture, either one field or one frame.
MJPG 0x02	Motion JPEG. Motion JPEG is a video compression format in which each video frame of a digital video sequence is independently compressed as a JPEG image.
YUY2 0x03	YUY2 video as specified in [MSDN-YUVFormats].
NV12 0x04	NV12 video as described in [MSDN-YUVFormats].

Errata Published*	Description

I420 0x05	I420 video. Identical to YV12 as described in [MSDN-YUVFormats] except that the order of the U and V planes is reversed.
RGB24 0x06	RGB, 24 bits per pixel.
RGB32 0x07	RGB, 32 bits per pixel.

...

In Section 2.2.3.17.1, PROPERTY_DESCRIPTION, updated values to hexadecimal format for consistency in the PropertySet and PropertyId field tables.

Changed from:

...

PropertySet (1 byte): An 8-bit unsigned integer that specifies the property set.

Value	Meaning
CameraControl 1	This property set category controls camera device settings.
VideoProcAmp 2	This property set controls devices that can adjust the image color attributes of analog or digital signals.

PropertyId (1 byte): An 8-bit unsigned integer that contains the identifier of the property within the property set specified by the PropertySet field.

CameraControl properties:

Value	Meaning
Exposure 1	This property controls the exposure time of the device.
Focus 2	This property controls the focus setting of the device.
Pan 3	This property controls the pan setting of the device.
Roll 4	This property controls the roll setting of the device.
Tilt 5	This property controls the tilt setting of the device.
Zoom 6	This property controls the zoom setting of the device.

VideoProcAmp properties:

Value	Meaning
BacklightCompensation 1	This property controls the backlight

Errata Published*	Description

	compensation setting of the device. This value MUST be either 0 or 1. The value 0 indicates that backlight compensation is disabled. The value 1 indicates that backlight compensation is enabled.
Brightness 2	This property controls the brightness setting of the device.
Contrast 3	This property controls the contrast setting of the device.
Hue 4	This property controls the hue setting of the device.
WhiteBalance 5	This property controls the white balance setting of the device.

...

Changed to:

...

PropertySet (1 byte): An 8-bit unsigned integer that specifies the property set.

Value	Meaning
CameraControl 0x01	This property set category controls camera device settings.
VideoProcAmp 0x02	This property set controls devices that can adjust the image color attributes of analog or digital signals.

PropertyId (1 byte): An 8-bit unsigned integer that contains the identifier of the property within the property set specified by the PropertySet field.

CameraControl properties:

Value	Meaning
Exposure 0x01	This property controls the exposure time of the device.
Focus 0x02	This property controls the focus setting of the device.
Pan 0x03	This property controls the pan setting of the device.
Roll 0x04	This property controls the roll setting of the device.
Tilt 0x05	This property controls the tilt setting of the device.
Zoom 0x06	This property controls the zoom setting of the device.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>VideoProcAmp properties:
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>BacklightCompensation 0x01</td><td>This property controls the backlight compensation setting of the device. This value MUST be either 0 or 1. The value 0 indicates that backlight compensation is disabled. The value 1 indicates that backlight compensation is enabled.</td></tr>
<tr><td>Brightness 0x02</td><td>This property controls the brightness setting of the device.</td></tr>
<tr><td>Contrast 0x03</td><td>This property controls the contrast setting of the device.</td></tr>
<tr><td>Hue 0x04</td><td>This property controls the hue setting of the device.</td></tr>
<tr><td>WhiteBalance 0x05</td><td>This property controls the white balance setting of the device.</td></tr>
</table>
...

In Section 2.2.3.19.1, PROPERTY_VALUE, updated values to hexadecimal format for consistency in the Mode field table.

Changed from:
...
Mode (1 byte): An 8-bit unsigned integer that specifies how the property was set.
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>Manual 1</td><td>The value was set manually.</td></tr>
<tr><td>Auto 2</td><td>The value was set automatically.</td></tr>
</table>
...

Changed to:
...
Mode (1 byte): An 8-bit unsigned integer that specifies how the property was set.
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>Manual 0x01</td><td>The value was set manually.</td></tr>
<tr><td>Auto 0x02</td><td>The value was set automatically.</td></tr>
</table>
...</td></tr>
</table>

*Date format: YYYY/MM/DD

[MS-RDPEDISP]: Remote Desktop Protocol: Display Update Virtual Channel Extension

This topic lists the Errata found in the MS-RDPEDISP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

Errata below are for Protocol Document Version V7.0 – 2018/09/12.

Errata Published*	Description
2019/08/19	In Section 1, Introduction, changed the source of the display configuration changes from server to client. Changed from: This document specifies the Remote Desktop Protocol: Display Control Channel Extension to the Remote Desktop Protocol: Basic Connectivity and Graphics Remoting, as specified in [MS-RDPBCGR] sections 1 to 5. This control protocol is used by the server to request display configuration changes in a remote session. Display configuration changes include the addition, removal and repositioning of monitors, resolution updates, and orientation updates. Changed to: This document specifies the Remote Desktop Protocol: Display Control Channel Extension to the Remote Desktop Protocol: Basic Connectivity and Graphics Remoting, as specified in [MS-RDPBCGR] sections 1 to 5. This control protocol is used by the client to request display configuration changes in a remote session. Display configuration changes include the addition, removal and repositioning of monitors, resolution updates, and orientation updates.

*Date format: YYYY/MM/DD

[MS-RDPEDYC]: Remote Desktop Protocol: Dynamic Channel Virtual Channel Extension

This topic lists the Errata found in [MS-RDPEDYC] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

June 1, 2017 - Download

December 1, 2017 - Download

September 12, 2018 - Download

[MS-RDPEFS]: Remote Desktop Protocol: File System Virtual Channel Extension

This topic lists the Errata found in [MS-RDPEFS] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

June 1, 2017 - Download

September 15, 2017 - Download

December 1, 2017 - Download

[MS-RDPEGDI]: Remote Desktop Protocol: Graphics Device Interface (GDI) Acceleration Extensions

This topic lists the Errata found in [MS-RDPEGDI] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

[MS-RDPEGFX]: Remote Desktop Protocol: Graphics Pipeline Extension

This topic lists the Errata found in [MS-RDPEGFX] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

March 20, 2017 - Download

September 15, 2017 - Download

September 12, 2018 - Download

Errata below are for Protocol Document Version V14.0 – 2018/09/12.

Errata Published*	Description
2020/05/11	In Section 3.3.1.4, Bitmap Cache, we clarified that the slot index is one-based, not zero-based. Changed from: The Bitmap Cache ADM element is used to store bitmaps of arbitrary dimensions. Each bitmap is associated with a key and is stored in a variable-length slot (identified by a slot index). Changed to: The Bitmap Cache ADM element is used to store bitmaps of arbitrary dimensions. Each bitmap is associated with a key and is stored in a variable-length slot (identified by a one-based slot index).
2019/02/19	In Section 2.2.4.5, RFX_AVC444_BITMAP_STREAM, “YUV420 frame” in the cbAvc420EncodedBitstream1 field description has been replaced with “luma frame”. Changed from: ... cbAvc420EncodedBitstream1 (30 bits): A 30-bit unsigned integer that specifies the size, in bytes, of the luma frame present in the avc420EncodedBitstream1 field. If no YUV420 frame is present, then this field MUST be set to zero. ... Changed to: ... cbAvc420EncodedBitstream1 (30 bits): A 30-bit unsigned integer that specifies the size, in

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>bytes, of the YUV420 frame present in the avc420EncodedBitstream1 field. If no YUV420 frame is present, then this field MUST be set to zero.
...

In Section 2.2.4.6, RFX_AVC444V2_BITMAP_STREAM, “YUV420 frame” in the cbAvc420EncodedBitstream1 field description has been replaced with “luma frame”.

Changed from:
...
cbAvc420EncodedBitstream1 (30 bits): A 30-bit unsigned integer that specifies the size, in bytes, of the luma frame present in the avc420EncodedBitstream1 field. If no YUV420 frame is present, then this field MUST be set to zero.
...

Changed to:
...
cbAvc420EncodedBitstream1 (30 bits): A 30-bit unsigned integer that specifies the size, in bytes, of the YUV420 frame present in the avc420EncodedBitstream1 field. If no YUV420 frame is present, then this field MUST be set to zero.
...</td></tr>
<tr><td>2018/12/10</td><td>In Section 2.2.1.6, RDPGFX_CAPSET, the RDPGFX_CAPVERSION_106 value has been changed from 0x000A0601 to 0x000A0600 in the version field description.

Changed from:
...
version (4 bytes): A 32-bit unsigned integer that specifies the version of the capability set.

<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>...</td><td>…</td></tr><tr><td>RDPGFX_CAPVERSION_106
0x000A0601</td><td>RDPGFX_CAPSET_VERSION106
(section 2.2.3.9)</td></tr></table>
Changed to:
...
version (4 bytes): A 32-bit unsigned integer that specifies the version of the capability set.

<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>...</td><td>…</td></tr><tr><td>RDPGFX_CAPVERSION_106
0x000A0600</td><td>RDPGFX_CAPSET_VERSION106
(section 2.2.3.9)</td></tr></table>

In Section 2.2.3.9, RDPGFX_CAPSET_VERSION106, the RDPGFX_CAPVERSION_106 value has been changed from 0x000A0601 to 0x000A0600 in the version field description.</td></tr>
</table>

Errata Published*	Description
	Changed from: ... version (4 bytes): A 32-bit unsigned integer that specifies the version of the capability set. This field MUST be set to RDPGFX_CAPVERSION_106 (0x000A0601). ... Changed to: ... version (4 bytes): A 32-bit unsigned integer that specifies the version of the capability set. This field MUST be set to RDPGFX_CAPVERSION_106 (0x000A0600). ...

*Date format: YYYY/MM/DD

[MS-RDPEGT]: Remote Desktop Protocol Geometry Tracking Virtual Channel Protocol Extension

This topic lists the Errata found in [MS-RDPEGFT] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-RDPEI]: Remote Desktop Protocol: Input Virtual Channel Extension

This topic lists the Errata found in [MS-RDPEI] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-RDPELE]: Remote Desktop Protocol: Licensing Extension

This topic lists the Errata found in [MS-RDPELE] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 12, 2018 - Download

March 4, 2020 - Download

Errata below are for Protocol Document Version V15.0 – 2020/03/04.

Errata Published*	Description

2020/07/06	In Section 3.2.1.10 Encryption Keys, revised description removing undefined field.
	Changed from:
	The server uses the 128-bit licensing encryption key to encrypt and to decrypt licensing information message data obtained in Server Platform Challenge messages (section 2.2.2.4), in Client HWID (section 2.2.2.3 and 2.2.2.3.1), and in Client Platform Challenge messages (section 2.2.2.5), as specified in section 5.1.2.
	
	Changed to:
	The server uses the 128-bit licensing encryption key (section 5.1.2) to encrypt the EncryptedPlatformChallenge field in the Server Platform Challenge message (section 2.2.2.4), and decrypt the EncryptedHWID field in the Client License Information (section 2.2.2.3) and Client Platform Challenge (section 2.2.2.5) messages.
	
	In Section 4.2 CLIENT NEW LICENSE REQUEST, revised 'pBlob' to 'blobData'.
	Changed from:
	
	`41 64 6d 69 6e 69 73 74 -\`
	`72 61 74 6f 72 00 -/ ClientUserName::pBlob`
	`0x14a: ClientMachineName (2 + 2 + 7 = 0xb bytes)`
	`10 -\`
	`00 -/ ClientMachineName::wBlobType = BB CLIENT MACHINE NAME`
	
	`07 -\`
	`00 -/ ClientMachineName::wBlobLen = 7 bytes`
	
	`52 4f 44 45 4e 54 00 -> ClientMachineName::pBlob`
	
	Changed to:
	
	`41 64 6d 69 6e 69 73 74 -\`
	`72 61 74 6f 72 00 -/ ClientUserName::blobData`
	`0x14a: ClientMachineName (2 + 2 + 7 = 0xb bytes)`
	`10 -\`
	`00 -/ ClientMachineName::wBlobType = BB_CLIENT_MACHINE_NAME`

Errata Published*	Description
	07 -\ 00 -/ ClientMachineName::wBlobLen = 7 bytes 52 4f 44 45 4e 54 00 -> ClientMachineName::blobData
2020/07/06	In Section 1.3.3, Licensing PDU Flows, changed from: If the target machine is a personal terminal server, whether the client sends the license or not, the server always sends a license error message with the error code STATUS_VALID_CLIENT and the state transition code ST_NO_TRANSITION. Also, in the case that the client sends a license, the server does not validate it. The licensing protocol is complete at this point. Changed to: If the target machine is a personal terminal server, whether the client sends the license or not, the server always sends a Licensing Error Message (section 2.2.2.8) with the error code STATUS_VALID_CLIENT and the state transition code ST_NO_TRANSITION. Also, in the case that the client sends a license, the server does not validate it. The licensing protocol is complete at this point. In Section 2.2.2.1.1, Product Information (PRODUCT_INFO), changed from: The Product Information packet contains the details of the product license that is required for connecting to the terminal server. The client uses this structure together with the scope list to search for and identify an appropriate license in its license store. Depending on the outcome of the search, the client sends a Client New License Request (section 2.2.2.2), Client License Information packet (section 2.2.2.3), or license error message (section 2.2.2.8) to the server. Changed to: The Product Information packet contains the details of the product license that is required for connecting to the terminal server. The client uses this structure together with the scope list to search for and identify an appropriate license in its license store. Depending on the outcome of the search, the client sends a Client New License Request (section 2.2.2.2), Client License Information packet (section 2.2.2.3), or Licensing Error Message (section 2.2.2.8) to the server. In Section 2.2.2.8, Licensing Error Message (LICENSE_ERROR_MESSAGE), changed from: The license error message specified in [MS-RDPBCGR] section 2.2.1.12.1.3 can be used by both client and server. Changed to: The Licensing Error Message specified in [MS-RDPBCGR] section 2.2.1.12.1.3 can be used by both client and server. In Section 3.1.5.2 Sending Licensing Error Messages, changed from: Both the client and the server can send a license error message (section 2.2.2.8). Whenever an error message is sent, the message type in the licensing preamble MUST be set to ERROR_ALERT (0xFF). For the PDU, see [MS-RDPBCGR] section 2.2.1.12.1.3 The client and the server MUST also set the appropriate state transition value in the

Errata Published*	Description
	dwStateTransition field in the PDU. This is used to determine the next action to take. For state transitions, see Processing License Error Messages. Changed to: Both the client and the server can send a Licensing Error Message (section 2.2.2.8). Whenever an error message is sent, the message type in the Licensing Preamble (section 2.2.1.2) MUST be set to ERROR_ALERT (0xFF). For the PDU, see [MS-RDPBCGR] section 2.2.1.12.1.3 The client and the server MUST also set the appropriate state transition value in the dwStateTransition field in the PDU. This is used to determine the next action to take. For state transitions, see Processing Licensing Error Messages (section 3.1.5.3). In Section 3.1.5.3 Processing Licensing Error Message, changed from: Both the server and the client can send a license error message (section 2.2.2.8) and indicate a state transition with the error code. Possible state transitions include the following: Changed to: Both the server and the client can send a Licensing Error Message (section 2.2.2.8) and indicate a state transition with the error code. Possible state transitions include the following: In Section 3.2.5.2 Processing Client New License Requests, changed from: In case of a personal terminal server, no processing is done on the server side, and the server sends a license error message with the error code STATUS_VALID_CLIENT and the state transition code ST_NO_TRANSITION. The licensing protocol is complete at this point. Changed to: In case of a personal terminal server, no processing is done on the server side, and the server sends a Licensing Error Message (section 2.2.2.8) with the error code STATUS_VALID_CLIENT and the state transition code ST_NO_TRANSITION. The licensing protocol is complete at this point. In Section 3.2.5.3Processing Client License Information (2 instances), changed from: In the case of a personal terminal server, the sent license is cached by the server, and then the server sends a license error message with error code STATUS_VALID_CLIENT and the state transition code ST_NO_TRANSITION. The licensing protocol is complete at this point. & Case 1: The client presents a valid permanent license that does not require an upgrade. The server MUST send a license error message (section 2.2.2.8) with the error code STATUS_VALID_CLIENT and the state transition code ST_NO_TRANSITION. The licensing protocol is complete at this point. Changed to: In the case of a personal terminal server, the sent license is cached by the server, and then the server sends a Licensing Error Message (section 2.2.2.8) with error code STATUS_VALID_CLIENT and the state transition code ST_NO_TRANSITION. The licensing protocol is complete at this point. & Case 1: The client presents a valid permanent license that does not require an upgrade. The server MUST send a Licensing Error Message with the error code STATUS_VALID_CLIENT and the state transition code ST_NO_TRANSITION. The licensing protocol is complete at this point. In Section 3.2.5.5 Processing Client Platform Challenge Responses, changed from:

Errata Published*	Description
	In this case, if the server's grace period has not been exceeded, the server responds as if the client presented a valid license by sending a license error message (section 2.2.2.8) with an error code of STATUS_VALID_CLIENT (0x00000007) and a state transition code of ST_NO_TRANSITION (0x00000002), ending the licensing protocol. If the server's grace period has been exceeded, it sends a license error message (section 2.2.2.8) with error code ERR_NO_LICENSE_SERVER (0x00000006) and a state transition of ST_TOTAL_ABORT (0x00000001). The licensing protocol is aborted. & In this case, if the grace period has not been exceeded, the server responds as if the client presented a valid license by sending a license error message (section 2.2.2.8) with an error code of STATUS_VALID_CLIENT (0x00000007) and a state transition code of ST_NO_TRANSITION (0x00000002), ending the licensing protocol. If the server's grace period has been exceeded, it sends a license error message (section 2.2.2.8) with an error code of ERR_INVALID_CLIENT (0x00000008) and a state transition of ST_TOTAL_ABORT (0x00000001). The licensing protocol is aborted. Changed to: In this case, if the server's grace period has not been exceeded, the server responds as if the client presented a valid license by sending a Licensing Error Message (section 2.2.2.8) with an error code of STATUS_VALID_CLIENT (0x00000007) and a state transition code of ST_NO_TRANSITION (0x00000002), ending the licensing protocol. If the server's grace period has been exceeded, it sends a Licensing Error Message with error code ERR_NO_LICENSE_SERVER (0x00000006) and a state transition of ST_TOTAL_ABORT (0x00000001). The licensing protocol is aborted. & In this case, if the grace period has not been exceeded, the server responds as if the client presented a valid license by sending a Licensing Error Message with an error code of STATUS_VALID_CLIENT (0x00000007) and a state transition code of ST_NO_TRANSITION (0x00000002), ending the licensing protocol. If the server's grace period has been exceeded, it sends a Licensing Error Message with an error code of ERR_INVALID_CLIENT (0x00000008) and a state transition of ST_TOTAL_ABORT (0x00000001). The licensing protocol is aborted. In Section 3.2.5.8 Handling Out of Sequence or Unrecognized Messages, changed from: If the server receives a message that is not expected according to the Licensing PDU Flow, or a malformed or an unrecognized message, the server MUST send a License Error message (section 2.2.2.8) with an error code of ERR_INVALID_CLIENT and a state transition code of ST_TOTAL_ABORT. Changed to: If the server receives a message that is not expected according to the Licensing PDU Flow (section 1.3.3), or a malformed or an unrecognized message, the server MUST send a Licensing Error Message (section 2.2.2.8) with an error code of ERR_INVALID_CLIENT and a state transition code of ST_TOTAL_ABORT. In Section 3.2.5.9 Handling Invalid MACs, changed from: If the MAC generated over decrypted fields of a message does not match the MAC contained in the message, the server MUST send a License Error message (section 2.2.2.8) with an error code of ERR_INVALID_MAC and a state transition code of ST_TOTAL_ABORT. Changed to:

Errata Published*	Description
	If the MAC generated over decrypted fields of a message does not match the MAC contained in the message, the server MUST send a Licensing Error Message (section 2.2.2.8) with an error code of ERR_INVALID_MAC and a state transition code of ST_TOTAL_ABORT. In Section 3.3.5.1 Processing Server License Requests, changed from: ●If the server certificate does not authenticate correctly, the client MUST return a license error message (section 2.2.2.8) with an error code of ERR_INVALID_SERVER_CERTIFICATE (0x01) and a state transition of ST_TOTAL_ABORT (0x01). The server MUST then end the licensing protocol. ●The client searches its license store to find a CAL that matches the Product Information packet provided in the Server License Request. If the client finds a matching license, it MUST respond with a Client License Information message. ●If the client does not find a license matching the product information provided in the Server License Request, it MUST request a new license by sending the Client New License Request message. ●The client MAY also choose to end the licensing protocol by sending a license error message (section 2.2.2.8) with an error code of ERR_NO_LICENSE (0x02) and a state transition of ST_TOTAL_ABORT (0x01). Changed to: ●If the Server Certificate (section 2.2.1.4) does not authenticate correctly, the client MUST return a Licensing Error Message (section 2.2.2.8) with an error code of ERR_INVALID_SERVER_CERTIFICATE (0x01) and a state transition of ST_TOTAL_ABORT (0x01). The server MUST then end the licensing protocol. ●The client searches its license store to find a CAL that matches the Product Information (section 2.2.2.1.1) packet provided in the Server License Request. If the client finds a matching license, it MUST respond with a Client License Information (section 2.2.2.3) message. ●If the client does not find a license matching the Product Information provided in the Server License Request, it MUST request a new license by sending the Client New License Request (section 2.2.2.2) message. ●The client MAY also choose to end the licensing protocol by sending a Licensing Error Message with an error code of ERR_NO_LICENSE (0x02) and a state transition of ST_TOTAL_ABORT (0x01). In Section 3.3.5.9 Handling Invalid MACs, changed from: If the MAC generated over decrypted fields of a message does not match the MAC contained in the message, the client MAY send a License Error message (section 2.2.2.8) with an error code of ERR_INVALID_MAC and a state transition code of ST_TOTAL_ABORT. The client then MUST disconnect the RDP connection. Changed to: If the MAC generated over decrypted fields of a message does not match the MAC contained in the message, the client MAY send a Licensing Error Message (section 2.2.2.8) with an error code of ERR_INVALID_MAC and a state transition code of ST_TOTAL_ABORT. The client then MUST disconnect the RDP connection. In Section 6 Appendix A: Product Behavior, changed from: <16> Section 3.1.5.3: In Windows XP, the RDP connection is not disconnected on receiving ST_TOTAL_ABORT as the state transition in the license error message (section 2.2.2.8). Changed to:

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
<16> Section 3.1.5.3: In Windows XP, the RDP connection is not disconnected on receiving ST_TOTAL_ABORT as the state transition in the Licensing Error Message (section 2.2.2.8).

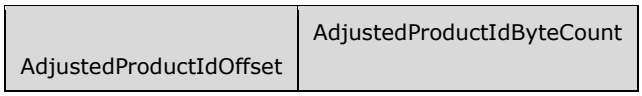
In Section 2.2.2 Licensing PDU, revised descriptions for the values of LicensingMessage.

Changed from:
LicensingMessage (variable): A variable-length licensing message whose structure depends on the value of the bMsgType field in the preamble structure. The following table lists possible values for bMsgType and the associated licensing message (this table also appears in [MS-RDPBCGR] section 2.2.1.12.1.1).
Sent by the server.
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>LICENSE_REQUEST
0x01</td><td>The Licensing PDU is a License Request PDU, and the LicensingMessage contains a Server License Request.</td></tr>
<tr><td>PLATFORM_CHALLENGE
0x02</td><td>The Licensing PDU is a Platform Challenge PDU, and the LicensingMessage contains a Server Platform Challenge.</td></tr>
<tr><td>NEW_LICENSE
0x03</td><td>The Licensing PDU is a New License PDU, and the LicensingMessage contains a Server New License structure.</td></tr>
<tr><td>UPGRADE_LICENSE
0x04</td><td>The Licensing PDU is an Upgrade License PDU, and the LicensingMessage contains a Server Upgrade License structure.
Sent by the client.</td></tr>
</table>
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>LICENSE_INFO
0x12</td><td>The Licensing PDU is a License Info PDU, and the LicensingMessage contains a Client License Information structure.</td></tr>
<tr><td>NEW_LICENSE_REQUEST
0x13</td><td>The Licensing PDU is a New License Request PDU, and the LicensingMessage contains a Client New License Request structure.</td></tr>
<tr><td>PLATFORM_CHALLENGE_RESPONSE
0x15</td><td>The Licensing PDU is a Platform Challenge Response PDU, and the LicensingMessage contains a Client Platform Challenge Response structure.
Sent by either the client or the server.</td></tr>
</table>
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>ERROR_ALERT
0xFF</td><td>The Licensing PDU is a Licensing Error Message PDU, and the LicensingMessage contains a license error message (section 2.2.2.8) structure.</td></tr>
</table>
Changed to:
LicensingMessage (variable): A variable-length licensing message whose structure depends on the value of the bMsgType field in the preamble structure. The following table lists possible values for bMsgType and the associated licensing message (this table also appears in [MS-RDPBCGR] section
</td></tr>
</table>

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
2.2.1.12.1.1).
Sent by the server.
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>LICENSE_REQUEST
0x01</td><td>The Licensing PDU is a License Request PDU, and the LicensingMessage contains a SERVER_LICENSE_REQUEST (section 2.2.2.1) structure.</td></tr>
<tr><td>PLATFORM_CHALLENGE
0x02</td><td>The Licensing PDU is a Platform Challenge PDU, and the LicensingMessage contains a SERVER_PLATFORM_CHALLENGE (section 2.2.2.4) structure.</td></tr>
<tr><td>NEW_LICENSE
0x03</td><td>The Licensing PDU is a New License PDU, and the LicensingMessage contains a SERVER_NEW_LICENSE (section 2.2.2.7) structure.</td></tr>
<tr><td>UPGRADE_LICENSE
0x04</td><td>The Licensing PDU is an Upgrade License PDU, and the LicensingMessage contains a SERVER_UPGRADE_LICENSE (section 2.2.2.6) structure.
Sent by the client.</td></tr>
</table>
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>LICENSE_INFO
0x12</td><td>The Licensing PDU is a License Info PDU, and the LicensingMessage contains a CLIENT_LICENSE_INFO (section 2.2.2.3) structure.</td></tr>
<tr><td>NEW_LICENSE_REQUEST
0x13</td><td>The Licensing PDU is a New License Request PDU, and the LicensingMessage contains a CLIENT_NEW_LICENSE_REQUEST (section 2.2.2.2) structure.</td></tr>
<tr><td>PLATFORM_CHALLENGE_RESPONSE
0x15</td><td>The Licensing PDU is a Platform Challenge Response PDU, and the LicensingMessage contains a CLIENT_PLATFORM_CHALLENGE_RESPONSE (section 2.2.2.5) structure.
Sent by either the client or the server.</td></tr>
</table>
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>ERROR_ALERT
0xFF</td><td>The Licensing PDU is a Licensing Error Message PDU, and the LicensingMessage contains a LICENSE_ERROR_MESSAGE (section 2.2.2.8) structure.</td></tr>
</table>

In Section 2.2.2.9.1 Licensed Product Info (LICENSED_PRODUCT_INFO), revised 'AdjustedProductdOffset' to 'AdjustedProductIdOffset'.

Changed from:
<table>
<tr><td>AdjustedProductdOffset</td><td>AdjustedProductIdByteCount</td></tr>
</table>
</td></tr>
</table>

Errata Published*	Description
	Changed to:  In Section 3.1.5.2 Sending Licensing Error Message, revised title. Changed from: 3.1.5.2 Sending License Error Messages Changed to: 3.1.5.2 Sending Licensing Error Messages In Section 3.1.5.3 Processing Licensing Error Message, revised title. Changed from: 3.1.5.3 Processing License Error Messages Changed to: 3.1.5.3 Processing Licensing Error Messages In Section 3.2.1.6 Platform Challenge, removed 'Encrypting License Data' from description. Changed from: The platform challenge is a random string generated by the server. This string is encrypted (see Encrypting Licensing Data (section 5.1.3)) with the licensing encryption key using RC4 and sent in the EncryptedPlatformChallenge field of the Server Platform Challenge message. It is created at the beginning of the licensing protocol and destroyed when the licensing protocol is completed. Changed to: The Platform Challenge is a random string generated by the server. This string is encrypted (section 5.1.3) with the licensing encryption key using RC4 and sent in the EncryptedPlatformChallenge field of the Server Platform Challenge (section 2.2.2.4) message. It is created at the beginning of the licensing protocol and destroyed when the licensing protocol is completed. In Section 3.3.1.9 Client Hardware Identification, revised 'PlatformID' to 'PlatformId'. Changed from: The content and format of the PlatformID field of Client Hardware Identification are the same as the PlatformID field of the Client License Information and Client New License Request messages. This ties a particular Client Hardware Identification to the client's operating system. The other 16 bytes (fields Data1 through Data4) of the Client Hardware Identification are intended to be hardware-specific. Clients SHOULD attempt to use operating system-specific or hardware-specific information that is easily and consistently retrievable. Examples include hard-wired processor IDs, Ethernet addresses of nonremovable Ethernet cards, and disk subsystem serial numbers. The client SHOULD cache the Client Hardware Identification for later retrieval after it is generated. Changed to: The content and format of the PlatformId field of Client Hardware Identification (section 2.2.2.3.1) are the same as the PlatformId field of the Client License Information (section 2.2.2.3) and Client New License Request (section 2.2.2.2) messages. This ties a particular Client Hardware Identification to the client's operating system. The other 16 bytes (fields Data1 through Data4) of the Client Hardware Identification are intended to be hardware specific. Clients SHOULD attempt

Errata Published*	Description

	to use operating system-specific or hardware-specific information that is easily and consistently retrievable. Examples include hard-wired processor IDs, Ethernet addresses of nonremovable Ethernet cards, and disk subsystem serial numbers. The client SHOULD cache the Client Hardware Identification for later retrieval after it is generated.
	
	In Section 4.1 Server License Request (SERVER_LICENSE_REQUEST), revised title.
	
	Changed from:
	4.1 SERVER LICENSE REQUEST
	Changed to:
	4.1 Server License Request (SERVER_LICENSE_REQUEST)
	
	In Section 4.2 Client New License Request (CLIENT_NEW_LICENSE_REQUEST), revised title.
	
	Changed from:
	4.2 CLIENT NEW LICENSE REQUEST
	Changed to:
	4.2 Client New License Request (CLIENT_NEW_LICENSE_REQUEST)
	
	In Section 4.2 Client New License Request (CLIENT_NEW_LICENSE_REQUEST), revised '_PREAMBE' to '_PREAMBLE'.
	
	Changed from:
	
	`0x00: LICENSE_PREAMBLE (4 bytes)`
	`13 -> LICENSE_PREAMBE::bMsgType = NEW_LICENSE_REQUEST`
	
	Changed to:
	
	`0x00: LICENSE_PREAMBLE (4 bytes)`
	`13 -> LICENSE_PREAMBLE::bMsgType = NEW_LICENSE_REQUEST`
	
	In Section 4.3 Client License Information (CLIENT_LICENSE_INFO), revised title.
	
	Changed from:
	4.3 CLIENT LICENSE INFO
	Changed to:
	4.3 Client License Information (CLIENT_LICENSE_INFO)
	
	In Section 4.3 Client License Information (CLIENT_LICENSE_INFO), revised '_PREAMBE' to '_PREAMBLE'.
	
	Changed from:
	
	`0x00: LICENSE_PREAMBLE (4 bytes)`
	`12 -> LICENSE_PREAMBLE::bMsgType =`

Errata Published*	Description
	CLIENT LICENSE INFO 83 -> LICENSE_PREAMBE::bVersion = 0x80 | 0x3 Changed to: 0x00: LICENSE_PREAMBLE (4 bytes) 12 -> LICENSE PREAMBLE::bMsgType = CLIENT_LICENSE_INFO 83 -> LICENSE_PREAMBLE::bVersion = 0x80 | 0x3 In Section 4.4 Server Platform Challenge (SERVER_PLATFORM_CHALLENGE), revised title. Changed from: 4.4 SERVER PLATFORM CHALLENGE Changed to: 4.4 Server Platform Challenge (SERVER_PLATFORM_CHALLENGE) In Section 4.5 Client Platform Challenge Response (CLIENT_PLATFORM_CHALLENGE_RESPONSE), revised title. Changed from: 4.5 CLIENT PLATFORM CHALLENGE RESPONSE Changed to: 4.5 Client Platform Challenge Response (CLIENT_PLATFORM_CHALLENGE_RESPONSE) In Section 4.6 Server New License (SERVER_NEW_LICENSE), revised title. Changed from: 4.6 SERVER NEW LICENSE Changed to: 4.6 Server New License (SERVER_NEW_LICENSE) In Section 4.7 Server Upgrade License (SERVER_UPGRADE_LICENSE), revised title. Changed from: 4.7 SERVER UPGRADE LICENSE Changed to: 4.7 Server Upgrade License (SERVER_UPGRADE_LICENSE)
2020/07/06	In Section 2.2.2.9, X.509 Certificate Extensions, revised certificate object identifier from 'OID' to 'szOID'. Changed from: ●"1.3.6.1.4.1.311.18.4" (OID_HYDRA_CERT_VERSION) ●"1.3.6.1.4.1.311.18.2" (OID_MANUFACTURER)

Errata Published*	Description
	●"1.3.6.1.4.1.311.18.5" (OID_LICENSED_PRODUCT_INFO) ●"1.3.6.1.4.1.311.18.6" (OID_MS_LICENSE_SERVER_INFO) ●"1.3.6.1.4.1.311.18.7" (OID_PRODUCT_SPECIFIC_OID) Changed to: ●"1.3.6.1.4.1.311.18.4" (szOID_PKIX_HYDRA_CERT_VERSION) ●"1.3.6.1.4.1.311.18.2" (szOID_PKIX_MANUFACTURER) ●"1.3.6.1.4.1.311.18.5" (szOID_PKIX_LICENSED_PRODUCT_INFO) ●"1.3.6.1.4.1.311.18.6" (szOID_PKIX_MS_LICENSE_SERVER_INFO) ●"1.3.6.1.4.1.311.18.7" (szOID_PKIS_PRODUCT_SPECIFIC_OID)

*Date format: YYYY/MM/DD

[MS-RDPEMC]: Remote Desktop Protocol: Multiparty Virtual Channel Extension

This topic lists the Errata found in [MS-RDPEMC] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-RDPEMT]: Remote Desktop Protocol: Multitransport Extension

This topic lists the Errata found in [MS-RDPEMT] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

March 2, 2016 - Download

Errata below are for Protocol Document Version V10.0 – 2018/09/12.

Errata Published*	Description
2019/03/18	In Section 1.3, Overview, clarified that a port number is not specified in an Initiate Multitransport Request PDU. Changed from: The Initiate Multitransport Request PDU contains information that uniquely identifies the multitransport connection; it contains a request ID and a cookie, a protocol identifier that identifies the type of multitransport connection that the client attempts to establish, and a port number that identifies the port on which the server is listening. When the client receives the Initiate Multitransport Request PDU, it attempts to establish a secure multitransport connection with the server. Changed to: The Initiate Multitransport Request PDU contains information that uniquely identifies the multitransport connection; it contains a request ID, a cookie, and a protocol identifier that identifies the type of multitransport connection that the client attempts to establish. When the client receives the Initiate Multitransport Request PDU, it attempts to establish a secure multitransport connection with the server.

*Date format: YYYY/MM/DD

[MS-RDPEPC]: Remote Desktop Protocol: Print Virtual Channel Extension

This topic lists the Errata found in [MS-RDPEPC] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

June 1, 2017 - Download

Errata below are for Protocol Document Version V10.0 – 2018/09/12.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td>2019/07/08</td><td>In Section 2.2.2.1, Client Device List Announce Request (DR_PRN_DEVICE_ANNOUNCE), added the section number that describes XPS mode to the RDPDR_PRINTER_ANNOUNCE_FLAG_XPSFORMAT value meaning in the Flags field table.

Changed from:
...
Flags (4 bytes): A 32-bit unsigned integer that indicates the properties of the client printer queue. This bit field MUST be a valid combination of any of the following values.
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>...</td><td>,,,</td></tr>
<tr><td>RDPDR_PRINTER_ANNOUNCE_FLAG_XPSFORMAT
0x00000010</td><td>This client/printer supports XML Paper Specification (XPS) format.</td></tr>
<tr><td>...</td><td>,,,</td></tr>
</table>

Changed to:
...
Flags (4 bytes): A 32-bit unsigned integer that indicates the properties of the client printer queue. This bit field MUST be a valid combination of any of the following values.
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>...</td><td>…</td></tr>
<tr><td>RDPDR_PRINTER_ANNOUNCE_FLAG_XPSFORMAT
0x00000010</td><td>This client/printer supports XML Paper Specification (XPS) format (section 3.1.1.2).</td></tr>
</table>
</td></tr>
</table>

Errata Published*	Description
	... | … | In Section 2.2.2.2, Server Printer Set XPS Mode (DR_PRN_USING_XPS), added that the DR_PRN_USING_XPS message indicates to the client that future printer write request messages will use the XPS format. Changed from: This message is sent from server to client to set the device in XPS mode (see section 3.1.1.2). ... Changed to: This message is sent from server to client to set the device in XPS mode (see section 3.1.1.2) and indicate to the client that future Printer Write Request (section 2.2.2.9) messages will use the XPS format. ... In Section 3.1.1.2, XPS Mode, added the section number that describes the server behavior if it chooses to use the XPS format. Changed from: ... The server MUST notify the client with the message DR_PRN_USING_XPS (section 2.2.2.2) if it chooses to use the XPS format. ... Changed to: ... The server MUST notify the client with the DR_PRN_USING_XPS (section 2.2.2.2) message as described in section 3.3.5.1.2 if it chooses to use the XPS format. ... In Section 3.3.5.1.2, Sending a Printer Set XPS Mode Message, clarified the server Printer Write Request message section number. Changed from: ... If the server chooses to send print data in XPS format, the server MUST send this message to the client prior to sending any data in the write request messages (section 2.2.2.1). Changed to: ... If the server chooses to send print data in XPS format, the server MUST send this message to the client prior to sending any data in the Printer Write Request (section 2.2.2.9) message.

*Date format: YYYY/MM/DD

[MS-RDPEPNP]: Remote Desktop Protocol: Plug and Play Devices Virtual Channel Extension

This topic lists the Errata found in [MS-RDPEPNP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-RDPERP]: Remote Desktop Protocol: Remote Programs Virtual Channel Extension

This topic lists the Errata found in [MS-RDPERP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

March 2, 2016 - Download

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

September 15, 2017 - Download

September 12, 2018 - Download

September 23, 2019 - Download

March 4, 2020 - Download

Errata below are for Protocol Document Version V29.0 – 2020/03/04.

Errata Published*	Description

2020/07/06	In Section 4.1.1.1 New or Existing Windows, added the 'TS_WINDOW_ORDER_HEADER' prefix to 'WindowId'.
	Changed from:
	
	`2e -> TS WINDOW ORDER HEADER::Header (1 Byte)`
	`81 00 -> TS_WINDOW_ORDER_HEADER::OrderSize (2 Bytes)`
	`9e df 08 19 -> TS_WINDOW_ORDER_HEADER::FieldsPresentFlags (4 Bytes)`
	`58 01 12 00 -> WindowId`
	`00 00 00 00 -> OwnerWindowId`
	`00 00 cf 14 -> Style`
	`00 01 00 00 -> ExtendedStyle`
	`05 -> ShowState`
	
	Changed to:
	
	`2e -> TS_WINDOW_ORDER_HEADER::Header (1 Byte)`
	`81 00 -> TS_WINDOW_ORDER_HEADER::OrderSize (2 Bytes)`
	`9e df 08 19 -> TS_WINDOW_ORDER_HEADER::FieldsPresentFlags (4 Bytes)`
	`58 01 12 00 -> TS WINDOW ORDER HEADER::WindowId (4 Bytes)`
	`00 00 00 00 -> OwnerWindowId`

Errata Published*	Description
	00 00 cf 14 -> Style 00 01 00 00 -> ExtendedStyle 05 -> ShowState In Section 4.1.1.2 Deleted Window, added the 'TS_WINDOW_ORDER_HEADER' prefix to 'WindowId'. Changed from: 00000000 2e 0b 00 00 00 00 21 24 00 03 00......!$... 2e -> TS_WINDOW_ORDER_HEADER::Header (1 Byte) 0b 00 -> TS WINDOW ORDER HEADER::OrderSize (2 Bytes) 00 00 00 21 -> TS_WINDOW_ORDER_HEADER::FieldsPresentFlags (4 Bytes) (WINDOW_ORDER_TYPE_WINDOW \| WINDOW_ORDER_STATE_DELETED) 24 00 03 00 -> WindowId (4 Bytes) Changed to: 00000000 2e 0b 00 00 00 00 21 24 00 03 00......!$... 2e -> TS_WINDOW_ORDER_HEADER::Header (1 Byte) 0b 00 -> TS_WINDOW_ORDER_HEADER::OrderSize (2 Bytes) 00 00 00 21 -> TS_WINDOW_ORDER_HEADER::FieldsPresentFlags (4 Bytes) (WINDOW_ORDER_TYPE_WINDOW | WINDOW_ORDER_STATE_DELETED) 24 00 03 00 -> TS_WINDOW_ORDER_HEADER::WindowId (4 Bytes) In Section 4.4.1 S_RAIL_ORDER_SYSPARAM, added the 'TS_HIGHCONTRAST::' prefix to 'Flags', 'ColorSchemeLength', and 'ColorSchema'. Changed from: 03 00 -> TS_RAIL_PDU_HEADER::orderType = TS_RAIL_ORDER_SYSPARAM(3) (2 Bytes) 12 00 -> TS_RAIL_PDU_HEADER::orderLength = 18 (2 Bytes) 43 00 00 00 -> SystemParam: SPI SETHIGHCONTRAST (4 Bytes) 7e 00 00 00 -> Flags: 0x7e (4 Bytes) 02 00 00 00 -> ColorSchemeLength: 2 (4 Bytes) 00 00 -> ColorScheme: 0 (2 Bytes) Changed to: 03 00 -> TS_RAIL_PDU_HEADER::orderType = TS_RAIL_ORDER_SYSPARAM(3) (2 Bytes) 12 00 -> TS_RAIL_PDU_HEADER::orderLength = 18 (2 Bytes) 43 00 00 00 -> SystemParam: SPI_SETHIGHCONTRAST (4 Bytes) 7e 00 00 00 -> TS HIGHCONTRAST::Flags: 0x7e (4 Bytes) 02 00 00 00 -> TS-HIGHCONTRAST::ColorSchemeLength: 2 (4 Bytes)

Errata Published*	Description
	00 00 -> TS_HIGHCONTRAST::ColorScheme: 0 (2 Bytes)
2020/07/06	In Section 2.2.1.3.1.2.1 New or Existing Window, removed the extraneous space from 'FIELD_ RESIZE' Changed from: WindowRightResizeMargin (4 bytes): An unsigned 32-bit integer specifying the width of the transparent hit-testable margin along the right edge of the window. Any mouse, pen or touch input within this margin SHOULD be sent to the server. This field is present only if the WINDOW_ORDER_FIELD_ RESIZE_MARGIN_X flag is set in the FieldsPresentFlags field of TS_WINDOW_ORDER_HEADER. Resize margins SHOULD be used to extend the window geometry (defined by the WindowOffsetX, WindowOffsetY, WindowWidth and WindowHeight fields) and are not included in the window boundaries. WindowTopResizeMargin (4 bytes): An unsigned 32-bit integer specifying the height of the transparent hit-testable margin along the top edge of the window. Any mouse, pen or touch input within this margin SHOULD be sent to the server. This field is present only if the WINDOW_ORDER_FIELD_ RESIZE_MARGIN_Y flag is set in the FieldsPresentFlags field of TS_WINDOW_ORDER_HEADER. Resize margins SHOULD be used to extend the window geometry (defined by the WindowOffsetX, WindowOffsetY, WindowWidth and WindowHeight fields) and are not included in the window boundaries. Changed to: WindowRightResizeMargin (4 bytes): An unsigned 32-bit integer specifying the width of the transparent hit-testable margin along the right edge of the window. Any mouse, pen or touch input within this margin SHOULD be sent to the server. This field is present only if the WINDOW_ORDER_FIELD_RESIZE_MARGIN_X flag is set in the FieldsPresentFlags field of TS_WINDOW_ORDER_HEADER. Resize margins SHOULD be used to extend the window geometry (defined by the WindowOffsetX, WindowOffsetY, WindowWidth and WindowHeight fields) and are not included in the window boundaries. WindowTopResizeMargin (4 bytes): An unsigned 32-bit integer specifying the height of the transparent hit-testable margin along the top edge of the window. Any mouse, pen or touch input within this margin SHOULD be sent to the server. This field is present only if the WINDOW_ORDER_FIELD_RESIZE_MARGIN_Y flag is set in the FieldsPresentFlags field of TS_WINDOW_ORDER_HEADER. Resize margins SHOULD be used to extend the window geometry (defined by the WindowOffsetX, WindowOffsetY, WindowWidth and WindowHeight fields) and are not included in the window boundaries. In Section 2.2.1.3.1.2.3 Cached Icon, capitalized field names cacheEntry and cacheId. Changed from: The Cached Icon Window Information Order is generated by the server when a new or existing window sets or updates the icon in its title bar or in the Alt-Tab dialog box. If the icon information was transmitted by the server in a previous Window Information Order or Notification Icon Information Order in the same session, and the icon was cacheable (that is, the server specified a cacheEntry and cacheId for the icon), the server reports the icon cache entries to avoid sending duplicate information. Changed to: The Cached Icon Window Information Order is generated by the server when a new or existing

Errata Published*	Description
	window sets or updates the icon in its title bar or in the Alt-Tab dialog box. If the icon information was transmitted by the server in a previous Window Information Order or Notification Icon Information Order in the same session, and the icon was cacheable (that is, the server specified a CacheEntry and CacheId for the icon), the server reports the icon cache entries to avoid sending duplicate information. In Section 2.2.1.3.1.2.3 Cached Icon, revised 'TS_CACHED ICON_INFO' to 'TS_CACHED_ICON_INFO'. Changed from: CachedIcon (3 bytes): Three bytes. TS_CACHED ICON_INFO structure. Describes a cached icon on the client. Changed to: CachedIcon (3 bytes): Three bytes. TS_CACHED_ICON_INFO (section 2.2.1.2.4) structure. Describes a cached icon on the client. In Section 4.1.1.6 Non-Monitored Desktop, revised title from 'Non-monitored Desktop' Changed from: 4.1.1.6 Non-monitored Desktop Changed to: 4.1.1.6 Non-Monitored Desktop In Section 3.2.5.1.8 Processing Desktop Information Orders, revised 'non-monitored desktop' to 'Non-Monitored Desktop' to reflect section title change. Changed from: • Upon receipt of a Desktop Information Order for a non-monitored desktop, as specified in section 2.2.1.3.3.2.2, the client SHOULD discard all of the existing RAIL windows and Notify Icons. Changed to: • Upon receipt of a Desktop Information Order for a Non-Monitored Desktop packet, as specified in section 2.2.1.3.3.2.2, the client SHOULD discard all of the existing RAIL windows and Notify Icons.

*Date format: YYYY/MM/DD

[MS-RDPESC]: Remote Desktop Protocol: Smart Card Virtual Channel Extension

This topic lists the Errata found in [MS-RDPESC] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

June 1, 2017 - Download

[MS-RDPESP]: Remote Desktop Protocol: Serial and Parallel Port Virtual Channel Extension

This topic lists the Errata found in [MS-RDPESP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-RDPEUDP]: Remote Desktop Protocol: UDP Transport Extension

This topic lists the Errata found in [MS-RDPEUDP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

March 2, 2016 - Download

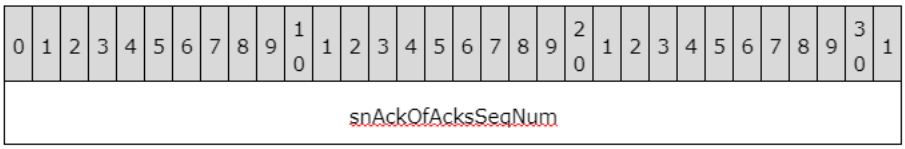
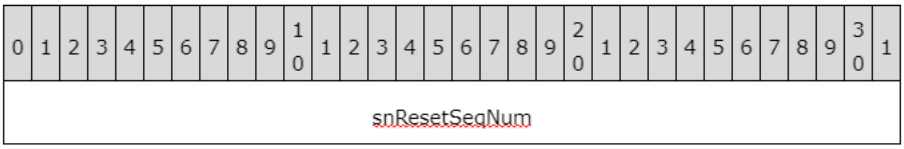
October 16, 2015 - Download

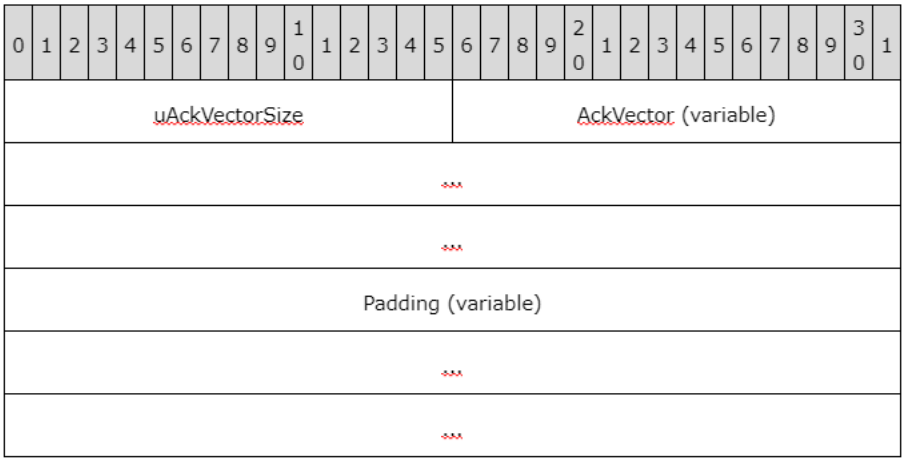
June 30, 2015 - Download

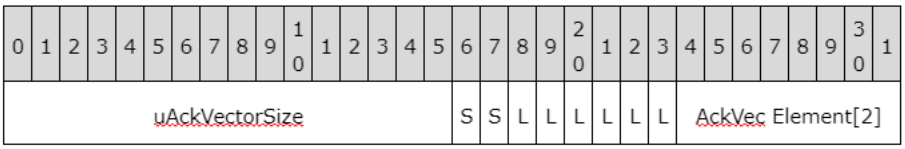
June 1, 2017 - Download

September 23, 2019 - Download

Errata below are for Protocol Document Version V14.0 – 2020/03/04.

Errata Published*	Description
2020/07/06	In Section 2.2.2.6 RDPUDP_ACK_OF_ACKVECTOR_HEADER Structure, revised the field snAckofAcksSeqNum to snResetSeqNum, and updated its description. Changed from: The RDPUDP_ACK_OF_ACKVECTOR_HEADER structure resets the start position of an ACK vector (section 2.2.3.1).  The sender sets the AckOfAck sequence number with the greatest cumulative ACK it has received and processed. The sender SHOULD send AckOfAck every 20 packets. Changed to: The RDPUDP_ACK_OF_ACKVECTOR_HEADER structure resets the start position of an ACK vector (section 2.2.3.1). This structure SHOULD be sent after every 20 packets.  The sender populates snResetSeqNum with the greatest cumulative ACK it has received and processed.
2020/07/06	In Section 2.2.2.7 RDPUDP_ACK_VECTOR_HEADER Structure, revised structure description, field AckVectorElement to AckVector and added normative information to field definitions. Changed from:

Errata Published*	Description
	The RDPUDP_ACK_VECTOR_HEADER structure contains the ACK vector (section 2.2.3.1) that specifies the states of the datagram in the receiver's queue. This vector is a variable-size array. The states are encoded by using run-length encoding (RLE) and are stored in this array.  AckVectorElement (variable): An array of ACK Vector elements. Each element is composed of a state, and the number of contiguous datagrams that share the same state. Padding (variable): A variable-sized array, of length zero or more, such that this structure ends on a DWORD ([MS-DTYP] section 2.2.9) boundary. Changed to: The RDPUDP_ACK_VECTOR_HEADER structure contains a variable size array of ACK Vector Elements (section 2.2.2.7.1), referred to as the ACK vector. The ACK vector captures the state of the queue of Source Packets at the receiver endpoint. Each position in the queue can have two values that indicate whether a Source Packet is present in the queue, or not. Run-length encoding (RLE) compression is used to encode the state of Source Packets in the array.  AckVector (variable): A variable size array of ACK Vector Elements (section 2.2.2.7.1). The size of the AckVector field is specified by the uAckVectorSize field. Padding (variable): A variable-sized array, of length zero or more, such that this structure ends on a DWORD ([MS-DTYP] section 2.2.9) boundary.

Errata Published*	Description
	Added section 2.2.2.7.1 ACK Vector Element. … An ACK Vector Element is an 8-bit structure. The two most significant bits of each element encode the VECTOR_ELEMENT_STATE enumeration (section 2.2.1.1), while the six least significant bits specify the length of a continuous sequence of datagrams that share the same state. Removed sections 2.2.3 Vectors & 2.2.3.1 ACK Vector. 2.2.3 Vectors 2.2.3.1 ACK Vector The ACK vector captures the state of the queue of Source Packets at the receiver endpoint. Each position in the queue can have two values that indicate whether a Source Packet is present in the queue, or not. The run-length encoding (RLE) compression is used for encoding the states of Source Packets in the array. An ACK Vector comprises a number of elements, as specified by the uAckVectorSize field in the RDPUDP_ACK_VECTOR_HEADER structure (section 2.2.2.7). Each element is 8 bits long.  The ACK vectors form a binary large object (BLOB), and are padded so that they are aligned to WORD ([MS-DTYP] section 2.2.61) boundaries. This is similar to the description of ACK vectors in the Datagram Congestion Control Protocol (DCCP), as described in [RFC4341]. Revised 'section 2.2.3.1' to '2.2.2.7.1' in the following sections: . 2.2.1.1 VECTOR_ELEMENT_STATE Enumeration . 3.1.1.4.1 Lost Datagrams . 3.1.5.1.2 ACK Datagrams .5.1.2 RDP-UDP Datagram Validation
2020/07/06	In Section 3.1.1.2 Sequence Number, revised transposed characters. Changed from: Initial Sequence Number = snInitialSequenceNumber in the RPDUDP_SYNDATA_PAYLOAD Structure (section 2.2.2.5). Changed to: Initial Sequence Number = snInitialSequenceNumber in the RDPUDP_SYNDATA_PAYLOAD Structure (section 2.2.2.5). In Section 3.1.5.1.1 SYN Datagrams, corrected typos. Changed from: The RDPUDP_SYNEX_PAYLOAD structure (section 2.2.2.9) MUST be appended to the UDP datagram if the RDPUDP_FLAG_SYNEX flag is set in uFlags. Not appending this structure implies that RDPUDP_PROTOCOL_VERSION_1 is the highest protocol version supported. This structure SHOULD NOT be appended if this datagram is in response to a SYN from the other endpoint where the RDP_FLAG_SYNEX flag was not specified. The uSynExFlags field MUST be set as follows: Changed to: The RDPUDP_SYNDATAEX_PAYLOAD structure (section 2.2.2.9) MUST be appended to the UDP datagram if the RDPUDP_FLAG_SYNEX flag is set in uFlags. Not appending this structure implies

Errata Published*	Description
	that RDPUDP_PROTOCOL_VERSION_1 is the highest protocol version supported. This structure SHOULD NOT be appended if this datagram is in response to a SYN from the other endpoint where the RDPUDP_FLAG_SYNEX flag was not specified. The uSynExFlags field MUST be set as follows: In 3.1.5.1.3 SYN+ACK Datagrams, revised title. Changed from: 3.1.5.1.3 SYN and ACK Datagrams Changed to: 3.1.5.1.3 SYN+ACK Datagrams In 3.1.5.1.3 SYN+ACK Datagrams, made edits to the text. Changed from: A SYN datagram is generated, as specified in section 3.1.5.1.1, with the following fields set as follows: & ●The RDPUDP_SYNEX_PAYLOAD structure (section 2.2.2.9) SHOULD only be present if it is also present in the received SYN packet. The uUdpVer field MUST be set to the highest RDP-UDP protocol version supported by both endpoints. The highest version supported by both endpoints, which is RDPUDP_PROTOCOL_VERSION_1 if either this packet or the SYN packet does not specify a version, is the version that MUST be used by both endpoints. Changed to: A SYN+ACK datagram consists of a SYN packet, generated as specified in section 3.1.5.1.1, with these additional fields set as follows: & ●The RDPUDP_SYNDATAEX_PAYLOAD structure (section 2.2.2.9) SHOULD only be present if it is also present in the received SYN packet. The uUdpVer field MUST be set to the highest RDP-UDP protocol version supported by both endpoints. The highest version supported by both endpoints, which is RDPUDP_PROTOCOL_VERSION_1 if either this packet or the SYN packet does not specify a version, is the version that MUST be used by both endpoints. In Section 3.1.5.1.4 ACK and Source Packets Data, renamed structure. Changed from: An RDPUDP_SOURCE_PAYLOAD structure (section 2.2.2.4) header MUST be appended. Changed to: An RDPUDP_SOURCE_PAYLOAD_HEADER structure (section 2.2.2.4) header MUST be appended.

*Date format: YYYY/MM/DD

[MS-RDPEUDP2]: Remote Desktop Protocol: UDP Transport Extension Version 2

This topic lists the Errata found in [MS-RDPEUDP2] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

March 13, 2019 - Download

September 23, 2019 - Download

[MS-RDPEV]: Remote Desktop Protocol: Video Redirection Virtual Channel Extension

This topic lists the Errata found in [MS-RDPEV] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-RDPEVOR]: Remote Desktop Protocol: Video Optimized Remoting Virtual Channel Extension

This topic lists the Errata found in [MS-RDPEVOR] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

[MS-RDPEXPS]: Remote Desktop Protocol: XML Paper Specification (XPS) Print Virtual Channel Extension

This topic lists the Errata found in [MS-RDPEXPS] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-RDPRFX]: Remote Desktop Protocol: RemoteFX Codec Extension

This topic lists the Errata found in [MS-RDPRFX] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

June 1, 2017 - Download

Errata below are for Protocol Document Version V20.0 – 2018/09/12.

Errata Published*	Description
2019/02/19	In Section 4.2.4.1, Input TS_RFX_TILESET Message, updated the first line of an annotated dump of a TS_RFX_TILESET message containing a single encoded 64x64 tile from "00000000 c7 cc 3e 0b 00 00 01 01 c2 ca 00 00 51 50 01 40" to "00000000 c7 cc 3e 0b 00 00 01 00 c2 ca 00 00 51 50 01 40". Changed from: The following is an annotated dump of a TS_RFX_TILESET (section 2.2.2.3.4) message containing a single encoded 64x64 tile. 00000000 c7 cc 3e 0b 00 00 01 01 c2 ca 00 00 51 50 01 40 ... Changed to: The following is an annotated dump of a TS_RFX_TILESET (section 2.2.2.3.4) message containing a single encoded 64x64 tile. 00000000 c7 cc 3e 0b 00 00 01 00 c2 ca 00 00 51 50 01 40 ...
2019/02/19	In Section 3.1.8.1.6, Linearization, updated the converted value of -10 to 10 after coefficients from LL3 undergo differential encoding. Changed from: ... The coefficients from LL3 also undergo differential encoding. Except for the first coefficient, every raster-scanned LL3 coefficient is subtracted from its previous neighbor. For example, if the raster-scanned LL3 coefficients are [64, 32, 42, 54, 50, 60, 40, 70] Then, after differential encoding, they would get converted to

Errata Published*	Description
	[64, -32, 10, 12, -4, -10, -20, 30] Changed to: ... The coefficients from LL3 also undergo differential encoding. Except for the first coefficient, every raster-scanned LL3 coefficient is subtracted from its previous neighbor. For example, if the raster-scanned LL3 coefficients are [64, 32, 42, 54, 50, 60, 40, 70] Then, after differential encoding, they would get converted to [64, -32, 10, 12, -4, 10, -20, 30]

*Date format: YYYY/MM/DD

[MS-RMPR]: Rights Management Services (RMS): Client-to-Server Protocol

This topic lists the Errata found in [MS-RMPR] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

Errata below are for Protocol Document Version V38.0 – 2018/09/12.

Errata Published*	Description
2019/10/16	In Section 2.2.3.4, string Element, parentheses around the ArrayOfString element have been removed. In Section 3.4.4.3.2.1, AcquireTemplates, parentheses around the ArrayOfString element have been removed. In Section 3.4.4.3.2.2, AcquireTemplatesResponse: Changed from: ArrayOfGuideTemplate Changed to: ArrayOfGuidTemplate In Section 3.5.4.2.3.2, ArrayOfGetClientLicensorCertResponse: Changed from: name="ArrayOfGetClIentLicensorCertResponse"> <xs:sequence> Changed to: name="ArrayOfGetClientLicensorCertResponse"> <xs:sequence>

Errata Published*	Description
	In Section 3.6.4.1, Synchronous Enrollment Operation, and Section 3.6.4.2, Asynchronous Enrollment Operation: Changed from: serverState Changed to: ServerState

*Date format: YYYY/MM/DD

[MS-RMSOD]: Rights Management Services Protocols Overview

This topic lists the Errata found in [MS-RMSOD] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

June 1, 2017 - Download

[MS-RPCE]: Remote Procedure Call Protocol Extensions

This topic lists the Errata found in the MS-RPCE document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

September 15, 2017 - Download

[MS-RPCH]: Remote Procedure Call over HTTP Protocol

This topic lists the Errata found in [MS-RPCH] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-RPRN]: Print System Remote Protocol

This topic lists the Errata found in [MS-RPRN] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

December 1, 2017 - Download

Errata below are for Protocol Document Version V32.0 – 2018/09/12.

Errata Published*	Description
2020/03/30	In Section 1.7, Versioning and Capability Negotiation, the build number for Windows Server 2019 has been added to product behavior note <2>. Changed from: Version / dwBuildNumber value Windows Server operating system / >= 16299 Windows 10 and Windows Server 2016 / >= 10586 Windows 8.1 and Windows Server 2012 R2 / >= 9431 Windows 8 and Windows Server 2012 / >= 9200 Windows 7 and Windows Server 2008 R2 / >= 7007 Windows Vista operating system with Service Pack 1 (SP1) and Windows Server 2008 / >= 6001 Windows Vista and Windows Server 2008 / >= 6000 Windows XP operating system Service Pack 1 (SP1) / >= 2196 Windows XP and Windows Server 2003 / >= 2196 Windows 2000 / >= 1382 Windows NT 4.0 / >= 1381

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>Changed to:

<table>
<tr><th>Version</th><th>dwBuildNumber value</th></tr>
<tr><td>Windows Server 2019</td><td>>= 17633</td></tr>
<tr><td>Windows Server operating system</td><td>>= 16299</td></tr>
<tr><td>Windows 10 and Windows Server 2016</td><td>>= 10586</td></tr>
<tr><td>Windows 8.1 and Windows Server 2012 R2</td><td>>= 9431</td></tr>
<tr><td>Windows 8 and Windows Server 2012</td><td>>= 9200</td></tr>
<tr><td>Windows 7 and Windows Server 2008 R2</td><td>>= 7007</td></tr>
<tr><td>Windows Vista operating system with Service Pack 1 (SP1) and Windows Server 2008</td><td>>= 6001</td></tr>
<tr><td>Windows Vista and Windows Server 2008</td><td>>= 6000</td></tr>
<tr><td>Windows XP operating system Service Pack 1 (SP1)</td><td>>= 2196</td></tr>
<tr><td>Windows XP and Windows Server 2003</td><td>>= 2196</td></tr>
<tr><td>Windows 2000</td><td>>= 1382</td></tr>
<tr><td>Windows NT 4.0</td><td>>= 1381</td></tr>
</table></td></tr>
<tr><td>2018/12/10</td><td>In Section 1.7, Versioning and Capability Negotiation, changed from:

• Capability Negotiation: Functional negotiation … by comparing the value returned by the server in the dwBuildNumber member of OSVERSIONINFO (section 2.2.3.10.1) with well-known version-specific dwBuildNumber values.<2>
<2> Section 1.7: The values of the dwBuildNumber member in the OSVERSIONINFO structure (section 2.2.3.10.1) for specific versions of Windows are shown in the table that follows.
<table>
<tr><th>Version</th><th>dwBuildNumber value</th></tr>
<tr><td>Windows 10 and Windows Server 2016</td><td>>= 10586</td></tr>
<tr><td>…</td><td>…</td></tr>
</table>
Changed to:
• Capability Negotiation: Functional negotiation … by comparing the value returned by the server in the dwBuildNumber member of OSVERSIONINFO (section 2.2.3.10.1) with well-known version-specific dwBuildNumber values.<2>
<2> Section 1.7: The values of the dwBuildNumber member in the OSVERSIONINFO structure (section 2.2.3.10.1) for specific versions of Windows are shown in the table that follows.
<table>
<tr><th>Version</th><th>dwBuildNumber value</th></tr>
<tr><td>Windows Server operating system</td><td>>= 16299</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><td>Windows 10 and Windows Server 2016</td><td>>= 10586</td></tr>
<tr><td>…</td><td>…</td></tr>
</table>

In Section 2.2.3.10.1, OSVERSIONINFO, changed from:

dwBuildNumber (4 bytes): The build number of the OS. This is a version-specific value.<168>
<168> Section 2.2.3.10.1: The values of the dwBuildNumber member in the OSVERSIONINFO structure (section 2.2.3.10.1) for specific versions of Windows is shown in the table that follows. On Windows Vista and later, an error is returned if the value is less than that shown in the table.

Changed to:

dwBuildNumber (4 bytes): The build number of the OS. This is a version-specific value.<168>
<168> Section 2.2.3.10.1: The values of the dwBuildNumber member in the OSVERSIONINFO structure (section 2.2.3.10.1) for specific versions of Windows are listed in the product behavior note for dwBuildNumber in Versioning and Capability Negotiation (section 1.7).

In Section 3.1.4.1.8.8, SPLCLIENT_CONTAINER Parameters, changed from:

pClientInfo: This parameter is a pointer to an SPLCLIENT_CONTAINER (section 2.2.1.2.14) structure that specifies client information. The Level member of the SPLCLIENT_CONTAINER structure MUST be 0x00000001.<245> The value of the dwBuildNum member is used to verify that the client OS version is valid. It is a version-specific number.<246>
<246> Section 3.1.4.1.8.8: The values of the dwBuildNumber member in the OSVERSIONINFO structure (section 2.2.3.10.1) for specific versions of Windows are listed in the product behavior note for dwBuildNumber in Versioning and Capability Negotiation (section 1.7).

Changed to:

pClientInfo: This parameter is a pointer to an SPLCLIENT_CONTAINER (section 2.2.1.2.14) structure that specifies client information. The Level member of the SPLCLIENT_CONTAINER structure MUST be 0x00000001.<245> The value of the dwBuildNum member is used to verify that the client OS version is valid. It is a version-specific number.<246>
<246> Section 3.1.4.1.8.8: The values of the dwBuildNumber member in the OSVERSIONINFO structure (section 2.2.3.10.1) for specific versions of Windows are listed in the product behavior note for dwBuildNumber in Versioning and Capability Negotiation (section 1.7).
On Windows Vista and later, an error is returned if the value is less than that shown for the corresponding Windows version in the table.</td></tr>
<tr><td>2018/10/29</td><td>In Section 2.2.3.10.1, OSVERSIONINFO, the description of dwBuildNumber has been changed from:</td></tr>
</table>

Errata Published*	Description
	dwBuildNumber (4 bytes): The build number of the OS.<168>. Changed to: dwBuildNumber (4 bytes): The build number of the OS. This SHOULD<168> be a version-specific value. In Section 3.1.4.1.8.8, SPLCLIENT_CONTAINER Parameters, the following has been changed from: pClientInfo: This parameter is a pointer to an SPLCLIENT_CONTAINER (section 2.2.1.2.14) structure that specifies client information. The Level member of the SPLCLIENT_CONTAINER structure MUST be 0x00000001.<245> Changed to: pClientInfo: This parameter is a pointer to an SPLCLIENT_CONTAINER (section 2.2.1.2.14) structure that specifies client information. The Level member of the SPLCLIENT_CONTAINER structure MUST be 0x00000001.<245> The dwBuildNum member is used to verify that the client OS version is valid. It SHOULD<246> be a version-specific number. In Section 7, Appendix B: Product Behavior, the following behavior notes have been changed. Changed from: <168> Section 2.2.3.10.1: The dwBuildNumber value for OSVERSIONINFO and OSVERSIONINFOEX for specific versions of Windows is shown in the table that follows. Changed to: <168> Section 2.2.3.10.1: The dwBuildNumber value for OSVERSIONINFO and OSVERSIONINFOEX for specific versions of Windows is shown in the table that follows. On Windows Vista and later, an error is returned if the value is less than that shown in the table. Changed from: <245> Section 3.1.4.1.8.8: Windows does not use the following members: pUserName, dwBuildNum, dwMajorVersion, dwMinorVersion, and wProcessorArchitecture. pMachineName is used only if the server cannot determine the client machine name using remote procedure call (RPC) functions. The pMachineName member can be NULL. Changed to: <245> Section 3.1.4.1.8.8: Windows does not use the following members: pUserName, dwMajorVersion, dwMinorVersion, and wProcessorArchitecture. pMachineName is used only if the server cannot determine the client machine name using remote procedure call (RPC) functions. The pMachineName member can be NULL. In that section a new behavior note 246 has been added:

Errata Published*	Description
	<246> Section 3.1.4.1.8.8: Windows version-specific values are listed in the product behavior note for dwBuildNumber in OSVERSIONINFO structure (section 2.2.3.10.1).

*Date format: YYYY/MM/DD

[MS-RRASM]: Routing and Remote Access Server (RRAS) Management Protocol

This topic lists the Errata found in [MS-RRASM] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

Errata below are for Protocol Document Version V23.0 – 2018/09/12.

Errata Published *	Description
2019/10/28	In Section 2.2.1.2.45, MIB_IPMCAST_OIF_STATS, changed dwIfNextHopIPAddr to dwNextHopAddr in the dwNextHopAddr field description. Changed from: ... dwNextHopAddr: Specifies the address of the next hop that corresponds to dwOutIfIndex. The dwOutIfIndex and dwIfNextHopIPAddr members uniquely identify a next hop on point-to-multipoint interfaces, where one interface connects to multiple networks. Examples of point-to-multipoint interfaces include non-broadcast multiple-access (NBMA) interfaces, and the internal interface on which all dial-up clients connect. For Ethernet and other broadcast interfaces, specify zero (0). Also specify zero (0) for point-to-point interfaces, which are identified by only dwOutIfIndex. ... Changed to: ... dwNextHopAddr: Specifies the address of the next hop that corresponds to dwOutIfIndex. The dwOutIfIndex and dwNextHopAddr members uniquely identify a next hop on point-to-multipoint interfaces, where one interface connects to multiple networks. Examples of point-to-multipoint interfaces include non-broadcast multiple-access (NBMA) interfaces, and the internal interface on which all dial-up clients connect. For Ethernet and other broadcast interfaces, specify zero (0). Also specify zero (0) for point-to-point interfaces, which are identified by only dwOutIfIndex. ... In Section 2.2.1.2.130, PPP_PROJECTION_INFO_1, changed dwAuthenticatedData to dwAuthenticationData in the dwAuthenticationData field description. Changed from: ... dwAuthenticationData: The same as dwAuthenticatedData in PPP_LCP_INFO. ... Changed to:

Errata Published *	Description
	... dwAuthenticationData: The same as dwAuthenticationData in PPP_LCP_INFO (see section 2.2.1.2.71). ... In Section 2.2.1.2.176, IGMP_MIB_GROUP_INFO, changed interface types RAS_SERVER to IGMP_IF_RAS_SERVER and RAS_CLIENT to IGMP_IF_RAS_CLIENT. Changed from: The IGMP_MIB_GROUP_INFO structure is used in the IGMP_MIB_IF_GROUPS_LIST (section 2.2.1.2.176) structure. If the interface is of type RAS_SERVER then the group membership of all the RAS clients is summarized, and the GroupUpTime and GroupExpiryTime is the maximum over all member RAS clients, while the V1HostPresentTimeLeft is set to 0. If the interface is of type RAS_CLIENT, the IpAddr is the next hop IP address of the RAS client. The membership is summarized over the RAS clients unless the IGMP_ENUM_FOR_RAS_CLIENTS_ID flag is set in Flags. ... Changed to: The IGMP_MIB_GROUP_INFO structure is used in the IGMP_MIB_IF_GROUPS_LIST (section 2.2.1.2.175) structure. If the interface is of type IGMP_IF_RAS_SERVER then the group membership of all the RAS clients is summarized, and the GroupUpTime and GroupExpiryTime is the maximum over all member RAS clients, while the V1HostPresentTimeLeft is set to 0. If the interface is of type IGMP_IF_RAS_CLIENT, the IpAddr is the next hop IP address of the RAS client. The membership is summarized over the RAS clients unless the IGMP_ENUM_FOR_RAS_CLIENTS_ID flag is set in Flags. ... In Section 2.2.1.2.181, IP_NAT_MIB_QUERY, changed instances of RMIBGetEntryFirst to RMIBEntryGetFirst. Changed from: The IP_NAT_MIB_QUERY structure is used to retrieve Network Address Translator (NAT) information and is passed to the following methods: • RMIBEntryGet (section 3.1.4.30) • RMIBGetEntryFirst (section 3.1.4.31) • RMIBEntryGetNext (section 3.1.4.32) .... Oid: This is an index of the NAT MIB. It MUST be one of the following values.

Errata Published *	Description

Value	Meaning
IP_NAT_INTERFACE_STATISTICS_OID 0x00000000	NAT interface statistics information is retrieved. When RMIBEntryGet, RMIBGetEntryFirst, and RMIBEntryGetNext return pMibOutEntry or pInfoStruct it MUST be typecast to IP_NAT_INTERFACE_STATISTICS (section 2.2.1.2.185).
IP_NAT_INTERFACE_MAPPING_TABLE_OID 0x00000001	NAT interface mapping table information. When RMIBEntryGet, RMIBGetEntryFirst, and RMIBEntryGetNext return pMibOutEntry or pInfoStruct it MUST be typecast to IP_NAT_ENUMERATE_SESSION_MAPPINGS (section 2.2.1.2.183).
IP_NAT_MAPPING_TABLE_OID 0x00000002	NAT mapping table information. Retrieves the session mappings of an interface. When RMIBEntryGet, RMIBGetEntryFirst, and RMIBEntryGetNext return pMibOutEntry or pInfoStruct it

Changed to:

The IP_NAT_MIB_QUERY structure is used to retrieve Network Address Translator (NAT) information and is passed to the following methods:

- RMIBEntryGet (section 3.1.4.30)
- RMIBEntryGetFirst (section 3.1.4.31)
- RMIBEntryGetNext (section 3.1.4.32)

...

Oid: This is an index of the NAT MIB. It MUST be one of the following values.

Value	Meaning
IP_NAT_INTERFACE_STATISTICS_OID 0x00000000	NAT interface statistics information is retrieved. When RMIBEntryGet, RMIBEntryGetFirst, and RMIBEntryGetNext return pMibOutEntry or pInfoStruct it MUST be typecast to IP_NAT_INTERFACE_STATISTICS (section 2.2.1.2.184).
IP_NAT_INTERFACE_MAPPING_TABLE_OID 0x00000001	NAT interface mapping table information. When RMIBEntryGet, RMIBEntryGetFirst, and RMIBEntryGetNext return pMibOutEntry or pInfoStruct it MUST be typecast to IP_NAT_ENUMERATE_SESSION_MAPPINGS (section 2.2.1.2.182).
IP_NAT_MAPPING_TABLE_OID 0x00000002	NAT mapping table information. Retrieves the session mappings of an interface. When RMIBEntryGet, RMIBEntryGetFirst, and RMIBEntryGetNext return pMibOutEntry or pInfoStruct it MUST be typecast to IP_NAT_ENUMERATE_SESSION_MAPPINGS.

In Section 2.2.1.2.260, BGP_POLICY, changed eType value from MatchMaxPrefix to MatchMaxPrefixes. And changed eAttrType values ModifyLocalPref to NewLocalPref, ModifyNextHop to NewNextHop, and ModifyMed to NewMed.

Changed from:

Errata Published *	Description
	... A BGP policy: • MUST NOT have more than one Match clause with eType in BGP_POLICY_MATCH set to MatchASNRange (0x3). • MUST NOT have more than one Match clause with eType in BGP_POLICY_MATCH set to MatchMaxPrefix (0x5). • MUST NOT have more than one modify Action clause with eAttrType in BGP_POLICY_MODIFY (section 2.2.1.2.259) set to ModifyLocalPref (0x3). • MUST NOT have more than one modify Action clause with eAttrType in BGP_POLICY_MODIFY set to ModifyNextHop (0x4). • MUST NOT have more than one modify Action clause with eAttrType in BGP_POLICY_MODIFY set to ModifyMed (0x5). • MUST have only one Action clause with bDeny in BGP_POLICY_ACTION set to TRUE when a Match clause with eType in BGP_POLICY_MATCH is specified as MatchMaxPrefix (0x5). Changed to: ... A BGP policy: • MUST NOT have more than one Match clause with eType in BGP_POLICY_MATCH set to MatchASNRange (0x3). • MUST NOT have more than one Match clause with eType in BGP_POLICY_MATCH set to MatchMaxPrefixes (0x5). • MUST NOT have more than one modify Action clause with eAttrType in BGP_POLICY_MODIFY (section 2.2.1.2.258) set to NewLocalPref (0x3). • MUST NOT have more than one modify Action clause with eAttrType in BGP_POLICY_MODIFY set to NewNextHop (0x4). • MUST NOT have more than one modify Action clause with eAttrType in BGP_POLICY_MODIFY set to NewMed (0x5). • MUST have only one Action clause with bDeny in BGP_POLICY_ACTION set to TRUE when a Match clause with eType in BGP_POLICY_MATCH is specified as MatchMaxPrefixes (0x5). In Section 3.1.4.44, RMprAdminServerSetInfo (Opnum 43), changed return value ERROR_REBOOT_REQUIRED to ERROR_SUCCESS_REBOOT_REQUIRED when the RRAS server completes the processing successfully. Changed from: ... When processing this call, the RRASM server MUST do the following: ... • If the RRAS server completes the processing successfully return either ERROR_SUCCESS or ERROR_REBOOT_REQUIRED<316> based on the impact of the configuration change as indicated by the RRAS server. Otherwise return the error status. ... Changed to: ... When processing this call, the RRASM server MUST do the following: ... • If the RRAS server completes the processing successfully return either ERROR_SUCCESS or ERROR_SUCCESS_REBOOT_REQUIRED<316> based on the impact of the configuration change as indicated by the RRAS server. Otherwise return the error status. ...

Errata Published *	Description
	In Section 3.1.4.48, RMprAdminServerSetInfoEx (Opnum 47), changed return value ERROR_REBOOT_REQUIRED to ERROR_SUCCESS_REBOOT_REQUIRED when the RRAS server completes the processing successfully. Changed from: ... When processing this call, the RRASM server MUST do the following: ... ● If the RRAS server completes the processing successfully, it MUST return either ERROR_SUCCESS, ERROR_REBOOT_REQUIRED<321>, or ERROR_RESTART_REQUIRED<322> based on the impact of the configuration change. Otherwise return the error status. ... Changed to: ... When processing this call, the RRASM server MUST do the following: ... ● If the RRAS server completes the processing successfully, it MUST return either ERROR_SUCCESS, ERROR_SUCCESS_REBOOT_REQUIRED<321>, or ERROR_RESTART_REQUIRED<322> based on the impact of the configuration change. Otherwise return the error status. ... In Section 3.4.4.5 RasRpcSubmitRequest (Opnum 12), changed instances of GetDevConfig to GetDevConfigStruct when describing client behavior for the ReqType REQTYPE_GETDEVCONFIG. Changed from: ... REQTYPE_GETDEVCONFIG Before calling the method, the client MUST set the GetDevConfig.size value to the size of the GetDevConfig.config buffer. If the returned GetDevConfig.retcode is set to ERROR_BUFFER_TOO_SMALL (0x0000025B), the buffer that was passed in was not big enough to hold the device configuration information. The client SHOULD again call the API with GetDevConfig.size set to the size of returned GetDevConfig.size. ... Changed to: ... REQTYPE_GETDEVCONFIG Before calling the method, the client MUST set the GetDevConfigStruct.size value to the size of the GetDevConfigStruct.config buffer. If the returned GetDevConfigStruct.retcode is set to ERROR_BUFFER_TOO_SMALL (0x0000025B), the buffer that was passed in was not big enough to hold the device configuration information. The client SHOULD again call the API with GetDevConfigStruct.size set to the size of returned GetDevConfigStruct.size. ... In Section 7, Appendix B: Product Behavior, changed the return value ERROR_REBOOT_REQUIRED

Errata Published *	Description
	to ERROR_SUCCESS_REBOOT_REQUIRED in product behavior note <316> when the configuration change requires a reboot of the machine for the settings to be applied. Changed from: <316> Section 3.1.4.44: Windows will return the error value ERROR_REBOOT_REQUIRED when the configuration change requires a reboot of the machine for the settings to be applied. One such implementation requirement is when the number of ports configured is more than the maximum number of ports that the tunneling protocols are configured to support initially. Changed to: <316> Section 3.1.4.44: Windows will return the error value ERROR_SUCCESS_REBOOT_REQUIRED when the configuration change requires a reboot of the machine for the settings to be applied. One such implementation requirement is when the number of ports configured is more than the maximum number of ports that the tunneling protocols are configured to support initially. In this document, numerous editorial fixes have also been made, e.g., changed instances of "Ipv6" and "IPv6" to "IPV6"; changed instances of "GetDevConfig" to "GetDevConfigStruct"; updated hexadecimal syntax to USHORT 16-bit format; and also added section numbers to programming elements where applicable. Sections updated: 2.2.1.2.103 2.2.1.2.104 2.2.1.2.134 2.2.1.2.136 2.2.1.2.156 2.2.1.2.158 2.2.2.2.79 2.2.5.1.1 3.1.4.30 3.1.4.31 3.1.4.33 3.1.4.38 3.1.4.44 3.3.4.5 7 - the following product behavior notes were upated: <266> <268> <272> <290> <293> <298> <305>
2019/10/28	In Section 2.2.1.2.37 MIB_IPMCAST_BOUNDARY, added names of dwStatus values in the table.

Errata Published *	Description
	Changed from: dwStatus: A status value that describes the current status of this entry in a mullticast forwarding entry (MFE) boundary table. Value Meaning 0x00000001 The entry has an active status. 0x00000002 The entry has a notInService status. 0x00000003 The entry has a notReady status. 0x00000004 The entry has a createAndGo status. 0x00000005 The entry has a createAndWait status. 0x00000006 The entry has a destroy status. Changed to: dwStatus: A status value that describes the current status of this entry in a multicast forwarding entry (MFE) boundary table. Value Meaning ROWSTATUS_ACTIVE 0x00000001 The entry has an active status. ROWSTATUS_NOTINSERVICE 0x00000002 The entry has a notInService status. ROWSTATUS_NOTREADY 0x00000003 The entry has a notReady status. ROWSTATUS_CREATEANDGO 0x00000004 The entry has a createAndGo status. ROWSTATUS_CREATEANDWAIT 0x00000005 The entry has a createAndWait status. ROWSTATUS_DESTROY 0x00000006 The entry has a destroy status. Section 2.2.1.2.105 IPX_MIB_INDEX, added missing value 3 in the table.

Errata Published *	Description
	Changed from: TableId: Specifies the type of table. Values MUST be one of the following values. Value Meaning IPX_BASE_ENTRY 0x00000000 IPX base. See IPXMIB_BASE (section 2.2.1.2.107). IPX_INTERFACE_TABLE 0x00000001 IPX interface table. See IPX_INTERFACE (section 2.2.1.2.109). IPX_DEST_TABLE 0x00000002 IPX destination table. See IPX_ROUTE (section 2.2.1.2.110). IPX_SERV_TABLE 0x00000004 IPX service table. See IPX_SERVICE (section 2.2.1.2.121). IPX_STATIC_SERV_TABLE 0x00000005 IPX static service table. See IPX_STATIC_SERVICE_INFO (section 2.2.1.2.95). Changed to: TableId: Specifies the type of table. Values MUST be one of the following values. Value Meaning IPX_BASE_ENTRY 0x00000000 IPX base. See IPXMIB_BASE (section 2.2.1.2.106). IPX_INTERFACE_TABLE 0x00000001 IPX interface table. See IPX_INTERFACE (section 2.2.1.2.108). IPX_DEST_TABLE 0x00000002 IPX destination table. See IPX_ROUTE (section 2.2.1.2.109). IPX_STATIC_ROUTE_TABLE 0x00000003 IPX Static Route Table. See IPX_STATIC_ROUTE_INFO (section 2.2.1.2.93). IPX_SERV_TABLE

Errata Published *	Description
	0x00000004 IPX service table. See IPX_SERVICE (section 2.2.1.2.120). IPX_STATIC_SERV_TABLE 0x00000005 IPX static service table. See IPX_STATIC_SERVICE_INFO (section 2.2.1.2.94). Section 2.2.1.2.177 IGMP_MIB_GROUP_INFO, updated names of values in the introduction: RAS_SERVER to IGMP_IF_RAS_SERVER, RAS_CLIENT to IGMP_IF_RAS_CLIENT, and IGMP_ENUM_FOR_RAS_CLIENTS_ID to IGMP_ENUM_FOR_RAS_CLIENTS. Changed from: The IGMP_MIB_GROUP_INFO structure is used in the IGMP_MIB_IF_GROUPS_LIST (section 2.2.1.2.176) structure. If the interface is of type RAS_SERVER then the group membership of all the RAS clients is summarized, and the GroupUpTime and GroupExpiryTime is the maximum over all member RAS clients, while the V1HostPresentTimeLeft is set to 0. If the interface is of type RAS_CLIENT, the IpAddr is the next hop IP address of the RAS client. The membership is summarized over the RAS clients unless the IGMP_ENUM_FOR_RAS_CLIENTS_ID flag is set in Flags. Changed to: The IGMP_MIB_GROUP_INFO structure is used in the IGMP_MIB_IF_GROUPS_LIST (section 2.2.1.2.175) structure. If the interface is of type IGMP_IF_RAS_SERVER then the group membership of all the RAS clients is summarized, and the GroupUpTime and GroupExpiryTime is the maximum over all member RAS clients, while the V1HostPresentTimeLeft is set to 0. If the interface is of type IGMP_IF_RAS_CLIENT, the IpAddr is the next hop IP address of the RAS client. The membership is summarized over the RAS clients unless the IGMP_ENUM_FOR_RAS_CLIENTS_ID flag is set in Flags. Section 2.2.1.2.178 IGMP_MIB_IF_STATS, in the LastQuerierChangeTime description changed member name from igmpInterfaceQuerier to QuerierIpAddr. Changed from: LastQuerierChangeTime: The number of seconds since igmpInterfaceQuerier was last changed. Changed to: LastQuerierChangeTime: The number of seconds since QuerierIpAddr was last changed. Section 2.2.1.2.179 IGMP_MIB_GROUP_SOURCE_INFO_V3, added section. Adjusted references and reference numbers 2.2.1.2.180 to 2.2.1.2.271 throughout to compensate for section number changes. Changed from: (missing section) Changed to: The IGMP_MIB_GROUP_SOURCE_INFO_V3 structure provides information about each source IP endpoint. typedef struct _IGMP_MIB_GROUP_SOURCE_INFO_V3 {

Errata Published *	Description
	DWORD Source; DWORD SourceExpiryTime; DWORD SourceUpTime; DWORD Flags; } IGMP_MIB_GROUP_SOURCE_INFO_V3, *PIGMP_MIB_GROUP_SOURCE_INFO_V3; Source: IP endpoint address of a source. SourceExpiryTime: The time, in seconds, that remains before source expires. Not valid for exclusion mode. SourceUpTime: The time, in seconds since the source was up. Flags: Reserved. This is unused and SHOULD be NULL, or MAY be set to 0x00000000. Section 2.2.1.2.180 IGMP_MIB_GROUP_INFO_V3, for Sources array of IGMP_MIB_GROUP_SOURCE_INFO_V3 added reference to 2.2.1.2.179. Changed from: NumSources: The number of entries of IGMP_MIB_GROUP_SOURCE_INFO_V3. Sources: The IGMP_MIB_GROUP_SOURCE_INFO_V3 structure. Changed to: NumSources: The number of entries of IGMP_MIB_GROUP_SOURCE_INFO_V3. Sources: The IGMP_MIB_GROUP_SOURCE_INFO_V3 structure (section 2.2.1.2.179). 6 Appendix A: Full IDL, moved location of struct IGMP_MIB_GROUP_SOURCE_INFO_V3 to before struct IGMP_MIB_GROUP_INFO_V3. Changed from: typedef struct _IPRIP_PEER_STATS { DWORD PS_LastPeerRouteTag; DWORD PS_LastPeerUpdateTickCount; DWORD PS_LastPeerUpdateVersion; DWORD PS_BadResponsePacketsFromPeer;

Errata Published *	Description

	DWORD PS_BadResponseEntriesFromPeer;
	} IPRIP_PEER_STATS, *PIPRIP_PEER_STATS;
	typedef struct _IGMP_MIB_GROUP_SOURCE_INFO_V3 {
	DWORD Source;
	DWORD SourceExpiryTime; //not valid for exclusion mode
	DWORD SourceUpTime;
	DWORD Flags;
	} IGMP_MIB_GROUP_SOURCE_INFO_V3, *PIGMP_MIB_GROUP_SOURCE_INFO_V3;
	typedef struct _IGMP_MIB_GET_INPUT_DATA {
	DWORD TypeId;
	USHORT Flags;
	USHORT Signature;
	DWORD IfIndex;
	DWORD RasClientAddr;
	DWORD GroupAddr;
	DWORD Count;
	} IGMP_MIB_GET_INPUT_DATA, *PIGMP_MIB_GET_INPUT_DATA;
	Changed to:
	typedef struct _IGMP_MIB_GROUP_IFS_LIST {
	DWORD GroupAddr;
	DWORD NumInterfaces;
	BYTE Buffer[1];
	} IGMP_MIB_GROUP_IFS_LIST, *PIGMP_MIB_GROUP_IFS_LIST;

Errata Published *	Description

	typedef struct _IGMP_MIB_GROUP_SOURCE_INFO_V3 {
	DWORD Source;
	DWORD SourceExpiryTime; //not valid for exclusion mode
	DWORD SourceUpTime;
	DWORD Flags;
	} IGMP_MIB_GROUP_SOURCE_INFO_V3, *PIGMP_MIB_GROUP_SOURCE_INFO_V3;
	typedef struct _IGMP_MIB_GROUP_INFO_V3
	{
	union {
	DWORD IfIndex;
	DWORD GroupAddr;
	};
	DWORD IpAddr;
	DWORD GroupUpTime;
	DWORD GroupExpiryTime;
	DWORD LastReporter;
	DWORD V1HostPresentTimeLeft;
	DWORD Flags;
	//v3 additions
	DWORD Version; //1/2/3
	DWORD Size; //size of this struct
	DWORD FilterType;//EXCLUSION/INCLUSION
	DWORD V2HostPresentTimeLeft;

Errata Published *	Description
	DWORD NumSources; //IGMP_MIB_GROUP_SOURCE_INFO_V3 Sources[0]; } IGMP_MIB_GROUP_INFO_V3, *PIGMP_MIB_GROUP_INFO_V3;

*Date format: YYYY/MM/DD

[MS-RRP]: Windows Remote Registry Protocol

This topic lists the Errata found in the MS-RRP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 12, 2018 - Download

March 4, 2020 - Download

Errata below are for Protocol Document Version V35.0 – 2020/03/04.

Errata Published*	Description
2020/04/27	In Section 3.1.5.7, BaseRegCreateKey (Opnum 6), we corrected hKEY to hKey in the explanatory text. Changed from: The server then checks to see if the key specified by the hKEY parameter is a key that can only be operated on in the 64-bit key namespace (KEYS64). See section 3.1.1.4. Changed to: The server then checks to see if the key specified by the hKey parameter is a key that can only be operated on in the 64-bit key namespace (KEYS64). See section 3.1.1.4. In Section 3.1.5.15, BaseRegOpenKey (Opnum 15), we corrected hKEY to hKey in the explanatory text. Changed from: The server then checks to see if the key specified by the hKEY parameter is a key that can only be operated on in the 64-bit key namespace (KEYS64). See section 3.1.1.4. Changed to: The server then checks to see if the key specified by the hKey parameter is a key that can only be operated on in the 64-bit key namespace (KEYS64). See section 3.1.1.4. In Section 3.1.5.22, BaseRegSetValue (Opnum 22), we corrected hKEY to hKey in the explanatory text. Changed from: If the key specified by hKEY has a KEYTYPE of symbolic link and lpValueName is specified to any

Errata Published*	Description
	string other than "SymbolicLinkValue", the server MUST fail the method and return ERROR_ACCESS_DENIED. ... The server MUST determine if the key path indicated by hKey refers to a path that is within the list of paths for which updates to either the 32-bit or 64-bit namespaces are copied into the 64-bit or 32-bit namespace, respectively, as specified in section 3.1.1.4. If the key indicated by hKey is within one of the paths, the server MUST set the UPDATECOPY column of the HANDLETABLE for the row indicated by hKEY to TRUE. This indicates that the value is copied between the 32-bit and 64-bit key namespaces when the handle is closed. ... The server MUST set the KEYISMODIFIED property of the key indicated by hKEY to TRUE. Changed to: If the key specified by hKey has a KEYTYPE of symbolic link and lpValueName is specified to any string other than "SymbolicLinkValue", the server MUST fail the method and return ERROR_ACCESS_DENIED. ... The server MUST determine if the key path indicated by hKey refers to a path that is within the list of paths for which updates to either the 32-bit or 64-bit namespaces are copied into the 64-bit or 32-bit namespace, respectively, as specified in section 3.1.1.4. If the key indicated by hKey is within one of the paths, the server MUST set the UPDATECOPY column of the HANDLETABLE for the row indicated by hKey to TRUE. This indicates that the value is copied between the 32-bit and 64-bit key namespaces when the handle is closed. ... The server MUST set the KEYISMODIFIED property of the key indicated by hKey to TRUE. In Section 3.1.5.31, BaseRegDeleteKeyEx (Opnum 35), we corrected hKEY to hKey in the explanatory text. Changed from: The server MUST then check to see if the key specified by the hKEY parameter is a key that can only be operated on in the 64-bit key namespace (KEYS64). See section 3.1.1.4. Changed to: The server MUST then check to see if the key specified by the hKey parameter is a key that can only be operated on in the 64-bit key namespace (KEYS64). See section 3.1.1.4. In Section 3.1.5.26, BaseRegQueryMultipleValues (Opnum 29), we corrected valListOut to val_listOut and valListIn to val_listIn in the explanatory text.

Errata Published*	Description
	Changed from: If any one of the parameters ldwTotsize and valListOut is NULL, the server MUST return ERROR_INVALID_PARAMETER. ... For each of the RVALENT structures returned by calling parameter valListIn: if the return value is greater than zero and the buffer is NULL, the server MUST return ERROR_INVALID_PARAMETER Changed to: If any one of the parameters ldwTotsize and val_listOut is NULL, the server MUST return ERROR_INVALID_PARAMETER. ... For each of the RVALENT structures returned by calling parameter val_listIn: if the return value is greater than zero and the buffer is NULL, the server MUST return ERROR_INVALID_PARAMETER In Section 3.1.5.30, BaseRegQueryMultipleValues2 (Opnum 34), we corrected valListOut to val_listOut and valListIn to val_listIn in the explanatory text. Changed from: If any one of the parameters ldwTotsize, ldwRequiredSize, and valListOut is NULL, the server MUST return ERROR_INVALID_PARAMETER. ... For each of the RVALENT structures returned by calling the valListIn parameter: if the return value is greater than zero and the buffer is NULL, the server MUST return ERROR_INVALID_PARAMETER Changed to: If any one of the parameters ldwTotsize, ldwRequiredSize, and val_listOut is NULL, the server MUST return ERROR_INVALID_PARAMETER. ... For each of the RVALENT structures returned by calling the val_listIn parameter: if the return value is greater than zero and the buffer is NULL, the server MUST return ERROR_INVALID_PARAMETER In Sections 3.1.5.10, BaseRegEnumKey (Opnum 9), 3.1.5.11 BaseRegEnumValue (Opnum 10), 3.1.5.14 BaseRegLoadKey (Opnum 13), and 3.1.5.22 BaseRegSetValue (Opnum 22), we corrected links to the top-level Section 3.1.1 to more appropriate child sections for key and value names.

Errata Published*	Description
	In Section 3.1.5.16, BaseRegQueryInfoKey (Opnum 16), we corrected lpcSubkeys to lpcSubKeys in the explanatory text. Changed from: The server MUST return a pointer to the variable that contains the number of subkeys for the specified key in the lpcSubkeys parameter. If there are no subkeys under the key indicated by hKey, the server MUST set this value to 0. Changed to: The server MUST return a pointer to the variable that contains the number of subkeys for the specified key in the lpcSubKeys parameter. If there are no subkeys under the key indicated by hKey, the server MUST set this value to 0.

*Date format: YYYY/MM/DD

[MS-RSMC]: Remote Session Monitoring and Control Protocol

This topic lists the Errata found in [MS-RSMC] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

[MS-RSVD]: Remote Shared Virtual Disk Protocol

This topic lists the Errata found in [MS-RSVD] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

September 26, 2016 - Download

March 20, 2017 - Download

June 1, 2017 - Download

September 15, 2017 - Download

December 1, 2017 - Download

[MS-SAMR]: Security Account Manager (SAM) Remote Protocol (Client-to-Server)

This topic lists the Errata found in [MS-SAMR] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

RSS
Atom

Errata are subject to the same terms as the Open Specifications documentation referenced.

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

September 15, 2017 - Download

September 12, 2018 - Download

Errata below are for Protocol Document Version V40.0 – 2018/09/12.

Errata Published*	Description
2019/05/27	In Section 2.1, Transport, changed from: The server SHOULD use this identity to perform method-specific access checks, as specified in the message processing section of each method.<11> The server SHOULD<12> reject calls that do not use an authentication level of either RPC_C_AUTHN_LEVEL_NONE or RPC_C_AUTHN_LEVEL_PKT_PRIVACY (see [MS-RPCE] section 2.2.1.1.8). Changed to: The server SHOULD use this identity to perform method-specific access checks, as specified in the message processing section of each method.<11> RPC clients for this protocol MUST use authentication level RPC_C_AUTHN_LEVEL_NONE when invoking RPC over SMB methods. The server SHOULD<12> reject calls that do not use an authentication level of either RPC_C_AUTHN_LEVEL_NONE or RPC_C_AUTHN_LEVEL_PKT_PRIVACY (see [MS-RPCE] section 2.2.1.1.8).

*Date format: YYYY/MM/DD

[MS-SAMS]: Security Account Manager (SAM) Remote Protocol (Server-to-Server)

This topic lists the Errata found in the MS-KPP document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-SCMR]: Service Control Manager Remote Protocol

This topic lists the Errata found in [MS-SCMR] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 12, 2018 - Download

September 23, 2019 - Download

[MS-SHLLINK]: Shell Link (.LNK) Binary File Format

This topic lists the Errata found in [MS-SHLLINK] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 12, 2018 - Download

[MS-SFMWA]: Server and File Management Web APIs

This topic lists the Errata found in [MS-SFMWA] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

March 20, 2017 - Download

[MS-SFU]: Kerberos Protocol Extensions Service for User and Constrained Delegation Protocol

This topic lists the Errata found in the MS-SFU document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

March 4, 2020 - Download

Errata below are for Protocol Document Version V18.0 – 2020/03/04.

Errata Published*	Description
2020/03/30	In Section 3.2.5.2.1, Using ServicesAllowedToSendForwardedTicketsTo, changed the secondary check to state that padata type does not have the resource-based constrained delegation bit set for the return values. Changed from: If the service ticket in the additional-tickets field is not set to forwardable and the PA-PACOPTIONS [167] ([MS-KILE] section 2.2.10) padata type has the resource-based constrained delegation bit set, then the KDC MUST return KRB-ERR-BADOPTION with STATUS_NO_MATCH. Changed to: If the service ticket in the additional-tickets field is not set to forwardable and the PA-PACOPTIONS [167] ([MS-KILE] section 2.2.10) padata type does not have the resource-based constrained delegation bit set, then the KDC MUST return KRB-ERR-BADOPTION with STATUS_NO_MATCH. In Section 3.2.5.2.3, Using ServicesAllowedToReceiveForwardedTicketsFrom, removed the first check for the KDC for service 1. Changed from: If this is the KDC for Service 1, and the service ticket in the additional-tickets field is not set to forwardable, and the USER_NOT_DELEGATED bit is set in the UserAccountControl field in the KERB_VALIDATION_INFO structure ([MS-PAC] section 2.5), then the KDC MUST… Changed to: If the service ticket in the additional-tickets field is not set to forwardable, and the USER_NOT_DELEGATED bit is set in the UserAccountControl field in the KERB_VALIDATION_INFO structure ([MS-PAC] section 2.5), then the KDC MUST…
2020/03/30	In Section 3.2.5.2.4, KDC Replies with Service Ticket, the source of the cname and crealm has been added.

Errata Published*	Description
	Changed from: The KDC MUST also add the name of Service 1 to the S4UTransitedServices list in the structure. Changed to: The KDC MUST also add the name of Service 1 to the S4UTransitedServices list in the structure. Windows KDC constructs the impersonated client's principal name from the PAC. The cname and crealm in the KDC reply are set to the impersonated client's principal name, realm.

*Date format: YYYY/MM/DD

[MS-SMB]: Server Message Block (SMB) Protocol

This topic lists the Errata found in [MS-SMB] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

June 1, 2017 - Download

December 1, 2017 - Download

September 12, 2018 - Download

[MS-SMB2]: Server Message Block (SMB) Protocol Versions 2 and 3

This topic lists the Errata found in [MS-SMB2] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

March 4, 2020 - Download

Rollup of Errata from March 5, 2020 – August 3, 2020 - Download

Errata below are for Protocol Document Version V60.0 – 2020/03/04.

Errata Published*	Description
2020/08/03	In Section 3.3.4.13, Server Application Registers a Share the following was changed from: • Share.Type MUST be set to shi503_type. The server SHOULD<212> set STYPE_CLUSTER_FS, STYPE_CLUSTER_SOFS, and STYPE_CLUSTER_DFS in an implementation-defined manner. Changed to: • Share.Type MUST be set to shi503_type. The server SHOULD<212> set STYPE_CLUSTER_FS, STYPE_CLUSTER_SOFS, and STYPE_CLUSTER_DFS as specified in [MS-SRVS] section 2.2.2.4 in an implementation-defined manner. In Section 3.3.5.7, Receiving an SMB2 TREE_CONNECT Request the following was changed from: ... If Share.Type includes STYPE_CLUSTER_FS, STYPE_CLUSTER_SOFS, or STYPE_CLUSTER_DFS and Connection.Dialect is greater than MaxClusterDialect and SMB2_TREE_CONNECT_FLAG_CLUSTER_RECONNECT is not set in Flags/Reserved field, the server MUST fail the request with STATUS_SMB_BAD_CLUSTER_DIALECT (0xC05D0001) and if Connection.Dialect is SMB 3.1.1, the server MUST return error data as specified in section 2.2.2 with ByteCount set to 10, ErrorContextCount set to 1, and ErrorData set to SMB2 ERROR Context response formatted as ErrorDataLength set to 2, ErrorId set to 0, and ErrorData set to MaxClusterDialect; otherwise, the server MUST return error data as specified in section 2.2.2 with ByteCount set to 2 and ErrorContextData set to MaxClusterDialect. Changed to: ... If Share.Type is STYPE_CLUSTER_FS, STYPE_CLUSTER_SOFS, or STYPE_CLUSTER_DFS as specified in [MS-SRVS] section 2.2.2.4 and Connection.Dialect is greater than MaxClusterDialect and SMB2_TREE_CONNECT_FLAG_CLUSTER_RECONNECT is not set in Flags/Reserved field, the server MUST fail the request with STATUS_SMB_BAD_CLUSTER_DIALECT (0xC05D0001) and if Connection.Dialect is SMB 3.1.1, the server MUST return error data as specified in section 2.2.2 with ByteCount set to 10, ErrorContextCount set to 1, and ErrorData set to SMB2 ERROR Context response formatted as ErrorDataLength set to 2, ErrorId set to 0, and ErrorData set to

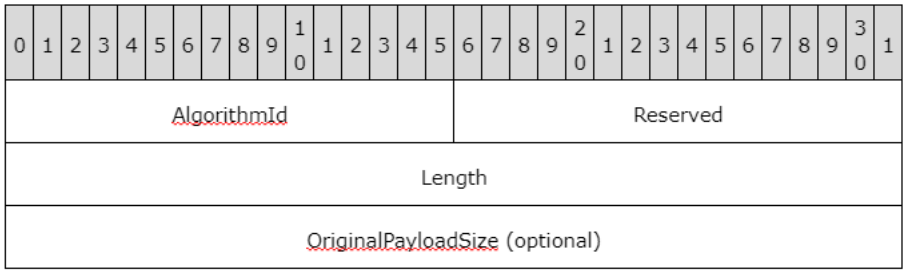
Errata Published*	Description
	MaxClusterDialect; otherwise, the server MUST return error data as specified in section 2.2.2 with ByteCount set to 2 and ErrorContextData set to MaxClusterDialect. . Changed from: If TreeConnect.Share.Type includes STYPE_CLUSTER_SOFS, Connection.Dialect is "3.1.1" and the SMB2 _TREE_CONNECT_FLAG_REDIRECT_TO_OWNER bit is set in the Flags field of the SMB2 TREE_CONNECT request, the server MUST query the underlying object store in an implementation-specific manner to determine whether the share is hosted on this node. If not, the server MUST fail the tree connect request by setting the Status field in SMB2 header to STATUS_BAD_NETWORK_NAME, return error data as specified in section 2.2.2 with ErrorData set to SMB2 ERROR Context response formatted as ErrorId set to SMB2_ERROR_ID_SHARE_REDIRECT, and ErrorContextData set to the Share Redirect error context data as specified in section 2.2.2.2.2 with IPAddrMoveList set to the list of IP addresses determined for where to access the share. Changed to: If TreeConnect.Share.Type is STYPE_CLUSTER_SOFS as specified in [MS-SRVS] section 2.2.2.4, Connection.Dialect is "3.1.1" and the SMB2 _TREE_CONNECT_FLAG_REDIRECT_TO_OWNER bit is set in the Flags field of the SMB2 TREE_CONNECT request, the server MUST query the underlying object store in an implementation-specific manner to determine whether the share is hosted on this node. If not, the server MUST fail the tree connect request by setting the Status field in SMB2 header to STATUS_BAD_NETWORK_NAME, return error data as specified in section 2.2.2 with ErrorData set to SMB2 ERROR Context response formatted as ErrorId set to SMB2_ERROR_ID_SHARE_REDIRECT, and ErrorContextData set to the Share Redirect error context data as specified in section 2.2.2.2.2 with IPAddrMoveList set to the list of IP addresses determined for where to access the share. Changed from: ● If Connection.Dialect belongs to the SMB 3.x dialect family and TreeConnect.Share.Type includes STYPE_CLUSTER_SOFS, the server MUST set the SMB2_SHARE_CAP_SCALEOUT bit in the Capabilities field. ● If Connection.Dialect belongs to the SMB 3.x dialect family and TreeConnect.Share.Type includes STYPE_CLUSTER_FS, STYPE_CLUSTER_SOFS, or STYPE_CLUSTER_DFS the server MUST set the SMB2_SHARE_CAP_CLUSTER bit in the Capabilities field. ● If Connection.Dialect is "3.0.2" or "3.1.1", TreeConnect.Share.Type includes STYPE_CLUSTER_SOFS, and TreeConnect.Share is asymmetric, the server MUST set the SMB2_SHARE_CAP_ASYMMETRIC bit in the Capabilities field. ● If Connection.Dialect is "3.1.1" and TreeConnect.Share.SupportsIdentityRemoting is set, the server MUST set the SMB2_SHAREFLAG_IDENTITY_REMOTING bit in the ShareFlags field of the SMB2 TREE_CONNECT response. ● If Connection.Dialect is "3.1.1", TreeConnect.Share.Type includes STYPE_CLUSTER_SOFS, and the SMB2_TREE_CONNECT_FLAG_REDIRECT_TO_OWNER bit is set in the Flags field of the SMB2 TREE_CONNECT request and the SMB2_SHARE _CAP_ASYMMETRIC bit is set in the Capabilities field, the server SHOULD<259> set the SMB2_SHARE_CAP_REDIRECT_TO_OWNER bit in the Capabilities field.

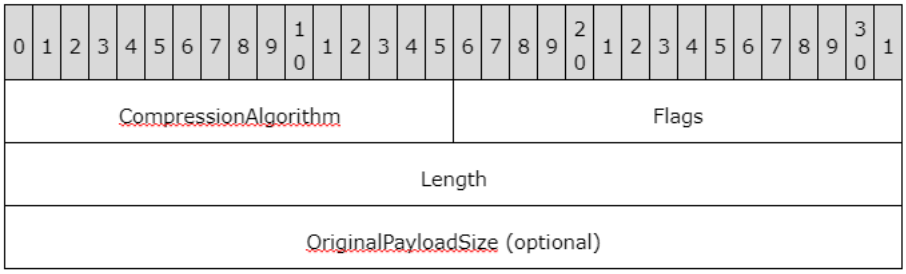
Errata Published*	Description
	Changed to: • If Connection.Dialect belongs to the SMB 3.x dialect family and TreeConnect.Share.Type is STYPE_CLUSTER_SOFS as specified in [MS-SRVS] section 2.2.2.4, the server MUST set the SMB2_SHARE_CAP_SCALEOUT bit in the Capabilities field. • If Connection.Dialect belongs to the SMB 3.x dialect family and TreeConnect.Share.Type is STYPE_CLUSTER_FS, STYPE_CLUSTER_SOFS, or STYPE_CLUSTER_DFS as specified in [MS-SRVS] section 2.2.2.4, the server MUST set the SMB2_SHARE_CAP_CLUSTER bit in the Capabilities field. • If Connection.Dialect is "3.0.2" or "3.1.1", TreeConnect.Share.Type is STYPE_CLUSTER_SOFS as specified in [MS-SRVS] section 2.2.2.4, and TreeConnect.Share is asymmetric, the server MUST set the SMB2_SHARE_CAP_ASYMMETRIC bit in the Capabilities field. • If Connection.Dialect is "3.1.1" and TreeConnect.Share.SupportsIdentityRemoting is set, the server MUST set the SMB2_SHAREFLAG_IDENTITY_REMOTING bit in the ShareFlags field of the SMB2 TREE_CONNECT response. • If Connection.Dialect is "3.1.1", TreeConnect.Share.Type sis STYPE_CLUSTER_SOFS as specified in [MS-SRVS] section 2.2.2.4, and the SMB2_TREE_CONNECT_FLAG_REDIRECT_TO_OWNER bit is set in the Flags field of the SMB2 TREE_CONNECT request and the SMB2_SHARE _CAP_ASYMMETRIC bit is set in the Capabilities field, the server SHOULD<259> set the SMB2_SHARE_CAP_REDIRECT_TO_OWNER bit in the Capabilities field. ... In Section 3.3.5.9, Receiving an SMB2 CREATE Request the following has been changed from: • The Treeconnect.Share.Type is STYPE_DISKTREE Changed to: • The Treeconnect.Share.Type is STYPE_DISKTREE as specified in [MS-SRVS] section 2.2.2.4. Changed from: For open requests on a share of type STYPE_DISKTREE (as indicated by TreeConnect.Share.Type) the server MUST do the following: Changed to: If TreeConnect.Share.Type is STYPE_DISKTREE as specified in [MS-SRVS] section 2.2.2.4, the server MUST do the following: Changed from: If Connection.Dialect belongs to the SMB 3.x dialect family, TreeConnect.Share.Type includes STYPE_CLUSTER_SOFS and the RequestedOplockLevel is SMB2_OPLOCK_LEVEL_BATCH, the server MUST set RequestedOplockLevel to SMB2_OPLOCK_LEVEL_II.

Errata Published*	Description
	If CreateOptions includes FILE_NO_INTERMEDIATE_BUFFERING and DesiredAccess includes FILE_APPEND_DATA, the server MUST set FILE_APPEND_DATA to zero in the DesiredAccess field in the request. Changed to: If Connection.Dialect belongs to the SMB 3.x dialect family, TreeConnect.Share.Type is STYPE_CLUSTER_SOFS as specified in [MS-SRVS] section 2.2.2.4, and the RequestedOplockLevel is SMB2_OPLOCK_LEVEL_BATCH, the server MUST set RequestedOplockLevel to SMB2_OPLOCK_LEVEL_II. If CreateOptions includes FILE_NO_INTERMEDIATE_BUFFERING the server MUST set FILE_APPEND_DATA to zero in the DesiredAccess field in the request. In Section 3.3.5.9.8, Handling the SMB2_CREATE_REQUEST_LEASE Create Context the following was changed from: If Connection.Dialect belongs to the SMB 3.x dialect family, TreeConnect.Share.Type includes STYPE_CLUSTER_SOFS and if LeaseState includes SMB2_LEASE_READ_CACHING, the server MUST set LeaseState to SMB2_LEASE_READ_CACHING, otherwise set LeaseState to SMB2_LEASE_NONE. Changed to: If Connection.Dialect belongs to the SMB 3.x dialect family, TreeConnect.Share.Type is STYPE_CLUSTER_SOFS as specified in [MS-SRVS] section 2.2.2.4, and if LeaseState includes SMB2_LEASE_READ_CACHING, the server MUST set LeaseState to SMB2_LEASE_READ_CACHING, otherwise set LeaseState to SMB2_LEASE_NONE. In Section 3.3.5.9.11 Handling the SMB2_CREATE_REQUEST_LEASE_V2 Create Context the following was changed from: If TreeConnect.Share.Type includes STYPE_CLUSTER_SOFS and if LeaseState includes SMB2_LEASE_READ_CACHING, the server MUST set LeaseState to SMB2_LEASE_READ_CACHING, otherwise set LeaseState to SMB2_LEASE_NONE. Changed to: If TreeConnect.Share.Type is STYPE_CLUSTER_SOFS as specified in [MS-SRVS] section 2.2.2.4, and if LeaseState includes SMB2_LEASE_READ_CACHING, the server MUST set LeaseState to SMB2_LEASE_READ_CACHING, otherwise set LeaseState to SMB2_LEASE_NONE.
2020/08/03	In Section 3.3.5.9, Receiving an SMB2 CREATE Request, the following was changed from: ... For open requests on a share of type STYPE_DISKTREE (as indicated by TreeConnect.Share.Type), the server MUST do the following: • If TreeConnect.Share.RestrictExclusiveOpens is TRUE and the ShareAccess field does not include FILE_SHARE_READ, and the DesiredAccess field does not include GENERIC_ALL, GENERIC_WRITE, FILE_WRITE_DATA, FILE_WRITE_ATTRIBUTES, FILE_WRITE_EA, or

Errata Published*	Description
	FILE_APPEND_DATA, the server SHOULD<261> set FILE_SHARE_READ in the ShareAccess field. ● If TreeConnect.Share.ForceSharedDelete is TRUE, the server MUST set FILE_SHARE_DELETE in the ShareAccess field. ● f TreeConnect.Share.ForceLevel2Oplock is TRUE, and RequestedOplockLevel is SMB2_OPLOCK_LEVEL_BATCH or SMB2_OPLOCK_LEVEL_EXCLUSIVE, the server SHOULD<263> set RequestedOplockLevel to SMB2_OPLOCK_LEVEL_II. ● If Connection.Dialect belongs to the SMB 3.x dialect family TreeConnect.Share.Type includes STYPE_CLUSTER_SOFS and the RequestedOplockLevel is SMB2_OPLOCK_LEVEL_BATCH, the server MUST set RequestedOplockLevel to SMB2_OPLOCK_LEVEL_II. ● If CreateOptions includes FILE_NO_INTERMEDIATE_BUFFERING and DesiredAccess includes FILE_APPEND_DATA, the server MUST set FILE_APPEND_DATA to zero in the DesiredAccess field in the request. ... ● Open.LocalOpen is set to the open of the object in the local resource received as part of the local create operation. ● Open.GrantedAccess is the access granted to the caller for the open by the underlying object store. It MUST be equal to the DesiredAccess specified in the request, except in the case where MAXIMUM_ALLOWED is included in the DesiredAccess. ● If Open.GrantedAccess includes FILE_EXECUTE, the server MUST set FILE_READ_DATA in Open.GrantedAccess. ● Open.OplockLevel is set to SMB2_OPLOCK_LEVEL_NONE. ... Changed to: ... For open requests on a share of type STYPE_DISKTREE (as indicated by TreeConnect.Share.Type), the server MUST do the following: ● If DesiredAccess is zero, the server SHOULD<260> fail the request with STATUS_ACCESS_DENIED. ● If TreeConnect.Share.RestrictExclusiveOpens is TRUE and the ShareAccess field does not include FILE_SHARE_READ, and the DesiredAccess field does not include GENERIC_ALL, GENERIC_WRITE, FILE_WRITE_DATA, FILE_WRITE_ATTRIBUTES, FILE_WRITE_EA, or FILE_APPEND_DATA, the server SHOULD<261> set FILE_SHARE_READ in the ShareAccess field. ● If TreeConnect.Share.ForceSharedDelete is TRUE, the server MUST set FILE_SHARE_DELETE in the ShareAccess field.

Errata Published*	Description
	• If DesiredAccess is not equal to TreeConnect.MaximalAccess and TreeConnect.Share.ConnectSecurity is not empty, the server MUST perform as below: • Clear ACCESS_SYSTEM_SECURITY in DesiredAccess. If DesiredAccess is zero, the server MUST fail the request with STATUS_ACCESS_DENIED. • If Session.SecurityContext is empty, the server MUST fail the request with STATUS_ACCESS_DENIED. • The server MUST perform access check for the share in the underlying object store using the parameters Session.SecurityContext, TreeConnect.Share.ConnectSecurity and DesiredAccess.<262> If the underlying object store returns a failure, the server MUST fail the request with STATUS_ACCESS_DENIED. • If TreeConnect.Share.ForceLevel2Oplock is TRUE, and RequestedOplockLevel is SMB2_OPLOCK_LEVEL_BATCH or SMB2_OPLOCK_LEVEL_EXCLUSIVE, the server SHOULD<263> set RequestedOplockLevel to SMB2_OPLOCK_LEVEL_II. • If Connection.Dialect belongs to the SMB 3.x dialect family TreeConnect.Share.Type includes STYPE_CLUSTER_SOFS and the RequestedOplockLevel is SMB2_OPLOCK_LEVEL_BATCH, the server MUST set RequestedOplockLevel to SMB2_OPLOCK_LEVEL_II. • If CreateOptions includes FILE_NO_INTERMEDIATE_BUFFERING and DesiredAccess includes FILE_APPEND_DATA, the server MUST set FILE_APPEND_DATA to zero in the DesiredAccess field in the request. <260> Section 3.3.5.9: Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2 operating systems do not perform this verification and continue to process the request. <262> Section 3.3.5.9: Windows performs the access check by mapping SMB2 parameters to the object store parameters as described in [MS-FSA] section 2.1.4.14 AccessCheck -- Algorithm to Perform a General Access Check. Object Store parameter SMB2 parameter SecurityContext Session.SecurityContext SecurityDescriptor TreeConnect.Share.ConnectSecurity DesiredAccess DesiredAccess ... • Open.LocalOpen is set to the open of the object in the local resource received as part of the local create operation. • If DesiredAccess is equal to TreeConnect.MaximalAccess, the server MUST set Open.GrantedAccess to TreeConnect.MaximalAccess. • If DesiredAccess is not equal to TreeConnect.MaximalAccess and TreeConnect.Share.ConnectSecurity is empty, the server MUST set Open.GrantedAccess to

Errata Published*	Description
	FILE_ALL_ACCESS. Otherwise, If MAXIMUM_ALLOWED is not included in the DesiredAccess, the server MUST set Open.GrantedAccess to the DesiredAccess specified in the request. Otherwise, the server MUST set Open.GrantedAccess to DesiredAccess with GENERIC_ALL set. ● If DesiredAccess received in the request includes ACCESS_SYSTEM_SECURITY, the server MUST set ACCESS_SYSTEM_SECURITY in Open.GrantedAccess. ● If Open.GrantedAccess includes FILE_EXECUTE, the server MUST set FILE_READ_DATA in Open.GrantedAccess. ● Open.OplockLevel is set to SMB2_OPLOCK_LEVEL_NONE. ...
2020/07/06	In Section 2.2.42. SMB2 Compression_Transform_Header, updated the description for when the SMB2_COMPRESSION_FLAG_CHAINED is set. Changed from: The SMB2 COMPRESSION_TRANSFORM_HEADER is used by the client or server when sending compressed messages. This optional header is only valid for the SMB 3.1.1 dialect<70>. Changed to: The SMB2 COMPRESSION_TRANSFORM_HEADER is used by the client or server when sending compressed messages. When SMB2_COMPRESSION_FLAG_CHAINED is set in Flags field, the 8 bytes comprising CompressionAlgorithm, Flags, and Length fields of this structure are interpreted as the first SMB2_COMPRESSION_PAYLOAD_HEADER, specified in section 2.2.42.1. This optional header is only valid for the SMB 3.1.1 dialect<70>. In Section 2.2.42.1 SMB2_COMPRESSION_PAYLOAD_HEADER, renamed the AlgorithmId field to CompressionAlgorithm and the Reserved field to Flags. Changed from: The SMB2_COMPRESSION_PAYLOAD_HEADER is used by the client or server when sending chained compressed payloads. This optional structure is only valid for the SMB 3.1.1 dialect<71>.  Reserved (2 bytes): This field MUST NOT be used and MUST be reserved. The sender MUST set this to 0, and the receiver MUST ignore it. Length (4 bytes): The length, in bytes, of the compressed payload. OriginalPayloadSize (4 bytes): This optional field is present only when AlgorithmId is LZNT1, LZ77,

Errata Published*	Description
	or LZ77+Huffman. The size, in bytes, of the uncompressed payload. Changed to: The SMB2_COMPRESSION_PAYLOAD_HEADER is used by the client or server when sending chained compressed payloads. This optional structure is only valid for the SMB 3.1.1 dialect<71>.  Flags (2 bytes): This field MUST be set to one of the following values: Value / Meaning SMB2_COMPRESSION_FLAG_NONE 0x0000 — Indicates that this is not the first payload header in a chain of compressed payloads. SMB2_COMPRESSION_FLAG_CHAINED 0x0001 — When set, indicates that this is the first payload header in a chain of compressed payloads. Length (4 bytes): The length, in bytes, of the compressed payload including the size of OriginalPayloadSize field, if present. OriginalPayloadSize (4 bytes): This optional field is present only when CompressionAlgorithm is LZNT1, LZ77, or LZ77+Huffman. The size, in bytes, of the uncompressed payload. In Section 3.1.4.4 Compressing the Message, updated the processing rules for when the SMB2_COMPRESSION_PAYLOAD_HEADER is set in the Flags field. Changed from: • If RemainingUncompressedDataSize is greater than 1024, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to LZNT1, LZ77, or LZ77+Huffman specified in Connection.CompressionIds. The uncompressed data MUST be compressed using the algorithm specified in AlgorithmId as specified in [MS-XCA] section 2. Length MUST be set to the size of the compressed data. OriginalPayloadSize MUST be set to the size of the uncompressed data. CompressedMessage MUST be appended with the compressed data. RemainingUncompressedDataSize MUST be decremented by the size of data before compression. TotalCompressedDataSize MUST be incremented by the size of compressed data. • Otherwise if RemainingUncompressedDataSize is greater than zero and (size of the uncompressed SMB2 message / RemainingUncompressedDataSize) is greater than 2, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to NONE. Length MUST be set to RemainingUncompressedDataSize. CompressedMessage MUST be appended with the uncompressed data. RemainingUncompressedDataSize MUST be decremented by the size of data before compression. TotalCompressedDataSize MUST be incremented by the size of compressed data.

Errata Published*	Description
	• If BackwardDataPattern is not NULL and BackwardDataPattern.Repetitions is greater than zero, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to Pattern_V1. Length MUST be set to the size of SMB2_COMPRESSION_PATTERN_PAYLOAD_V1. CompressedMessage MUST be appended with BackwardDataPattern. TotalCompressedDataSize MUST be incremented by BackwardDataPattern.Repetitions. • If RemainingUncompressedDataSize is greater than zero, the sender MUST repeat step 2. If TotalCompressedDataSize+8 is less than the size of uncompressed SMB2 message, the sender MUST prepend CompressedMessage with first 8 bytes of SMB2 COMPRESSION_TRANSFORM_HEADER. OriginalCompressedSegmentSize MUST be set to the size of uncompressed SMB2 message. The compressed SMB2 message is sent. Otherwise, the original, uncompressed SMB2 message is sent. Changed to: • If RemainingUncompressedDataSize is greater than 1024, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. CompressionAlgorithm MUST be set to LZNT1, LZ77, or LZ77+Huffman specified in Connection.CompressionIds. The uncompressed data MUST be compressed using the algorithm specified in CompressionAlgorithm as specified in [MS-XCA] section 2. Length MUST be set to sum of the size of the compressed data and the size of OriginalPayloadSize field. OriginalPayloadSize MUST be set to the size of the uncompressed data. CompressedMessage MUST be appended with the compressed data. RemainingUncompressedDataSize MUST be decremented by the size of data before compression. TotalCompressedDataSize MUST be incremented by the size of compressed data. • Otherwise if RemainingUncompressedDataSize is greater than zero and (size of the uncompressed SMB2 message / RemainingUncompressedDataSize) is greater than 2, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. CompressionAlgorithm MUST be set to NONE. Length MUST be set to RemainingUncompressedDataSize. CompressedMessage MUST be appended with the uncompressed data. RemainingUncompressedDataSize MUST be decremented by the size of data before compression. TotalCompressedDataSize MUST be incremented by the size of compressed data. • If BackwardDataPattern is not NULL and BackwardDataPattern.Repetitions is greater than zero, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. CompressionAlgorithm MUST be set to Pattern_V1. Length MUST be set to the size of SMB2_COMPRESSION_PATTERN_PAYLOAD_V1. CompressedMessage MUST be appended with BackwardDataPattern. TotalCompressedDataSize MUST be incremented by BackwardDataPattern.Repetitions. • If RemainingUncompressedDataSize is greater than zero, the sender MUST repeat step 2. If TotalCompressedDataSize+8 is less than the size of uncompressed SMB2 message, the sender MUST prepend CompressedMessage with first 8 bytes of SMB2 COMPRESSION_TRANSFORM_HEADER. OriginalCompressedSegmentSize MUST be set to the size of uncompressed SMB2 message. The Flags field in the first SMB2_COMPRESSION_PAYLOAD_HEADER MUST be set to SMB2_COMPRESSION_FLAG_CHAINED. The compressed SMB2 message is sent. Otherwise, the original, uncompressed SMB2 message is sent. In Section 3.1.5.3 Decompressing the Chained Message, updated the processing rules for compression. Changed from:

Errata Published*	Description
	The compressed data MUST be decompressed as follows: • The first 8 bytes of the data MUST be interpreted as SMB2_COMPRESSION_PAYLOAD_HEADER, specified in section 2.2.42.1. • If AlgorithmId in SMB2_COMPRESSION_PAYLOAD_HEADER is not one of the values specified in section 2.2.3.1.3, the connection MUST be disconnected as specified in section 3.2.7.1 or 3.3.7.1. • If AlgorithmId in SMB2_COMPRESSION_PAYLOAD_HEADER is NONE: If Length is greater than (the size of the received compressed message – 8) or OriginalCompressedSegmentSize inSMB2 COMPRESSION_TRANSFORM_HEADER, the connection MUST be disconnected as specified in section 3.2.7.1 or 3.3.7.1. Length number of bytes following SMB2_COMPRESSION_PAYLOAD_HEADER MUST be interpreted as uncompressed data and MUST be appended to DecompressedMessage. Otherwise, the data MUST be decompressed as follows: If AlgorithmId is Pattern_V1, the next 8 bytes MUST be interpreted as SMB2_COMPRESSION_PATTERN_PAYLOAD_V1, specified in section 2.2.42.2. If Repetitions in SMB2_COMPRESSION_PATTERN_PAYLOAD_V1 is greater than OriginalCompressedSegmentSize inSMB2 COMPRESSION_TRANSFORM_HEADER, the connection MUST be disconnected as specified in section 3.2.7.1 or 3.3.7.1. Otherwise, DecompressedMessage MUST be appended with Repetitions number of bytes initialized with the character specified in Pattern field. Otherwise, the data of size specified in Length field MUST be decompressed using the algorithm specified in AlgorithmId field as specified in [MS-XCA] section 2. If the size of the decompressed data is not equal to OriginalPayloadSize, the connection MUST be disconnected as specified in section 3.2.7.1 or section 3.3.7.1. DecompressedMessage MUST be appended with the decompressed data. RemainingCompressedDataSize MUST be decremented by the size in Length field. If the size of RemainingCompressedDataSize is greater than the size of SMB2_COMPRESSION_PAYLOAD_HEADER, the receiver MUST repeat step 2. DecompressedMessage MUST be returned. Changed to: The compressed data MUST be decompressed as follows: • The first 8 bytes of the data MUST be interpreted as SMB2_COMPRESSION_PAYLOAD_HEADER, specified in section 2.2.42.1. • If CompressionAlgorithm in SMB2_COMPRESSION_PAYLOAD_HEADER is not one of the values specified in section 2.2.3.1.3, the connection MUST be disconnected as specified in section 3.2.7.1 or 3.3.7.1. • If CompressionAlgorithm in SMB2_COMPRESSION_PAYLOAD_HEADER is NONE: If Length is greater than (the size of the received compressed message – 8) or

Errata Published*	Description
	OriginalCompressedSegmentSize inSMB2 COMPRESSION_TRANSFORM_HEADER, the connection MUST be disconnected as specified in section 3.2.7.1 or 3.3.7.1. Length number of bytes following SMB2_COMPRESSION_PAYLOAD_HEADER MUST be interpreted as uncompressed data and MUST be appended to DecompressedMessage. Otherwise, the data MUST be decompressed as follows: If CompressionAlgorithm is Pattern_V1, the next 8 bytes MUST be interpreted as SMB2_COMPRESSION_PATTERN_PAYLOAD_V1, specified in section 2.2.42.2. If Repetitions in SMB2_COMPRESSION_PATTERN_PAYLOAD_V1 is greater than OriginalCompressedSegmentSize inSMB2 COMPRESSION_TRANSFORM_HEADER, the connection MUST be disconnected as specified in section 3.2.7.1 or 3.3.7.1. Otherwise, DecompressedMessage MUST be appended with Repetitions number of bytes initialized with the character specified in Pattern field. Otherwise, the data of size specified in Length field minus size of OriginalPayloadSize field MUST be decompressed using the algorithm specified in CompressionAlgorithm field as specified in [MS-XCA] section 2. If the size of the decompressed data is not equal to OriginalPayloadSize, the connection MUST be disconnected as specified in section 3.2.7.1 or section 3.3.7.1. DecompressedMessage MUST be appended with the decompressed data. RemainingCompressedDataSize MUST be decremented by the size in Length field. If the size of RemainingCompressedDataSize is greater than the size of SMB2_COMPRESSION_PAYLOAD_HEADER, the receiver MUST repeat step 2. DecompressedMessage MUST be returned. In Section 3.2.5.1.1.2 Decompressing the Message, updated the processing rules for compression. Changed from: • The client MUST disconnect the connection as specified in section 3.2.7.1 if any of the following conditions are satisfied: • If the size of the message received from the server is less than the size of SMB2 COMPRESSION_TRANSFORM_HEADER, specified in section 2.2.42. • If Connection.CompressionIds does not contain CompressionAlgorithm in SMB2 COMPRESSION_TRANSFORM_HEADER. • If OriginalCompressedSegmentSize in the SMB2 COMPRESSION_TRANSFORM_HEADER is greater than the sum of (256, the size of SMB2 COMPRESSION_TRANSFORM_HEADER, largest of (Connection.MaxReadSize, Connection.MaxWriteSize,and Connection.MaxTransactSize)). Changed to: • The client MUST disconnect the connection as specified in section 3.2.7.1 if any of the following conditions are satisfied: • If the size of the message received from the server is less than the size of SMB2 COMPRESSION_TRANSFORM_HEADER, specified in section 2.2.42. • If Flags field in SMB2 COMPRESSION_TRANSFORM_HEADER is equal to SMB2_COMPRESSION_FLAG_NONE and Connection.CompressionIds does not contain

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>CompressionAlgorithm in SMB2 COMPRESSION_TRANSFORM_HEADER.
• If OriginalCompressedSegmentSize in the SMB2 COMPRESSION_TRANSFORM_HEADER is greater than the sum of (256, the size of SMB2 COMPRESSION_TRANSFORM_HEADER, largest of (Connection.MaxReadSize, Connection.MaxWriteSize,and Connection.MaxTransactSize)).

In Section 3.3.5.2.1.2 Decompressing the Message, updated the processing rules for compression.

Changed from:
• The server MUST disconnect the connection as specified in section 3.3.7.1 if any of the following conditions are satisfied:
• If the size of the message received from the client is less than the size of SMB2 COMPRESSION_TRANSFORM_HEADER, specified in section 2.2.42.
• If Connection.CompressionIds does not contain the CompressionAlgorithm field in the SMB2 COMPRESSION_TRANSFORM_HEADER.
• If OriginalCompressedSegmentSize in the SMB2 COMPRESSION_TRANSFORM_HEADER is greater than the sum of (256, the size of SMB2 COMPRESSION_TRANSFORM_HEADER, largest of (Connection.MaxReadSize, Connection.MaxWriteSize,and Connection.MaxTransactSize)).
Changed to:
•The server MUST disconnect the connection as specified in section 3.3.7.1 if any of the following conditions are satisfied:
• If the size of the message received from the client is less than the size of SMB2 COMPRESSION_TRANSFORM_HEADER, specified in section 2.2.42.
• If Flags field in SMB2 COMPRESSION_TRANSFORM_HEADER is equal to SMB2_COMPRESSION_FLAG_NONE and Connection.CompressionIds does not contain the CompressionAlgorithm field in the SMB2 COMPRESSION_TRANSFORM_HEADER.
• If OriginalCompressedSegmentSize in the SMB2 COMPRESSION_TRANSFORM_HEADER is greater than the sum of (256, the size of SMB2 COMPRESSION_TRANSFORM_HEADER, largest of (Connection.MaxReadSize, Connection.MaxWriteSize,and Connection.MaxTransactSize)).</td></tr>
<tr><td>2020/06/22</td><td>In Section 2.2.19, SMB2 READ Request, updated the description of the RemainingBytes field, adding the field to the value descriptions in the table for Channel.

Changed from:
Channel (4 bytes): For SMB 2.0.2 and 2.1 dialects, this field MUST NOT be used and MUST be reserved. The client MUST set this field to 0, and the server MUST ignore it on receipt. For the SMB 3.x dialect family, this field MUST contain exactly one of the following values:
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>SMB2_CHANNEL_NONE
0x00000000</td><td>No channel information is present in the request. The ReadChannelInfoOffset and ReadChannelInfoLength fields MUST be set to 0 by the client and MUST be ignored by the server.</td></tr>
<tr><td>SMB2_CHANNEL_RDMA_V1
0x00000001</td><td>One or more SMB_DIRECT_BUFFER_DESCRIPTOR_V1 structures as specified in [MS-SMBD] section 2.2.3.1 are present in the channel information specified by ReadChannelInfoOffset and ReadChannelInfoLength fields.</td></tr>
<tr><td>SMB2_CHANNEL_RDMA_V1_INVALIDATE
0x00000002</td><td>This flag is not valid for the SMB 3.0 dialect. One or more SMB_DIRECT_BUFFER_DESCRIPTOR_V1 structures, as specified in [MS-SMBD] section</td></tr>
</table></td></tr>
</table>

Errata Published*	Description

	2.2.3.1, are present in the channel information specified by the ReadChannelInfoOffset and ReadChannelInfoLength fields. The server is requested to perform remote invalidation when responding to the request as specified in [MS-SMBD] section 3.1.4.2. RemainingBytes (4 bytes): The number of subsequent bytes that the client intends to read from the file after this operation completes. This value is provided to facilitate read-ahead caching, and is not binding on the server.

Changed to:

Channel (4 bytes): For SMB 2.0.2 and 2.1 dialects, this field MUST NOT be used and MUST be reserved. The client MUST set this field to 0, and the server MUST ignore it on receipt. For the SMB 3.x dialect family, this field MUST contain exactly one of the following values:

Value	Meaning
SMB2_CHANNEL_NONE 0x00000000	No channel information is present in the request. The RemainingBytes, ReadChannelInfoOffset, and ReadChannelInfoLength fields MUST be set to 0 by the client and MUST be ignored by the server.
SMB2_CHANNEL_RDMA_V1 0x00000001	One or more SMB_DIRECT_BUFFER_DESCRIPTOR_V1 structures as specified in [MS-SMBD] section 2.2.3.1 are present in the channel information specified by RemainingBytes,ReadChannelInfoOffset, and ReadChannelInfoLength fields.
SMB2_CHANNEL_RDMA_V1_INVALIDATE 0x00000002	This flag is not valid for the SMB 3.0 dialect. One or more SMB_DIRECT_BUFFER_DESCRIPTOR_V1 structures, as specified in [MS-SMBD] section 2.2.3.1, are present in the channel information specified by the RemainingBytes, ReadChannelInfoOffset, and ReadChannelInfoLength fields. The server is requested to perform remote invalidation when responding to the request as specified in [MS-SMBD] section 3.1.4.2. RemainingBytes (4 bytes): For the SMB 3.x dialect family, if the Channel field of the request contains SMB2_CHANNEL_RDMA_V1 or SMB2_CHANNEL_RDMA_V1_INVALIDATE, this field contains the length, in bytes, of the data to be read.

In Section 2.2.21, SMB2 WRITE Request, updated the description of the RemainingBytes, WriteChannelInfoOffset, and WriteChannelInfoLength fields.

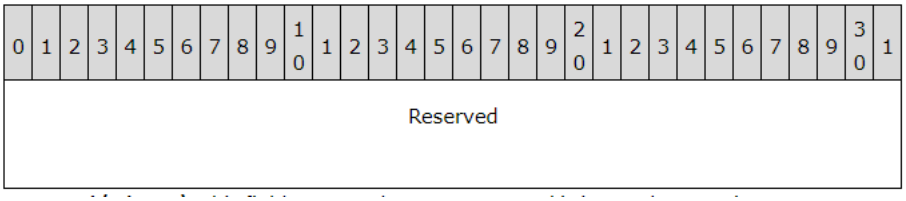
Changed from:

RemainingBytes (4 bytes): For the SMB 3.x dialect family and the Channel field of the request contains SMB2_CHANNEL_RDMA_V1 or SMB2_CHANNEL_RDMA_V1_INVALIDATE, this field contains the length, in bytes, of the data being written.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>WriteChannelInfoOffset (2 bytes): For the SMB 3.x dialect family and the Channel field of the request contains SMB2_CHANNEL_RDMA_V1 or SMB2_CHANNEL_RDMA_V1_INVALIDATE, it contains the offset, in bytes, from the beginning of the SMB2 header to the channel data as specified by the Channel field of the request.
WriteChannelInfoLength (2 bytes): For the SMB 3.x dialect family and the Channel field of the request contains SMB2_CHANNEL_RDMA_V1 or SMB2_CHANNEL_RDMA_V1_INVALIDATE, it contains the length, in bytes, of the channel data as specified by the Channel field of the request.

Changed to:
RemainingBytes (4 bytes): For the SMB 3.x dialect family, if the Channel field of the request contains SMB2_CHANNEL_RDMA_V1 or SMB2_CHANNEL_RDMA_V1_INVALIDATE, this field contains the length, in bytes, of the data being written.
WriteChannelInfoOffset (2 bytes): For the SMB 3.x dialect family, if the Channel field of the request contains SMB2_CHANNEL_RDMA_V1 or SMB2_CHANNEL_RDMA_V1_INVALIDATE, it contains the offset, in bytes, from the beginning of the SMB2 header to the channel data as specified by the Channel field of the request.
WriteChannelInfoLength (2 bytes): For the SMB 3.x dialect family, if the Channel field of the request contains SMB2_CHANNEL_RDMA_V1 or SMB2_CHANNEL_RDMA_V1_INVALIDATE, it contains the length, in bytes, of the channel data as specified by the Channel field of the request.</td></tr>
<tr><td>2020/06/22</td><td>In Section 2.2.3.1, SMB2 NEGOTIATE_CONTEXT Request Values, added SMB2_TRANSPORT_CAPABILITIES value to ContextType.

Changed from:
ContextType (2 bytes): Specifies the type of context in the Data field. This field MUST be one of the following values:
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>SMB2_PREAUTH_INTEGRITY_CAPABILITIES
0x0001</td><td>The Data field contains a list of preauthentication integrity hash functions as well as an optional salt value, as specified in section 2.2.3.1.1.</td></tr>
<tr><td>SMB2_ENCRYPTION_CAPABILITIES
0x0002</td><td>The Data field contains a list of encryption algorithms, as specified in section 2.2.3.1.2.</td></tr>
<tr><td>SMB2_COMPRESSION_CAPABILITIES
0x0003</td><td>The Data field contains a list of compression algorithms, as specified in section 2.2.3.1.3<13>.</td></tr>
<tr><td>SMB2_NETNAME_NEGOTIATE_CONTEXT_ID
0x0005</td><td>The Data field contains the server name to which the client connects.<14></td></tr>
</table>
Changed to:
ContextType (2 bytes): Specifies the type of context in the Data field. This field MUST be one of the following values:
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>SMB2_PREAUTH_INTEGRITY_CAPABILITIES
0x0001</td><td>The Data field contains a list of preauthentication integrity hash functions as well as an optional salt value, as specified in section 2.2.3.1.1.</td></tr>
<tr><td>SMB2_ENCRYPTION_CAPABILITIES</td><td>The Data field contains a list of encryption</td></tr>
</table></td></tr>
</table>

Errata Published*	Description
	0x0002 — algorithms, as specified in section 2.2.3.1.2. SMB2_COMPRESSION_CAPABILITIES 0x0003 — The Data field contains a list of compression algorithms, as specified in section 2.2.3.1.3<13>. SMB2_NETNAME_NEGOTIATE_CONTEXT_ID 0x0005 — The Data field contains the server name to which the client connects<14>. SMB2_TRANSPORT_CAPABILITIES 0x0006 — The Data field contains transport capabilities, as specified in section 2.2.3.1.5<15>. Added a new Section 2.2.3.1.5, SMB2_TRANSPORT_CAPABILITIES. 2.2.3.1.5 SMB2_TRANSPORT_CAPABILITIES The SMB2_TRANSPORT_CAPABILITIES context is specified in an SMB2 NEGOTIATE request to indicate transport capabilities over which the connection is made. The server MUST ignore the context on receipt. The format of the data in the Data field of this SMB2_NEGOTIATE_CONTEXT is as follows.  Reserved (4 bytes): This field SHOULD be set to zero and is ignored on receipt. In Section 6, Appendix : Product Behavior, added a new product behavior note to support a new value. <15> Section 2.2.3.1: Windows 10 v1909 operating system and prior and Windows Server v1909 operating system and prior do not send or process SMB2_TRANSPORT_CAPABILITIES.
2020/06/22	In Section 3.2.1.9, Per Server, added abstract data model element CipherId. Changed from : ServerName: A Unicode UTF-16 fully qualified domain name, a NetBIOS name, or an IP address of the server machine. Changed to: ServerName: A Unicode UTF-16 fully qualified domain name, a NetBIOS name, or an IP address of the server machine. CipherId: The encryption algorithm that was negotiated between client and server. In Section 3.2.4.2.2.2, SMB2 Only Negotiate, updated processing rules for CipherId. Changed from: • If IsEncryptionSupported is TRUE, it MUST do the following: • Increment NegotiateContextCount by 1.

Errata Published*	Description
	• Add an SMB2_NEGOTIATE_CONTEXT with ContextType as SMB2_ENCRYPTION_CAPABILITIES to the negotiate request as specified in section 2.2.3.1 and initialize the Ciphers field with the ciphers supported by the client in the order of preference.<107> Changed to: • If IsEncryptionSupported is TRUE, it MUST do the following: • Increment NegotiateContextCount by 1. • Add an SMB2_NEGOTIATE_CONTEXT with ContextType as SMB2_ENCRYPTION_CAPABILITIES to the negotiate request as specified in section 2.2.3.1. • If an alternate connection is being established to an already connected Server, set Ciphers to Server.CipherId and CipherCount to 1. Otherwise, set Ciphers with the ciphers supported by the client, if any, in the order of preference and CipherCount to number of ciphers in Ciphers field.<107> In Section 3.2.5.2, Receiving an SMB NEGOTIATE Response, updated processing rules for CipherId. Changed from: • Processing the SMB2_ENCRYPTION_CAPABILITIES negotiate context • The client MUST return an error to the calling application in the following cases: • The DataLength of the negotiate context is less than the size of SMB2_ENCRYPTION_CAPABILITIES structure. • CipherCount is not 1. • Ciphers[0] is not 0 and not one of the ciphers that the client specified in its negotiate request. • The client MUST set Connection.CipherId to Ciphers[0]. • If Connection.CipherId is nonzero, the client MUST set Connection.SupportsEncryption to TRUE. Otherwise, it MUST be set to FALSE. Changed to: • Processing the SMB2_ENCRYPTION_CAPABILITIES negotiate context • The client MUST return an error to the calling application in the following cases: • The DataLength of the negotiate context is less than the size of SMB2_ENCRYPTION_CAPABILITIES structure. • CipherCount is not 1. • Ciphers[0] is not 0 and not one of the ciphers that the client specified in its negotiate request. • The client MUST set Connection.CipherId to Ciphers[0] and if Server.CipherId is empty, set Server.CipherId to Ciphers[0]. • If Connection.CipherId is nonzero, the client MUST set Connection.SupportsEncryption to TRUE. Otherwise, it MUST be set to FALSE.
2020/06/22	Added a new Section 3.3.5.15.19, Handling a Set Read CopyNumber Request. 3.3.5.15.19 Handling a Set Read CopyNumber Request This section applies only to servers that implement the SMB 3.1.1 dialect. When the server receives a request that contains an SMB2 header with a Command value equal to SMB2 IOCTL and a CtlCode of FSCTL_MARK_HANDLE, message handling proceeds as follows: If the InputCount in SMB2 IOCTL request is less than the size of FSCTL_MARK_HANDLE request,

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
as specified in [MS-FSCC] section 2.3.31, the server MUST fail the request with STATUS_INVALID_PARAMETER.
The server MUST fail the request with STATUS_NOT_SUPPORTED in the following cases:
● If Connection.Dialect is "2.0.2", "2.1", "3.0" or "3.0.2".
● If HandleInfo received in the request is not one of the values defined in [MS-FSCC] section 2.3.31.
The server MUST process this request as a pass-through operation as specified in section 3.3.5.15.8.

In Section 3.3.5.15, Receiving an SMB2 IOCTL Request: updated PBN#324, adding FSCTL_MARK_HANDLE.

Changed from:
Windows 10 and later and Windows Server 2016 and later allow the additional CtlCode value, as specified in [MS-FSCC].
<table>
<tr><th>FSCTL name</th><th>FSCTL function number</th></tr>
<tr><td>FSCTL_DUPLICATE_EXTENTS_TO_FILE</td><td>0x98344
Windows 10 v1803 operating system and later and Windows Server v1803 operating system and later allow the additional CtlCode value, as specified in [MS-FSCC].</td></tr>
</table>
Changed to:
Windows 10 and later and Windows Server 2016 and later allow the additional CtlCode value, as specified in [MS-FSCC].
<table>
<tr><th>FSCTL name</th><th>FSCTL function number</th></tr>
<tr><td>FSCTL_DUPLICATE_EXTENTS_TO_FILE</td><td>0x98344
Windows 10 v1607 operating system and later and Windows Server 2016 operating system and later allow the additional CtlCode value, as specified in [MS-FSCC].</td></tr>
</table>
<table>
<tr><th>FSCTL name</th><th>FSCTL function number</th></tr>
<tr><td>FSCTL_MARK_HANDLE</td><td>0x900FC
[A1] Windows 10 v1803 operating system and later and Windows Server v1803 operating system and later allow the additional CtlCode value, as specified in [MS-FSCC].</td></tr>
</table>
In Section 3.3.5.15.8, Handling a Pass Through Operation Request: updated PBN #341 updating the information for FSCTL_MARK_HANDLE (0x000900FC).

Changed from:
FSCTL_OPLOCK_BREAK_NOTIFY (0x00090014)
FSCTL_MOVE_FILE (0x00090074)
FSCTL_MARK_HANDLE (0x000900FC)
</td></tr>
</table>

Errata Published*	Description
	FSCTL_QUERY_RETRIEVAL_POINTERS (0x0009003B) FSCTL_PIPE_ASSIGN_EVENT (0x00110000) FSCTL_GET_VOLUME_BITMAP (0x0009006F) FSCTL_GET_NTFS_FILE_RECORD (0x00090068) FSCTL_INVALIDATE_VOLUMES (0x00090054) FSCTL_READ_USN_JOURNAL (0x000900BB) FSCTL_CREATE_USN_JOURNAL (0x000900E7) FSCTL_QUERY_USN_JOURNAL (0x000900F4) FSCTL_DELETE_USN_JOURNAL (0x000900F8) FSCTL_ENUM_USN_DATA (0x000900B3) FSCTL_QUERY_DEPENDENT_VOLUME (0x000901F0) FSCTL_SD_GLOBAL_CHANGE (0x000901F4) FSCTL_GET_BOOT_AREA_INFO (0x00090230) FSCTL_GET_RETRIEVAL_POINTER_BASE (0x00090234) FSCTL_SET_PERSISTENT_VOLUME_STATE (0x00090238) FSCTL_QUERY_PERSISTENT_VOLUME_STATE (0x0009023C) FSCTL_REQUEST_OPLOCK (0x00090240) FSCTL_TXFS_MODIFY_RM (0x00098144) FSCTL_TXFS_QUERY_RM_INFORMATION (0x00094148) FSCTL_TXFS_ROLLFORWARD_REDO (0x00098150) FSCTL_TXFS_ROLLFORWARD_UNDO (0x00098154) FSCTL_TXFS_START_RM (0x00098158) FSCTL_TXFS_SHUTDOWN_RM (0x0009815C) FSCTL_TXFS_READ_BACKUP_INFORMATION (0x00094160) FSCTL_TXFS_WRITE_BACKUP_INFORMATION (0x00098164) FSCTL_TXFS_CREATE_SECONDARY_RM (0x00098168) FSCTL_TXFS_GET_METADATA_INFO (0x0009416C) FSCTL_TXFS_GET_TRANSACTED_VERSION (0x00094170) FSCTL_TXFS_SAVEPOINT_INFORMATION (0x00098178) FSCTL_TXFS_CREATE_MINIVERSION (0x0009817C) FSCTL_TXFS_TRANSACTION_ACTIVE (0x0009418C) FSCTL_TXFS_LIST_TRANSACTIONS (0x000941E4) FSCTL_TXFS_READ_BACKUP_INFORMATION2 (0x000901F8) FSCTL_TXFS_WRITE_BACKUP_INFORMATION2 (0x00090200) FSCTL_QUERY_FILE_REGIONS (0x00090284) FSCTL_IS_CSV_FILE (0x00090248) FSCTL_IS_FILE_ON_CSV_VOLUME (0x0009025C) Windows Vista SP1, Windows 7, Windows Server 2008, and Windows Server 2008 R2 fail FSCTLs whose transfer type is METHOD_NEITHER with error STATUS_NOT_SUPPORTED except the following ones. For more information about FSCTL transfer type, see [MSDN-IoCtlCodes]. FSCTL_PIPE_TRANSCEIVE (0x0011C017) Changed to: FSCTL_OPLOCK_BREAK_NOTIFY (0x00090014) FSCTL_MOVE_FILE (0x00090074)

Errata Published*	Description
	FSCTL_QUERY_RETRIEVAL_POINTERS (0x0009003B) FSCTL_PIPE_ASSIGN_EVENT (0x00110000) FSCTL_GET_VOLUME_BITMAP (0x0009006F) FSCTL_GET_NTFS_FILE_RECORD (0x00090068) FSCTL_INVALIDATE_VOLUMES (0x00090054) FSCTL_READ_USN_JOURNAL (0x000900BB) FSCTL_CREATE_USN_JOURNAL (0x000900E7) FSCTL_QUERY_USN_JOURNAL (0x000900F4) FSCTL_DELETE_USN_JOURNAL (0x000900F8) FSCTL_ENUM_USN_DATA (0x000900B3) FSCTL_QUERY_DEPENDENT_VOLUME (0x000901F0) FSCTL_SD_GLOBAL_CHANGE (0x000901F4) FSCTL_GET_BOOT_AREA_INFO (0x00090230) FSCTL_GET_RETRIEVAL_POINTER_BASE (0x00090234) FSCTL_SET_PERSISTENT_VOLUME_STATE (0x00090238) FSCTL_QUERY_PERSISTENT_VOLUME_STATE (0x0009023C) FSCTL_REQUEST_OPLOCK (0x00090240) FSCTL_TXFS_MODIFY_RM (0x00098144) FSCTL_TXFS_QUERY_RM_INFORMATION (0x00094148) FSCTL_TXFS_ROLLFORWARD_REDO (0x00098150) FSCTL_TXFS_ROLLFORWARD_UNDO (0x00098154) FSCTL_TXFS_START_RM (0x00098158) FSCTL_TXFS_SHUTDOWN_RM (0x0009815C) FSCTL_TXFS_READ_BACKUP_INFORMATION (0x00094160) FSCTL_TXFS_WRITE_BACKUP_INFORMATION (0x00098164) FSCTL_TXFS_CREATE_SECONDARY_RM (0x00098168) FSCTL_TXFS_GET_METADATA_INFO (0x0009416C) FSCTL_TXFS_GET_TRANSACTED_VERSION (0x00094170) FSCTL_TXFS_SAVEPOINT_INFORMATION (0x00098178) FSCTL_TXFS_CREATE_MINIVERSION (0x0009817C) FSCTL_TXFS_TRANSACTION_ACTIVE (0x0009418C) FSCTL_TXFS_LIST_TRANSACTIONS (0x000941E4) FSCTL_TXFS_READ_BACKUP_INFORMATION2 (0x000901F8) FSCTL_TXFS_WRITE_BACKUP_INFORMATION2 (0x00090200) FSCTL_QUERY_FILE_REGIONS (0x00090284) FSCTL_IS_CSV_FILE (0x00090248) FSCTL_IS_FILE_ON_CSV_VOLUME (0x0009025C) Windows 10 v1511 operating system and prior and Windows Server 2012 R2 operating system and prior block FSCTL_MARK_HANDLE (0x000900FC) and do not pass it through to the object store. The request is failed with STATUS_NOT_SUPPORTED. Windows Vista SP1, Windows 7, Windows Server 2008, and Windows Server 2008 R2 fail FSCTLs whose transfer type is METHOD_NEITHER with error STATUS_NOT_SUPPORTED except the following ones. For more information about FSCTL transfer type, see [MSDN-IoCtlCodes]. FSCTL_PIPE_TRANSCEIVE (0x0011C017)

Errata Published*	Description
2020/06/08	In Section 3.2.1.7, Per Pending Request, added a definition forRequest.BufferDescriptorList. Changed from: Request.Timestamp: The time at which the request was sent to the server. Changed to: Request.Timestamp: The time at which the request was sent to the server. If the client implements the SMB 3.x dialect family, it also implements the following: Request.BufferDescriptorList: For a READ/WRITE request sent over RDMA, this is a list of SMB_DIRECT_BUFFER_DESCRIPTOR_V1 structures returned by [MS-SMBD] section 3.1.4.3.
2020/06/08	In Section 3.3.5.4, Receiving an SMB2 NEGOTIATE Request, the following has been changed from: • Building an SMB2_COMPRESSION_CAPABILITIES negotiate response context: • If the server processed the SMB2_COMPRESSION_CAPABILITIES negotiate request context, then the server MUST build an SMB2_COMPRESSION_CAPABILITIES negotiate response context by setting the following: • If Connection.CompressionIds is empty, • Set CompressionAlgorithmCount to 1.Set CompressionAlgorithms to "NONE". • Otherwise, • Set CompressionAlgorithmCount to the number of compression algorithms in Connection.CompressionIds. • Set CompressionAlgorithms to Connection.CompressionIds. • If IsChainedCompressionSupported is TRUE and SMB2_COMPRESSION_CAPABILITIES_FLAG_CHAINED bit is set in Flags field of negotiate request context, SMB2_COMPRESSION_CAPABILITIES_FLAG_CHAINED bit MUST be set in Flags field and Connection.SupportsChainedCompression MUST be set to TRUE. Changed to: • Building an SMB2_COMPRESSION_CAPABILITIES negotiate response context: • If the server processed the SMB2_COMPRESSION_CAPABILITIES negotiate request context, then the server MUST build an SMB2_COMPRESSION_CAPABILITIES negotiate response context by setting the following: • If IsChainedCompressionSupported is TRUE and SMB2_COMPRESSION_CAPABILITIES_FLAG_CHAINED bit is set in Flags field of negotiate request context, SMB2_COMPRESSION_CAPABILITIES_FLAG_CHAINED bit MUST be set in Flags field and Connection.SupportsChainedCompression MUST be set to TRUE. • If Connection.CompressionIds is empty, • Set CompressionAlgorithmCount to 1. • Set CompressionAlgorithms to "NONE". • Otherwise, • Set CompressionAlgorithmCount to the number of compression algorithms in Connection.CompressionIds. • Set CompressionAlgorithms to Connection.CompressionIds.
2020/05/25	**For a Diff of the below changes, see the PDF doc here.** In Section 6, Appendix A: Product Behavior, product behavior note <341> for Section 3.3.5.15.8

Errata Published*	Description
	was changed from: ... FSCTL_IS_FILE_ON_CSV_VOLUME (0x0009025C) Windows-based SMB2 servers fail FSCTLs whose transfer type is METHOD_NEITHER with error STATUS_NOT_SUPPORTED except the following ones. For more information about FSCTL transfer type, see [MSDN-IoCtlCodes]. FSCTL_PIPE_TRANSCEIVE (0x0011C017) ... Changed to: ... FSCTL_IS_FILE_ON_CSV_VOLUME (0x0009025C) Windows Vista SP1, Windows 7, Windows Server 2008, and Windows Server 2008 R2 fail FSCTLs whose transfer type is METHOD_NEITHER with error STATUS_NOT_SUPPORTED except the following ones. For more information about FSCTL transfer type, see [MSDN-IoCtlCodes]. FSCTL_PIPE_TRANSCEIVE (0x0011C017) ...
2020/05/25	In Section 3.1.5.3, Decompressing the Chained Message, the following was changed from: If IsCompressionSupported is FALSE, Connection.SupportsChainedCompression is FALSE, or Connection.CompressionIds is empty, the receiver MUST skip the processing in this section. 1. The sender MUST initialize RemainingCompressedDataSize with the size of the received compressed SMB2 message and DecompressedMessage with empty buffer. 2. The compressed message MUST be decompressed until the size of RemainingCompressedDataSize is greater than the size of SMB2COMPRESSION_PAYLOAD_HEADER: ● The first 8 bytes of the data MUST be interpreted as SMB2_COMPRESSION_PAYLOAD_HEADER, specified in section 2.2.42.1. ● If AlgorithmId in SMB2_COMPRESSION_PAYLOAD_HEADER is not one of the values specified in section 2.2.3.1.3, the connection MUST be disconnected as specified in section 3.2.7.1 or 3.3.7.1. ● If AlgorithmId in SMB2_COMPRESSION_PAYLOAD_HEADER is NONE: ● If Length is greater than (the size of the received compressed message – 8) or OriginalCompressedSegmentSize in SMB2 COMPRESSION_TRANSFORM_HEADER, the connection

Errata Published*	Description
	MUST be disconnected as specified in section 3.2.7.1 or 3.3.7.1. • Length number of bytes following SMB2_COMPRESSION_PAYLOAD_HEADER MUST be interpreted as uncompressed data and MUST be appended to DecompressedMessage. • Otherwise, the data MUST be decompressed as follows: • If AlgorithmId is Pattern_V1, the next 8 bytes MUST be interpreted as SMB2_COMPRESSION_PATTERN_PAYLOAD_V1, specified in section 2.2.42.2. • If Repetitions in SMB2_COMPRESSION_PATTERN_PAYLOAD_V1 is greater than OriginalCompressedSegmentSize in SMB2 COMPRESSION_TRANSFORM_HEADER, the connection MUST be disconnected as specified in section 3.2.7.1 or 3.3.7.1. • Otherwise, DecompressedMessage MUST be appended with Repetitions number of bytes initialized with the character specified in Pattern field. • Otherwise, the data of size specified in Length field MUST be decompressed using the algorithm specified in AlgorithmId field as specified in [MS-XCA] section 2. DecompressedMessage MUST be appended with the decompressed data. • RemainingCompressedDataSize MUST be decremented by the size in Length field. 3. DecompressedMessage MUST be returned. Changed to: If IsCompressionSupported is FALSE, Connection.SupportsChainedCompression is FALSE, or Connection.CompressionIds is empty, the receiver MUST skip the processing in this section. 1. The receiver MUST initialize RemainingCompressedDataSize with the size of the received compressed SMB2 message and DecompressedMessage with empty buffer. 2. If the size of RemainingCompressedDataSize is greater than the size of SMB2_COMPRESSION_PAYLOAD_HEADER, the compressed message MUST be decompressed as follows: • The first 8 bytes of the data MUST be interpreted as SMB2_COMPRESSION_PAYLOAD_HEADER, specified in section 2.2.42.1. • If AlgorithmId in SMB2_COMPRESSION_PAYLOAD_HEADER is not one of the values specified in section 2.2.3.1.3, the connection MUST be disconnected as specified in section 3.2.7.1 or 3.3.7.1. • If AlgorithmId in SMB2_COMPRESSION_PAYLOAD_HEADER is NONE: • If Length is greater than (the size of the received compressed message – 8) or OriginalCompressedSegmentSize in SMB2 COMPRESSION_TRANSFORM_HEADER, the connection MUST be disconnected as specified in section 3.2.7.1 or 3.3.7.1. • Length number of bytes following SMB2_COMPRESSION_PAYLOAD_HEADER MUST be interpreted as uncompressed data and MUST be appended to DecompressedMessage.

Errata Published*	Description
	• Otherwise, the data MUST be decompressed as follows: • If AlgorithmId is Pattern_V1, the next 8 bytes MUST be interpreted as SMB2_COMPRESSION_PATTERN_PAYLOAD_V1, specified in section 2.2.42.2. • If Repetitions in SMB2_COMPRESSION_PATTERN_PAYLOAD_V1 is greater than OriginalCompressedSegmentSize in SMB2 COMPRESSION_TRANSFORM_HEADER, the connection MUST be disconnected as specified in section 3.2.7.1 or 3.3.7.1. • Otherwise, DecompressedMessage MUST be appended with Repetitions number of bytes initialized with the character specified in Pattern field. • Otherwise, the data of size specified in Length field MUST be decompressed using the algorithm specified in AlgorithmId field as specified [MS-XCA] section 2. DecompressedMessage MUST be appended with the decompressed data. • RemainingCompressedDataSize MUST be decremented by the size in Length field. • If the size of RemainingCompressedDataSize is greater than the size of SMB2_COMPRESSION_PAYLOAD_HEADER, the receiver MUST repeat step 2. 3. DecompressedMessage MUST be returned.
2020/05/25	In Section 3.1.4.4, Compressing the Message, the following was changed from: 1. The sender MUST initialize RemainingUncompressedDataSize with the size of uncompressed SMB2 message, TotalCompressedDataSize with 0, and CompressedMessage with empty buffer. 2. The message MUST be compressed until RemainingUncompressedDataSize is greater than zero: • If Connection.CompressionIds includes Pattern_V1, message MUST be scanned for data patterns as specified in section 3.1.4.4.1. If the returned FrontDataPattern.Repetitions is greater than zero, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to Pattern_V1. Length MUST be set to the size of SMB2_COMPRESSION_PATTERN_PAYLOAD_V1. CompressedMessage MUST be appended with front data pattern returned by section 3.1.4.4.1. RemainingUncompressedDataSize MUST be decremented by FrontDataPattern.Repetitions. TotalCompressedDataSize MUST be incremented by FrontDataPattern.Repetitions. • If RemainingUncompressedDataSize is greater than 1024, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to first preferred algorithm in Connection.CompressionIds. The data MUST be compressed using the algorithm specified in AlgorithmId as specified in [MS-XCA] section 2. Length MUST be set to the size of the compressed data. A 4-byte field, indicating the size of the original plain text size of the data compressed, MUST be appended to CompressedMessage. CompressedMessage MUST be appended with the compressed data. RemainingUncompressedDataSize MUST be decremented by the size of data before compression. TotalCompressedDataSize MUST be incremented by the size of compressed data. • Otherwise if RemainingUncompressedDataSize is greater than zero, if (size of the uncompressed SMB2 message / RemainingUncompressedDataSize) is greater than 2,

Errata Published*	Description
	CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to NONE. Length MUST be set to RemainingUncompressedDataSize. CompressedMessage MUST be appended with the Remaining uncompressed data. RemainingUncompressedDataSize MUST be decremented by the size of data before compression. TotalCompressedDataSize MUST be incremented by the size of compressed data. ● If BackDataPattern.Repetitions is greater than zero, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to Pattern_V1. Length MUST be set to the size of SMB2_COMPRESSION_PATTERN_PAYLOAD_V1. CompressedMessage MUST be appended with BackDataPattern returned by section 3.1.4.4.1. RemainingUncompressedDataSize MUST be decremented by BackDataPattern.Repetitions. TotalCompressedDataSize MUST be incremented by BackDataPattern.Repetitions. 3. If TotalCompressedDataSize+8 is less than the size of uncompressed SMB2 message, the sender MUST prepend CompressedMessage with first 8 bytes of SMB2 COMPRESSION_TRANSFORM_HEADER. OriginalCompressedSegmentSize MUST be set to the size of uncompressed SMB2 message. 4. Otherwise, the uncompressed SMB2 message is sent. Changed to: 1. The sender MUST initialize RemainingUncompressedDataSize with the size of uncompressed SMB2 message, TotalCompressedDataSize with 0, and CompressedMessage with empty buffer. 2. If RemainingUncompressedDataSize is greater than zero, the message MUST be compressed as follows: ● If Connection.CompressionIds includes Pattern_V1, message MUST be scanned for data patterns as specified in section 3.1.4.4.1. If the returned FrontDataPattern.Repetitions is greater than zero, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to Pattern_V1. Length MUST be set to the size of SMB2_COMPRESSION_PATTERN_PAYLOAD_V1. CompressedMessage MUST be appended with front data pattern returned by section 3.1.4.4.1. RemainingUncompressedDataSize MUST be decremented by FrontDataPattern.Repetitions. TotalCompressedDataSize MUST be incremented by FrontDataPattern.Repetitions. ● If RemainingUncompressedDataSize is greater than 1024, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to first preferred algorithm in Connection.CompressionIds. The data MUST be compressed using the algorithm specified in AlgorithmId as specified in [MS-XCA] section 2. Length MUST be set to the size of the compressed data. A 4-byte field, indicating the size of the original plain text size of the data compressed, MUST be appended to CompressedMessage. CompressedMessage MUST be appended with the compressed data. RemainingUncompressedDataSize MUST be decremented by the size of data before compression. TotalCompressedDataSize MUST be incremented by the size of compressed data. ● Otherwise if RemainingUncompressedDataSize is greater than zero, if (size of the uncompressed SMB2 message / RemainingUncompressedDataSize) is greater than 2, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to NONE. Length MUST be set to RemainingUncompressedDataSize. CompressedMessage MUST be appended with the Remaining uncompressed data. RemainingUncompressedDataSize MUST be decremented by the size of data before compression. TotalCompressedDataSize MUST be incremented by the size of compressed data.

Errata Published*	Description
	• If BackDataPattern.Repetitions is greater than zero, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to Pattern_V1. Length MUST be set to the size of SMB2_COMPRESSION_PATTERN_PAYLOAD_V1. CompressedMessage MUST be appended with BackDataPattern returned by section 3.1.4.4.1. RemainingUncompressedDataSize MUST be decremented by BackDataPattern.Repetitions. TotalCompressedDataSize MUST be incremented by BackDataPattern.Repetitions. • If RemainingUncompressedDataSize is greater than zero, the sender MUST repeat step 2. 3. If TotalCompressedDataSize+8 is less than the size of uncompressed SMB2 message, the sender MUST prepend CompressedMessage with first 8 bytes of SMB2 COMPRESSION_TRANSFORM_HEADER. OriginalCompressedSegmentSize MUST be set to the size of uncompressed SMB2 message. Otherwise, the uncompressed SMB2 message is sent.
2020/05/25	In Section 2.2.42.1, SMB2_COMPRESSION_PAYLOAD_HEADER, the following was changed from: The SMB2_COMPRESSION_PAYLOAD_HEADER is used by the client or server when sending chained compressed payloads. This structure MUST start at an 8-byte aligned boundary relative to the start of the message. This optional structure is only valid for the SMB 3.1.1 dialect<70>. Changed to: The SMB2_COMPRESSION_PAYLOAD_HEADER is used by the client or server when sending chained compressed payloads. This optional structure is only valid for the SMB 3.1.1 dialect<70>.
2020/05/25	In Section 2.2.42.1, SMB2_COMPRESSION_PAYLOAD_HEADER, a new field was added: OriginalPayloadSize (4 bytes): This optional field is present only when AlgorithmId is LZNT1, LZ77, or LZ77+Huffman. The size, in bytes, of the uncompressed payload. Section 3.1.4.4 Compressing the Message was changed from: If IsCompressionSupported is FALSE or Connection.CompressionIds is empty, the sender MUST skip the processing in this section. If Connection.SupportsChainedCompression is TRUE, the sender <77>MUST prepare the compressed message2.2.42 as the following: 1. The sender MUST initialize RemainingUncompressedDataSize with the size of uncompressed SMB2 message, TotalCompressedDataSize with 0, and CompressedMessage with empty buffer. 2. If RemainingUncompressedDataSize is greater than zero, the message MUST be compressed as follows: • If Connection.CompressionIds includes Pattern_V1, message MUST be scanned for data patterns as specified in section 3.1.4.4.1. If the returned FrontDataPattern.Repetitions is greater than zero, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to Pattern_V1. Length MUST be set to the size of SMB2_COMPRESSION_PATTERN_PAYLOAD_V1. CompressedMessage MUST be appended with front data pattern returned by section 3.1.4.4.1. RemainingUncompressedDataSize

Errata Published*	Description
	MUST be decremented by FrontDataPattern.Repetitions. TotalCompressedDataSize MUST be incremented by FrontDataPattern.Repetitions. ● If RemainingUncompressedDataSize is greater than 1024, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to first preferred algorithm in Connection.CompressionIds. The data MUST be compressed using the algorithm specified in AlgorithmId as specified in section 2. Length MUST be set to the size of the compressed data. A 4-byte field, indicating the size of the original plain text size of the data compressed, MUST be appended to CompressedMessage. CompressedMessage MUST be appended with the compressed data. RemainingUncompressedDataSize MUST be decremented by the size of data before compression. TotalCompressedDataSize MUST be incremented by the size of compressed data. ● Otherwise if RemainingUncompressedDataSize is greater than zero, if (size of the uncompressed SMB2 message / RemainingUncompressedDataSize) is greater than 2, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to NONE. Length MUST be set to RemainingUncompressedDataSize. CompressedMessage MUST be appended with the Remaining uncompressed data. RemainingUncompressedDataSize MUST be decremented by the size of data before compression. TotalCompressedDataSize MUST be incremented by the size of compressed data. ● If BackDataPattern.Repetitions is greater than zero, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to Pattern_V1. Length MUST be set to the size of SMB2_COMPRESSION_PATTERN_PAYLOAD_V1. CompressedMessage MUST be appended with BackDataPattern returned by section 3.1.4.4.1. RemainingUncompressedDataSize MUST be decremented by BackDataPattern.Repetitions. TotalCompressedDataSize MUST be incremented by BackDataPattern.Repetitions. ● If RemainingUncompressedDataSize is greater than zero, the sender MUST repeat step 2. 3. If TotalCompressedDataSize+8 is less than the size of uncompressed SMB2 message, the sender MUST prepend CompressedMessage with first 8 bytes of SMB2 COMPRESSION_TRANSFORM_HEADER. OriginalCompressedSegmentSize MUST be set to the size of uncompressed SMB2 message. Otherwise, the uncompressed SMB2 message is sent. Otherwise, the sender SHOULD construct the SMB2 COMPRESSION_TRANSFORM_HEADER specified in section as follows: 1. CompressionAlgorithm MUST be set to one from Connection.CompressionIds. 2. The sender MAY choose to leave the leading portion of the SMB2 message uncompressed and compressing only the trailing portion. 3. The sender MUST perform the following: ● If the entire SMB2 message is being compressed, then set Offset to zero; otherwise, set Offset to the length, in bytes, of the uncompressed part of the message. ● Set OriginalCompressedSegmentSize to the uncompressed length, in bytes, of the portion of the message that is being compressed. The sender MUST compress the data using the CompressionAlgorithm as specified in [MS-XCA] section 2.

Errata Published*	Description
	If the size of the compressed data is less than OriginalCompressedSegmentSize, the sender MUST perform the following: ● If Offset is zero, the sender MUST replace the SMB2 message with the SMB2 COMPRESSION_TRANSFORM_HEADER followed by the compressed SMB2 message. Otherwise, the sender MUST replace the portion of the SMB2 message selected for compression with the compressed part and prepend the SMB2 message with the SMB2 COMPRESSION_TRANSFORM_HEADER. Otherwise, the uncompressed SMB2 message without the SMB2 COMPRESSION_TRANSFORM_HEADER is used. Changed to: If IsCompressionSupported is FALSE or Connection.CompressionIds is empty, the sender MUST skip the processing in this section. If Connection.SupportsChainedCompression is FALSE, the sender SHOULD<77> construct the SMB2 COMPRESSION_TRANSFORM_HEADER specified in section 2.2.42 as follows: 1. CompressionAlgorithm MUST be set to LZNT1, LZ77, or LZ77+Huffman specified in Connection.CompressionIds. 2. The sender MAY choose to leave the leading portion of the SMB2 message uncompressed and compressing only the trailing portion.<78> 3. The sender MUST perform the following: ● If the entire SMB2 message is being compressed, then set Offset to zero; otherwise, set Offset to the length, in bytes, of the uncompressed part of the message. ● Set OriginalCompressedSegmentSize to the uncompressed length, in bytes, of the portion of the message that is being compressed. 4. The sender MUST compress the data using the CompressionAlgorithm as specified in [MS-XCA] section 2. 5. If the size of the compressed data is less than OriginalCompressedSegmentSize, the sender MUST perform the following: ● If Offset is zero, the sender MUST replace the SMB2 message with the SMB2 COMPRESSION_TRANSFORM_HEADER followed by the compressed SMB2 message. ● Otherwise, the sender MUST replace the portion of the SMB2 message selected for compression with the compressed part and prepend the SMB2 message with the SMB2 COMPRESSION_TRANSFORM_HEADER. The compressed SMB2 message is sent. 6. Otherwise, the original, uncompressed SMB2 message without the SMB2 COMPRESSION_TRANSFORM_HEADER is sent. Otherwise, the sender MUST prepare the compressed message as follows: 1. The sender MUST initialize RemainingUncompressedDataSize with the size of uncompressed data, TotalCompressedDataSize with 0, and CompressedMessage with empty buffer.

Errata Published*	Description
	2. The uncompressed data MUST be compressed as follows: ● If Connection.CompressionIds includes Pattern_V1 and RemainingUncompressedDataSize is greater than 32, the uncompressed data MUST be scanned for data patterns as specified in section 3.1.4.4.1. If the returned ForwardDataPattern.Repetitions is greater than zero, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to Pattern_V1. Length MUST be set to the size of SMB2_COMPRESSION_PATTERN_PAYLOAD_V1. CompressedMessage MUST be appended with ForwardDataPattern. RemainingUncompressedDataSize MUST be decremented by ForwardDataPattern.Repetitions. If the returned BackwardDataPattern is not NULL and BackwardDataPattern.Repetitions is greater than zero, RemainingUncompressedDataSize MUST be decremented by BackwardDataPattern.Repetitions. TotalCompressedDataSize MUST be incremented by ForwardDataPattern.Repetitions. ● If RemainingUncompressedDataSize is greater than 1024, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to LZNT1, LZ77, or LZ77+Huffman specified in Connection.CompressionIds. The uncompressed data MUST be compressed using the algorithm specified in AlgorithmId as specified in [MS-XCA] section 2. Length MUST be set to the size of the compressed data. OriginalPayloadSize MUST be set to the size of the uncompressed data. CompressedMessage MUST be appended with the compressed data. RemainingUncompressedDataSize MUST be decremented by the size of data before compression. TotalCompressedDataSize MUST be incremented by the size of compressed data. ● Otherwise if RemainingUncompressedDataSize is greater than zero and (size of the uncompressed SMB2 message / RemainingUncompressedDataSize) is greater than 2, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to NONE. Length MUST be set to RemainingUncompressedDataSize. CompressedMessage MUST be appended with the uncompressed data. RemainingUncompressedDataSize MUST be decremented by the size of data before compression. TotalCompressedDataSize MUST be incremented by the size of compressed data. ● If BackwardDataPattern is not NULL and BackwardDataPattern.Repetitions is greater than zero, CompressedMessage MUST be appended with newly constructed SMB2_COMPRESSION_PAYLOAD_HEADER. AlgorithmId MUST be set to Pattern_V1. Length MUST be set to the size of SMB2_COMPRESSION_PATTERN_PAYLOAD_V1. CompressedMessage MUST be appended with BackwardDataPattern. TotalCompressedDataSize MUST be incremented by BackwardDataPattern.Repetitions. ● If RemainingUncompressedDataSize is greater than zero, the sender MUST repeat step 2. 3. If TotalCompressedDataSize+8 is less than the size of uncompressed SMB2 message, the sender MUST prepend CompressedMessage with first 8 bytes of SMB2 COMPRESSION_TRANSFORM_HEADER. OriginalCompressedSegmentSize MUST be set to the size of uncompressed SMB2 message. The compressed SMB2 message is sent. Otherwise, the original, uncompressed SMB2 message is sent. Section 3.1.4.4.1, Algorithm for Scanning Data Patterns V1, was changed from: The inputs for this algorithm are: InputBuffer: Input data to scan data patterns

Errata Published*	Description
	InputBufferSize: Size of InputBuffer FrontScan: A Boolean value indicating if data is to be scanned forward (TRUE) or backward (FALSE). The output is two DataPatterns of type SMB2_COMPRESSION_PATTERN_PAYLOAD_V12.2.42.2. If the length of InputBuffer is less than or equal to 32, no pattern compression processing is performed. Scan for data patterns by setting FrontScan to TRUE as specified in section . Returned DataPattern MUST be interpreted as FrontDataPattern. FrontDataPattern.Pattern MUST be set to the first byte in InputBuffer. If FrontDataPattern.Repetitions is equal to InputBufferSize, return FrontDataPattern. Otherwise, scan for data patterns by setting FrontScan to FALSE as specified in section 3.1.4.4.1.1. Returned DataPattern MUST be interpreted as BackDataPattern. BackDataPattern.Pattern MUST be set to the last byte in InputBuffer. Return FrontDataPattern and BackDataPattern. Changed to: Construct a new SMB2_COMPRESSION_PATTERN_PAYLOAD_V1 structure, specified in section 2.2.42.2 and scan forward in the buffer for a consecutive series of bytes equal to the first byte: ● For each consecutive byte matched, SMB2_COMPRESSION_PATTERN_PAYLOAD_V1.Repetitions MUST be incremented by 1. ● If none, stop scan. If SMB2_COMPRESSION_PATTERN_PAYLOAD_V1.Repetitions is less than 64, SMB2_COMPRESSION_PATTERN_PAYLOAD_V1.Repetitions MUST be set to 0. SMB2_COMPRESSION_PATTERN_PAYLOAD_V1.Pattern MUST be set to the first byte in the buffer. If SMB2_COMPRESSION_PATTERN_PAYLOAD_V1.Repetitions is equal to the size of the buffer, the processing MUST return the SMB2_COMPRESSION_PATTERN_PAYLOAD_V1 as ForwardDataPattern, and BackwardDataPattern set to NULL. Otherwise, construct a new SMB2_COMPRESSION_PATTERN_PAYLOAD_V1 structure, specified in section 2.2.42.2 and scan backward in the buffer for a consecutive series of bytes equal to the last byte: ● For each consecutive byte matched, SMB2_COMPRESSION_PATTERN_PAYLOAD_V1.Repetitions MUST be incremented by 1. ● If none, stop scan. If SMB2_COMPRESSION_PATTERN_PAYLOAD_V1.Repetitions is less than 64, SMB2_COMPRESSION_PATTERN_PAYLOAD_V1.Repetitions MUST be set to 0. SMB2_COMPRESSION_PATTERN_PAYLOAD_V1.Pattern MUST be set to the last byte in the buffer.

Errata Published*	Description
	The processing MUST return both SMB2_COMPRESSION_PATTERN_PAYLOAD_V1 structures respectively as ForwardDataPattern and BackwardDataPattern. Section 3.1.4.4.1.1, Scan for Data Patterns, was removed. In Section 3.1.5.3, Decompressing the Chained Message, the following was changed from: 1. The receiver MUST initialize RemainingCompressedDataSize with the size of the received compressed SMB2 message and DecompressedMessage with empty buffer. 2. If the size of RemainingCompressedDataSize is greater than the size of SMB2_COMPRESSION_PAYLOAD_HEADER, the compressed message MUST be decompressed as follows: ... ● Otherwise, the data of size specified in Length field MUST be decompressed using the algorithm specified in AlgorithmId field as specified in [MS-XCA] section 2. DecompressedMessage MUST be appended with the decompressed data. Changed to: 1. The receiver MUST initialize RemainingCompressedDataSize with the size of the received compressed data and DecompressedMessage with empty buffer. 2. The compressed data MUST be decompressed as follows: ... ● Otherwise, the data of size specified in Length field MUST be decompressed using the algorithm specified in AlgorithmId field as specified in [MS-XCA] section 2. If the size of the decompressed data is not equal to OriginalPayloadSize, the connection MUST be disconnected as specified in section 3.2.7.1 or section 3.3.7.1. DecompressedMessage MUST be appended with the decompressed data.
2020/05/25	In Section 6, Appendix A: Product Behavior, product behavior note <235> was changed from: <235> Section 3.3.5.4: Windows 10 v1903 and later and Windows Server v1903 and later only set CompressionAlgorithms to the first common algorithm supported by the client and server. Changed to: <235> Section 3.3.5.4: Windows 10 v1903, Windows 10 v1909, Windows Server v1903, and Windows Server v1909 only set CompressionAlgorithms to the first common algorithm supported by the client and server. Windows 10 v2004 and Windows Server v2004 select a common pattern scanning algorithm and the first common compression algorithm, specified in section 2.2.3.1.3, supported by the client and server.

Errata Published*	Description
2020/05/11	In Section 3.3.5.9.12, Handling the SMB2_CREATE_DURABLE_HANDLE_RECONNECT_V2 Create Context, the following was changed from: ● If the request includes the SMB2_DHANDLE_FLAG_PERSISTENT bit in the Flags field of the SMB2_CREATE_DURABLE_HANDLE_RECONNECT_V2 create context, the server MUST look up an existing Open in the GlobalOpenTable by doing a lookup with the CreateGuid of the create context. If the lookup fails, the server SHOULD<292> fail the request with STATUS_OBJECT_NAME_NOT_FOUND and proceed as specified in "Failed Open Handling" in section 3.3.5.9. Changed to: ● If the request includes the SMB2_DHANDLE_FLAG_PERSISTENT bit in the Flags field of the SMB2_CREATE_DURABLE_HANDLE_RECONNECT_V2 create context, TreeConnect.Share.IsCA is TRUE, and Connection.ServerCapabilities includes SMB2_GLOBAL_CAP_PERSISTENT_HANDLES, the server MUST look up an existing Open in the GlobalOpenTable by doing a lookup with the CreateGuid of the create context. If the lookup fails, the server SHOULD<292> fail the request with STATUS_OBJECT_NAME_NOT_FOUND and proceed as specified in "Failed Open Handling" in section 3.3.5.9.
2020/05/11	In Section 3.2.5.5, Receiving an SMB2 TREE_CONNECT Response, step 11 in Product Behavior Note <156> was changed from: 11. The client attempts to establish an alternate channel on each selected interface and address pair. The client will create only a single connection per address pair when the server interface is neither RSS- nor RDMA-capable. Changed to: 11. By default, Windows clients create four connections per RSS-capable address pair or two connections per RDMA-capable address pair or only a single connection when the address pair is neither RSS-capable nor RDMA-capable.
2020/05/11	In Section 3.2.5.3.3, Handling Session Binding, the following was added: ... If SMB2_SESSION_FLAG_IS_GUEST bit is set in the SessionFlags field of the SMB2 SESSION_SETUP Response, the client SHOULD<153> return STATUS_INVALID_NETWORK_RESPONSE to the caller. <246> Section 3.3.5.6: Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2 servers will not reset ResilientOpenScavengerExpiryTime. ... The client MUST ignore the SMB2_SESSION_FLAG_ENCRYPT_DATA bit in the SessionFlags field of the SMB2 SESSION_SETUP Response. ...

Errata Published*	Description
	In Section 3.3.5.5.3, Handling GSS-API Authentication, the following was changed from: 10. If global EncryptData is TRUE, the server MUST do the following: If Connection.ServerCapabilities includes SMB2_GLOBAL_CAP_ENCRYPTION or RejectUnencryptedAccess is TRUE, Changed to: 10. If global EncryptData is TRUE, Connection.Dialect belongs to the SMB 3.x dialect family, Connection.ServerCapabilities includes SMB2_GLOBAL_CAP_ENCRYPTION, RejectUnencryptedAccess is TRUE, and SMB2_SESSION_FLAG_BINDING is not set in the Flags field of the request, the server MUST do the following:
2020/05/11	In Section 3.3.1.12, Per Lease the following was added: ● Lease.FileDeleteOnClose: A Boolean, if set to TRUE, indicating that file deletion on close is pending. In Section 3.3.5.9.7, Handling the SMB2_CREATE_DURABLE_HANDLE_RECONNECT Create Context, step 5 was changed from: 5. If Open.Lease is not NULL and Open.FileName does not match the file name specified in the Buffer field of the SMB2 CREATE request, the server MUST fail the request with STATUS_INVALID_PARAMETER. Changed to: 5. If Open.Lease is not NULL, Open.Lease.FileDeleteOnClose is FALSE, and Open.Lease.FileName does not match the file name specified in the Buffer field of the SMB2 CREATE request, the server MUST fail the request with STATUS_INVALID_PARAMETER. In Section 3.3.5.9.8, Handling the SMB2_CREATE_REQUEST_LEASE Create Context, the following was added: If both SMB2_CREATE_DURABLE_HANDLE_RECONNECT_V2 and SMB2_CREATE_REQUEST_LEASE create contexts are present in the request, they are processed as specified in section 3.3.5.9.12, and this section does not apply. ... ● Lease.FileDeleteOnClose is set to FALSE. ... If Open.Lease is not NULL and CreateOptions field in the CREATE request includes FILE_DELETE_ON_CLOSE, the server MUST set Open.Lease.FileDeleteOnClose to TRUE. The following was changed from: The server MUST attempt to locate a Lease by performing a lookup in the LeaseTable.LeaseList

Errata Published*	Description
	using the LeaseKey in the SMB2_CREATE_REQUEST_LEASE as the lookup key. If a lease is found but Lease.Filename does not match the file name for the incoming request, the request MUST be failed with STATUS_INVALID_PARAMETER. Changed to: The server MUST attempt to locate a Lease by performing a lookup in the LeaseTable.LeaseList using the LeaseKey in the SMB2_CREATE_REQUEST_LEASE as the lookup key. If a lease is found, Lease.FileDeleteOnClose is FALSE, and Lease.Filename does not match the file name for the incoming request, the request MUST be failed with STATUS_INVALID_PARAMETER. In Section 3.3.5.9.11, Handling the SMB2_CREATE_REQUEST_LEASE_V2 Create Context, the following was added: If both SMB2_CREATE_DURABLE_HANDLE_RECONNECT_V2 and SMB2_CREATE_REQUEST_LEASE_V2 create contexts are present in the request, they are processed as specified in section 3.3.5.9.12, and this section does not apply. ... ● Lease.FileDeleteOnClose is set to FALSE. ... If Open.Lease is not NULL and CreateOptions field in the CREATE request includes FILE_DELETE_ON_CLOSE, the server MUST set Open.Lease.FileDeleteOnClose to TRUE. The following paragraph was changed from: The server MUST attempt to locate a Lease by performing a lookup in the LeaseTable.LeaseList using the LeaseKey in the SMB2_CREATE_REQUEST_LEASE_V2 as the lookup key. If a lease is found but Lease.Filename does not match the file name for the incoming request, the request MUST be failed with STATUS_INVALID_PARAMETER. Changed to: The server MUST attempt to locate a Lease by performing a lookup in the LeaseTable.LeaseList using the LeaseKey in the SMB2_CREATE_REQUEST_LEASE_V2 as the lookup key. If a lease is found , Lease.FileDeleteOnClose is FALSE, and Lease.Filename does not match the file name for the incoming request, the request MUST be failed with STATUS_INVALID_PARAMETER. In Section 3.3.5.9.12, Handling the SMB2_CREATE_DURABLE_HANDLE_RECONNECT_V2 Create Context, the following was changed from: ● Open.Lease is not NULL and Open.FileName does not match the file name specified in the Buffer field of the SMB2 CREATE request. Changed to: ● Open.Lease is not NULL, Open.Lease.FileDeleteOnClose is FALSE, and Open.Lease.FileName does not match the file name specified in the Buffer field of the SMB2 CREATE request.

Errata Published*	Description
	In Section 3.3.5.21.1, Handling SMB2_0_INFO_FILE, the following was changed from: If the object store supports security and the information class is FileBasicInformation or FilePipeInformation, and Open.GrantedAccess does not include FILE_WRITE_ATTRIBUTES, the server MUST fail the request with STATUS_ACCESS_DENIED. If the object store supports security and the information class is FileRenameInformation, FileDispositionInformation, or FileShortNameInformation, and Open.GrantedAccess does not include DELETE, the server MUST fail the request with STATUS_ACCESS_DENIED. If the object store supports security and the information class is FileFullEaInformation, and Open.GrantedAccess does not include FILE_WRITE_EA, the server MUST fail the request with STATUS_ACCESS_DENIED. If the object store supports security and the information class is FileFullEaInformation and the EA buffer in the Buffer field is not in a valid format, the server MUST fail the request with STATUS_EA_LIST_INCONSISTENT. If the object store supports security and the information class is FileAllocationInformation, FileEndOfFileInformation, or FileValidDataLengthInformation, and Open.GrantedAccess does not include FILE_WRITE_DATA, the server MUST fail the request with STATUS_ACCESS_DENIED. ... Otherwise, the server MUST initialize an SMB2 SET_INFO Response following the syntax given in section 2.2.40. If the underlying object store returns successfully, the information class is FileRenameInformation, Connection.Dialect is "2.1" or belongs to the SMB 3.x dialect family, the server supports leasing, and Open.Lease is not NULL, the server MUST update Open.Lease.Filename to the new name for the file. Changed to: If the object store supports security and FileInfoClass is FileBasicInformation or FilePipeInformation, and Open.GrantedAccess does not include FILE_WRITE_ATTRIBUTES, the server MUST fail the request with STATUS_ACCESS_DENIED. If the object store supports security and FileInfoClass is FileRenameInformation, FileDispositionInformation, or FileShortNameInformation, and Open.GrantedAccess does not include DELETE, the server MUST fail the request with STATUS_ACCESS_DENIED. If the object store supports security and FileInfoClass is FileFullEaInformation, and Open.GrantedAccess does not include FILE_WRITE_EA, the server MUST fail the request with STATUS_ACCESS_DENIED. If the object store supports security and FileInfoClass is FileFullEaInformation and the EA buffer in the Buffer field is not in a valid format, the server MUST fail the request with STATUS_EA_LIST_INCONSISTENT. If the object store supports security and FileInfoClass is FileAllocationInformation, FileEndOfFileInformation, or FileValidDataLengthInformation, and Open.GrantedAccess does not

Errata Published*	Description
	include FILE_WRITE_DATA, the server MUST fail the request with STATUS_ACCESS_DENIED. ... If the underlying object store returns successfully, FileInfoClass is FileDispositionInformation, Connection.Dialect is not "2.0.2" , and Open.Lease is not NULL, the server MUST set Open.Lease.FileDeleteOnClose to TRUE. If the underlying object store returns successfully, FileInfoClass is FileRenameInformation, Connection.Dialect is not "2.0.2", and Open.Lease is not NULL, the server MUST update Open.Lease.Filename to the new name for the file and Open.Lease.FileDeleteOnClose to FALSE.
2020/05/11	In Section 3.2.3, Initialization, the following was added: If the client implements the SMB 2.1 dialect or SMB 3.x dialect family: The following was changed from: ClientGuid: If implemented, MUST be set to a newly generated GUID. Changed to: ClientGuid: MUST be set to a newly generated GUID. In Section 3.2.4.2.2.2, SMB2-Only Negotiate, the following was changed from: ● If the client implements the SMB 2.1 or SMB 3.x dialect, ClientGuid SHOULD be set to the Guid provided by the application. Otherwise, it MUST be set to 0. The client MUST set Connection.ClientGuid to the ClientGuid initialized above. <106> Section 3.2.4.2.2.2: Windows 7 without [MSKB-3002286] sets ClientGuid to the global ClientGuid value. Changed to: ● If the client implements the SMB 2.1 or SMB 3.x dialect, ClientGuid MUST be set to the global ClientGuid value. Otherwise, it MUST be set to 0. The client MUST set Connection.ClientGuid to the ClientGuid initialized above. In Section 3.3.1.5, Global, the following was added: ● GlobalClientTable: A list of clients, indexed by the ClientGuid as specified in section 3.3.1.16. A new Section 3.3.1.16, Per Client, was added: If the server implements the SMB 3.x dialect family, it implements the following: Client.ClientGuid: An identifier of the client machine. Client.Dialect: The dialect of SMB2 negotiated with the client. This value MUST be either "2.0.2",

Errata Published*	Description
	"2.1", "3.0", "3.0.2", or "3.1.1". In Section 3.3.3, Initialization, the following was added: • GlobalClientTable MUST be set to an empty list. In Section 3.3.5.5.3, Handling GSS-API Authentication, the following was changed from: The server MUST look up all existing connections from the client in the global ConnectionList where Connection.ClientGuid matches Session.Connection.ClientGuid. For any matching Connection, if Connection.Dialect is not the same as Session.Connection.Dialect, the server SHOULD close the newly created Session, as specified in section 3.3.4.12, by providing Session.SessionGlobalId as the input parameter, and fail the session setup request with STATUS_USER_SESSION_DELETED. <243> Section 3.3.5.5.3: Windows Vista SP1, Windows Server 2008, Windows 7, and Windows Server 2008 R2 servers do not fail the request if dialects do not match. Changed to: If the server implements the SMB 3.x dialect family and Session.Connection.Dialect is not "2.0.2", the server MUST look up a client entry in GlobalClientTable using Session.Connection.ClientGuid. If no entry is found, the server MUST create a new Client entry by setting Client.ClientGuid to Session.Connection.ClientGuid and Client.Dialect to Session.Connection.Dialect. The server MUST insert the Client entry into GlobalClientTable. If an entry is found and Client.Dialect is not equal to Session.Connection.Dialect, the server MUST close the newly created Session, as specified in section 3.3.4.12, by providing Session.SessionGlobalId as the input parameter, and fail the session setup request with STATUS_USER_SESSION_DELETED. In Section 3.3.7.1, Handling Loss of a Connection, the following was added: If the server implements the SMB 3.x dialect family, the server MUST enumerate all connections in ConnectionList using the removed Connection.ClientGuid where Connection.Dialect is not "2.0.2". If no Connection entry is found, the server MAY remove the Client entry identified by Connection.ClientGuid from GlobalClientTable.
2020/04/13	In Section 3.3.5.6, Receiving an SMB2 LOGOFF Request, the following was changed from: When the server receives a request with an SMB2 header with a Command value equal to SMB2 LOGOFF, message handling MUST proceed as follows. The server MUST locate the session being logged off, as specified in section 3.3.5.2.9. The server MUST remove this session from the GlobalSessionTable and also from the Connection.SessionTable, and deregister the session by invoking the <247>event specified in [MS-SRVS] section 3.1.6.3, providing Session.SessionGlobalId as input parameter. ServerStatistics.sts0_sopens MUST be decreased by 1. The server MUST close every Open in Session.OpenTable of the old session, where Open.IsDurable is FALSE and Open.IsResilient is FALSE, as specified in section 3.3.4.17. For all opens in Session.OpenTable where Open.IsDurable is TRUE or Open.IsResilient is TRUE, the server MUST set Open.Session, Open.Connection, and Open.TreeConnect to NULL.Any tree connects in Session.TreeConnectTable of the old session MUST be deregistered by invoking the event specified in [MS-SRVS] section 3.1.6.7, providing the tuple <TreeConnect.Share.ServerName, TreeConnect.Share.Name> and TreeConnect.TreeGlobalId as input parameters, and each of them MUST be freed. For each deregistered TreeConnect, TreeConnect.Share.CurrentUses MUST be decreased by 1.

Errata Published*	Description
	If Connection.Dialect belongs to the SMB 3.x dialect family, the server MUST remove the session from each Channel.Connection.SessionTable in Session.ChannelList. All channels in Session.ChannelList MUST be removed and freed. The server MUST construct an SMB2 LOGOFF Response with a status code of STATUS_SUCCESS, following the syntax specified in section 2.2.8, and send it to the client. The session itself is then freed. Changed to: When the server receives a request with an SMB2 header with a Command value equal to SMB2 LOGOFF, message handling MUST proceed as follows. The server MUST locate the session being logged off, as specified in section 3.3.5.2.9. For each Open in Session.OpenTable, the server MUST perform the following: ● If Open.IsResilient is TRUE, the server MUST do the following: ● The server MUST set Open.Session, Open.Connection, and Open.TreeConnect to NULL. ● The server MUST set Open.ResilientOpenTimeout to the current time plus Open.ResiliencyTimeOut. ● The server SHOULD<247> start or reset the Resilient Open Scavenger Timer, as specified in section 3.3.2.4, under the following conditions: ● If the Resilient Open Scavenger Timer is not already active. ● If the Resilient Open Scavenger Timer is active and ResilientOpenScavengerExpiryTime is greater than Open.ResilientOpenTimeOut. In both of the preceding cases, the server MUST set the timer to expire at Open.ResilientOpenTimeOut and MUST set ResilientOpenScavengerExpiryTime to Open.ResilientOpenTimeOut. ● If Open.IsDurable is TRUE, the server MUST do the following: ● The server MUST set Open.Session, Open.Connection, and Open.TreeConnect to NULL. ● The server MUST set Open.DurableOpenScavengerTimeOut to the current time plus Open.DurableOpenTimeOut. ● The server MUST start the Durable Open Scavenger Timer, as specified in section 3.3.2.2. ● Otherwise the server MUST close the Open as specified in section 3.3.4.17. Any tree connects in Session.TreeConnectTable of the old session MUST be deregistered by invoking the event specified in [MS-SRVS] section 3.1.6.7, providing the tuple

Errata Published*	Description
	<TreeConnect.Share.ServerName, TreeConnect.Share.Name> and TreeConnect.TreeGlobalId as input parameters, and each of them MUST be freed. For each deregistered TreeConnect, TreeConnect.Share.CurrentUses MUST be decreased by 1. If Connection.Dialect belongs to the SMB 3.x dialect family, the server MUST remove the session from each Channel.Connection.SessionTable in Session.ChannelList. All channels in Session.ChannelList MUST be removed and freed. The server MUST remove this session from the GlobalSessionTable and also from the Connection.SessionTable, and deregister the session by invoking the event specified in [MS-SRVS] section 3.1.6.3, providing Session.SessionGlobalId as input parameter. ServerStatistics.sts0_sopens MUST be decreased by 1. The server MUST construct an SMB2 LOGOFF Response with a status code of STATUS_SUCCESS, following the syntax specified in section 2.2.8, and send it to the client. The session itself is then freed. <247> Section 3.3.5.6: Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2 servers will not reset ResilientOpenScavengerExpiryTime.
2020/04/13	In Section 3.2.5.14.11, Handling a Network Interfaces Response, the following was changed from: The client MUST extract IPv4Address and IPv6Address addresses from each NETWORK_INTERFACE_INFO structure and MUST set Connection.Server.AddressList to the received values. The client MUST return the list of network interfaces received from the server to the calling application. Changed to: If the Status field of the SMB2 header of the response indicates an error, the client MUST return the received status code to the calling application. If the Status field of the SMB2 header of the response indicates success, the client MUST extract IPv4Address and IPv6Address addresses from each NETWORK_INTERFACE_INFO structure and MUST set Connection.Server.AddressList to the received values. In Section 3.3.5.15.11, Handling a Query Network Interface Request, the following was changed from: The server MUST enumerate the local network interfaces in an implementation-specific manner. For each IP address in each network interface, the server MUST construct a NETWORK_INTERFACE_INFO structure as specified in section 2.2.32.5, with the following values: Changed to: This section applies only to servers that implement the SMB 3.x dialect family. When the server receives a request with an SMB2 header with a Command value equal to SMB2 IOCTL and a CtlCode of FSCTL_QUERY_NETWORK_INTERFACE_INFO, message handling proceeds as follows:

Errata Published*	Description
	If Connection.Dialect does not belong to the SMB 3.x dialect family or Connection.ServerCapabilities does not include SMB2_GLOBAL_CAP_MULTI_CHANNEL, the server MAY fail the request with STATUS_NOT_SUPPORTED. Otherwise, the server MUST enumerate the local network interfaces in an implementation-specific manner. For each IP address in each network interface, the server MUST construct a NETWORK_INTERFACE_INFO structure as specified in section 2.2.32.5, with the following values:

*Date format: YYYY/MM/DD

[MS-SMBD]: SMB2 Remote Direct Memory Access (RDMA) Transport Protocol

This topic lists the Errata found in [MS-SMBD] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

Errata below are for Protocol Document Version V12.0 – 2018/09/12.

Errata Published*	Description
2019/11/11	In Section 2.2.3.1, Buffer Descriptor V1 Structure, changed the structure name from SMB_DIRECT_BUFFER_DESCRIPTOR_1 to SMB_DIRECT_BUFFER_DESCRIPTOR_V1. Changed from: The SMB_DIRECT_BUFFER_DESCRIPTOR_1 structure represents a registered RDMA buffer and is used to Advertise the source and destination of RDMA Read and RDMA Write operations, respectively. The upper layer optionally embeds one or more of these structures in its payload when requesting RDMA direct placement of peer data via the protocol. ... Changed to: The SMB_DIRECT_BUFFER_DESCRIPTOR_V1 structure represents a registered RDMA buffer and is used to Advertise the source and destination of RDMA Read and RDMA Write operations, respectively. The upper layer optionally embeds one or more of these structures in its payload when requesting RDMA direct placement of peer data via the protocol. ...
2019/11/11	In Section 3.1.5.1, Sending Upper Layer Messages, the following was changed from: ... The new messages to be sent, if any, MUST be appended to the list of messages in the Connection.SendQueue. If there are no messages to be sent and Connection.SendImmediate is TRUE, a newly constructed Data Transfer Message MUST be added to Connection.SendQueue. • the credit processing specified in section 3.1.5.9 MUST be performed, and the CreditsGranted field of the first message in Connection.SendQueue MUST be incremented by the number of new credits returned. For each message in Connection.SendQueue: • If Connection.SendCredits is 0, stop processing messages, and break the loop. • If Connection.SendCredits is 1 and the CreditsGranted field of the message is 0, then at least one credit MUST be granted to the peer to prevent deadlock. If the processing specified in section 3.1.5.9 returns zero, stop processing Sends, and break the loop. Otherwise, increment the CreditsGranted field of the current first message in Connection.SendQueue by the number of

Errata Published*	Description
	new credits returned. • The first message MUST be removed from Connection.SendQueue. • The value of Connection.SendCredits MUST be decremented by one. • The value of the CreditsRequested field of the message MUST be set to Connection.SendCreditTarget. • If Connection.KeepaliveRequested is "PENDING", the Flags field of the message MUST be set to SMB_DIRECT_RESPONSE_REQUESTED, Connection.KeepaliveRequested MUST be set to "SENT", and the Idle Connection Timer SHOULD<3> be set to an implementation-specific value. Otherwise, the Flags field of the message MUST be set to 0x0000. • If the message to be sent was provided with an optional remote memory token to be invalidated on the receiving peer, the token SHOULD be provided in an implementation-specific manner to the RDMA provider when sending. If sending of remote invalidation is not supported by the RDMA provider, the token MAY be ignored. • The message MUST be sent on the connection in an implementation-specific manner, and any error MUST be returned to the caller. • If Connection.SendQueue is empty, Connection.SendImmediate MUST be set to FALSE and success MUST be returned to the caller. Changed to: ... For each message in Connection.SendQueue: • If Connection.SendCredits is 0, stop processing. • If CreditsGranted field of the first message in Connection.SendQueue is zero, the credit processing specified in section 3.1.5.9 MUST be performed, and the CreditsGranted field of the message MUST be set to the number of new credits returned. • If Connection.SendCredits is 1 and the CreditsGranted field of the message is 0, stop processing. • The first message MUST be removed from Connection.SendQueue. • The value of Connection.SendCredits MUST be decremented by one. • The value of the CreditsRequested field of the message MUST be set to Connection.SendCreditTarget. • If Connection.KeepaliveRequested is "PENDING", the Flags field of the message MUST be set to SMB_DIRECT_RESPONSE_REQUESTED, Connection.KeepaliveRequested MUST be set to "SENT", and the Idle Connection Timer SHOULD<3> be set to an implementation-specific value. Otherwise, the Flags field of the message MUST be set to 0x0000. • If the message to be sent was provided with an optional remote memory token to be invalidated on the receiving peer, the token SHOULD be provided in an implementation-specific manner to the RDMA provider when sending. If sending of remote invalidation is not supported by the RDMA provider, the token MAY be ignored. • The message MUST be sent on the connection in an implementation-specific manner. • Connection.SendImmediate MUST be set to FALSE. In Section 3.1.5.8, Receiving a Data Transfer Message, the following was changed from: ... If Connection.SendQueue is empty, the credit processing specified in section 3.1.5.9 MUST be performed. If the number of new credits returned is greater than zero, the receiver MUST set Connection.SendImmediate to TRUE and MUST promptly send a Data Transfer message on the Connection, as specified in section 3.1.5.1.

Errata Published*	Description
	... Changed to: ... If Connection.SendQueue is empty, the credit processing specified in section 3.1.5.9 MUST be performed. If the number of new credits returned is greater than zero, the receiver MUST promptly send a newly constructed Data Transfer message with its CreditsGranted field set to the number of new credits on the Connection, as specified in section 3.1.5.1. ...

*Date format: YYYY/MM/DD

[MS-SPNG]: Simple and Protected GSS-API Negotiation Mechanism (SPNEGO) Extension

This topic lists the Errata found in [MS-SPNG] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-SQOS]: Storage Quality of Service Protocol

This topic lists the Errata found in [MS-SQOS] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-SSTP]: Secure Socket Tunneling Protocol (SSTP)

This topic lists the Errata found in [MS-SSTP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-SSTR]: Smooth Streaming Protocol

This topic lists the Errata found in the [MS-SSTR] document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

March 16, 2018 - Download

Errata below are for Protocol Document Version V8.0 – 2019/03/13.

Errata Published*	Description
2020/07/06	In Section 1.5 Prerequisites/Preconditions, added reference to the amendment for HEVC. Changed from: It is also assumed that the client is integrated with a higher-layer implementation that supports any media formats that are used and can otherwise play the media that is transmitted by the server.<1> <1> Section 1.5: The Smooth Streaming Protocol is supported… Changed to: It is also assumed that the client is integrated with a higher-layer implementation that supports any media formats that are used and can otherwise play the media that is transmitted by the server.<1><2> <1> Section 1.5: For requirements to enable cloud-based Smooth Streaming of High Efficiency Video Coding (HEVC) encoded video see the amendment for HEVC [MSDOCS-SSTR-HEVC]. <2> Section 1.5: The Smooth Streaming Protocol is supported…

*Date format: YYYY/MM/DD

[MS-SWN]: Service Witness Protocol

This topic lists the Errata found in [MS-SWN] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

Errata below are for Protocol Document Version V11.0 – 2018/09/12.

Errata Published*	Description
2019/02/19	In Section 7, Appendix B: Product Behavior Product Behavior, note 2 has been changed from: <2> Section 3.1.3: Windows Server 2012 sets this value to 0x00010001. Windows Server 2012 R2, Windows Server 2016, Windows Server operating system, and Windows Server 2019 set this value to 0xFFFFFFFF. Changed to: <2> Section 3.1.3: Windows Server 2012 sets this value to 0x00010001. Windows Server 2012 R2, Windows Server 2016, Windows Server operating system, and Windows Server 2019 set this value to 0x00020000.

*Date format: YYYY/MM/DD

[MS-TCC]: Tethering Control Channel Protocol

This topic lists the Errata found in [MS-TCC] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-TDS]: Tabular Data Stream Protocol

This topic lists the Errata found in [MS-TDS] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

March 20, 2017 - Download

August 21, 2017 - Download

December 1, 2017 - Download

September 12, 2018 - Download

October 14, 2019 - Download

June 15, 2020 - Download

[MS-TLSP]: Transport Layer Security (TLS) Profile

This topic lists the Errata found in [MS-TLSP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

September 15, 2017 - Download

[MS-TPMVSC]: Trusted Platform Module (TPM) Virtual Smart Card Management Protocol

This topic lists the Errata found in [MS-TPMVSC] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-TSCH]: Task Scheduler Service Remoting Protocol

This topic lists the Errata found in [MS-TSCH] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

[MS-TSGU]: Terminal Services Gateway Server Protocol

This topic lists the Errata found in [MS-TSGU] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

December 1, 2017 - Download

September 12, 2018 - Download

Errata below are for Protocol Document Version V39.0 – 2018/09/12.

Errata Published*	Description
2019/10/28	In Section 3.1.1, Abstract Data Model, changed HTTP_CHANNEL_REQUEST to HTTP_CHANNEL_PACKET in the Target server names and Channel id element descriptions. Changed from: Target server names: An array of alias names for a target server. A target server alias name is a string of Unicode characters. The server name applies to the machine to which the RDG server connects.<23> ... • For HTTP transport, this is initialized when the RDG server receives an HTTP_CHANNEL_REQUEST from the RDG client. ... Channel id: An unsigned long representing the channel identifier for tracking purposes on the RDG server. The Channel id, which is then generated on the server, is stored by the RDG server and RDG client and can later be used for subsequent channel-related calls.<25> ... • For HTTP transport, this is generated after the RDG server receives HTTP_CHANNEL_REQUEST …. Changed to: Target server names: An array of alias names for a target server. A target server alias name is a string of Unicode characters. The server name applies to the machine to which the RDG server connects.<23> ... • For HTTP transport, this is initialized when the RDG server receives an HTTP_CHANNEL_PACKET (section 2.2.10.2) from the RDG client. ...

Errata Published*	Description
	Channel id: An unsigned long representing the channel identifier for tracking purposes on the RDG server. The Channel id, which is then generated on the server, is stored by the RDG server and RDG client and can later be used for subsequent channel-related calls.<25> ... ● For HTTP transport, this is generated after the RDG server receives HTTP_CHANNEL_PACKET. ...
2019/10/28	In Section 2.2.9.2.1.1, TSG_PACKET_HEADER, changed the field names ComponentID to ComponentId and PacketID to PacketId. Changed from: The TSG_PACKET_HEADER structure contains information about the ComponentID and PacketID fields of the TSG_PACKET structure. The value of PacketID in TSG_PACKET MUST be set to TSG_PACKET_TYPE_HEADER. ... Changed to: The TSG_PACKET_HEADER structure contains information about the ComponentId and PacketId fields of the TSG_PACKET structure. The value of PacketId in TSG_PACKET MUST be set to TSG_PACKET_TYPE_HEADER. ... In Section 3.5.1, Abstract Data Model, changed the structure name AUTHENTICATION_COOKIE_DATA to AUTHN_COOKIE_DATA in the UDPAuthCookie description. Changed from: ... UDPAuthCookie: A signed and encoded byte BLOB containing an AUTHENTICATION_COOKIE_DATA structure. ... Changed to: ... UDPAuthCookie: A signed and encoded byte BLOB containing an AUTHN_COOKIE_DATA structure. ... In Section 3.7.1, Abstract Data Model, changed the structure name AUTHENTICATION_COOKIE_DATA to AUTHN_COOKIE_DATA in the UDPAuthCookie description. Changed from: UDPAuthCookie: A signed and encoded byte BLOB containing an AUTHENTICATION_COOKIE_DATA structure. ... Changed to: UDPAuthCookie: A signed and encoded byte BLOB containing an AUTHN_COOKIE_DATA structure. ...

Errata Published*	Description
	In Section 4.3.1, Normal Scenario, changed the structure name AUTHENTICATION_COOKIE_DATA to AUTHN_COOKIE_DATA and the ADM element name AUTHENTICATION_COOKIE_DATA.szServerName to AUTHN_COOKIE_DATA.szServerName. Changed from: .. 6. The RDG server decrypts the packet received with DTLS. The RDG server decodes the message and verifies the signature on the decoded message. The RDG server maps the decoded message to the AUTHENTICATION_COOKIE_DATA structure. 7. The RDG server connects to the target server specified in the ADM element AUTHENTICATION_COOKIE_DATA.szServerName. ... Changed to: ... 6. The RDG server decrypts the packet received with DTLS. The RDG server decodes the message and verifies the signature on the decoded message. The RDG server maps the decoded message to the AUTHN_COOKIE_DATA structure. 7. The RDG server connects to the target server specified in the ADM element AUTHN_COOKIE_DATA.szServerName....

*Date format: YYYY/MM/DD

[MS-TSTS]: Terminal Services Terminal Server Runtime Interface Protocol

This topic lists the Errata found in [MS-TSTS] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

Errata below are for Protocol Document Version V26.0 – 2018/09/12.

Errata Published*	Description
2019/04/15	In Section 3.10.4.1.1, RpcShadow2 (Opnum 0), the format of the pszInvitation field has been clarified. In addition, a reference to a Windows platform-specific API has been removed and substituted with a link to MS-RAI Section 2.2.2. Changed from: pszInvitation: The output data containing the invitation string for the shadow session. The data returned is an invitation string in an XML format that can be used with the Windows Desktop Sharing API IRDPSRAPIViewer::Connect method to connect to the session running in the target session (specified by TargetSessionId). The caller must allocate a buffer to hold this data and specify the size of the buffer in cchInvitation. Changed to: pszInvitation: The output data containing the invitation string for the shadow session. The data returned is a Unicode string in the XML format specified in [MS-RAI] section 2.2.2 that can be used to connect to a session running in the target session (specified by TargetSessionId). The caller must allocate a buffer to hold this data and specify the size of the buffer in cchInvitation.

*Date format: YYYY/MM/DD

[MS-TSWP]: Terminal Services Workspace Provisioning Protocol

This topic lists the Errata found in [MS-TSWP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 12, 2018 - Download

[MS-UAMG]: Update Agent Management Protocol

This topic lists the Errata found in [MS-UAMG] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

RSS
Atom

Errata are subject to the same terms as the Open Specifications documentation referenced.

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-UCODEREF]: Windows Protocols Unicode Reference

This topic lists the Errata found in [MS-UCODEREF] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-VAPR]: Virtual Application Publication and Reporting (App-V) Protocol

This topic lists the Errata found in [MS-VAPR] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-VHDX]: Virtual Hard Disk v2 (VHDX) File Format

This topic lists the Errata found in [MS-VHDX] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

RSS
Atom

Errata are subject to the same terms as the Open Specifications documentation referenced.

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

Errata below are for Protocol Document Version V4.0 – 2018/09/12.

Errata Published*	Description
2020/05/25	In Section 2.2.2, Headers, the following was changed from: ... Reserved (4016 bytes): MUST be set to 0 and ignored. The LogLength and LogOffset fields specify the byte offset in the file and the length of the log. These values MUST be multiples of 1 MB and LogOffset MUST be at least 1 MB. The log MUST NOT overlap any other structures. The space between a 4-KB structure containing header data and a 64-KB alignment boundary for the header is reserved. Changed to: ... LogLength (4 bytes): A 32-bit unsigned integer. Specifies the size, in bytes of the log. This value MUST be a multiple of 1MB. LogOffset (8 bytes): A 64-bit unsigned integer. Specifies the byte offset in the file of the log. This value MUST be a multiple of 1MB. The log MUST NOT overlap any other structures. Reserved (4016 bytes): MUST be set to 0 and ignored. The space between a 4-KB structure containing header data and a 64-KB alignment boundary for the header is reserved.

*Date format: YYYY/MM/DD

[MS-W32T]: W32Time Remote Protocol

This topic lists the Errata found in [MS-W32T] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-WCCE]: Windows Client Certificate Enrollment Protocol

This topic lists the Errata found in [MS-WCCE] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

Errata below are for Protocol Document Version V43.0 – 2018/09/12.

Errata Published*	Description
2020/08/17	In Section 3.2.1.4.3.2.20 PropID = 0x00000014 (CR_PROP_CRLSTATE) "CA CRL State", clarified processing rules to better reflect the behavior that occurs when the client requests the CRL status of all CA signing certificates. Changed from: The CA MUST do the following for each one of the rows in Signing_Cert table: • The CA MUST evaluate the certificate (1) status stored in the Signing_Cert_Certificate column by building its chain based on the specification defined in [RFC3280]. • If the certificate (1) is not valid the CA uses one of the status codes in the following table as the status for this signing certificate. • If the signing certificate is valid, the CA MUST evaluate the base CRL stored in the CRL_Raw_CRL column of the CRL table row where the value of CRL_Name_Id is equal to the row of the preceding signing certificate and verify that it was signed by the key (2) associated with the signing certificate stored in the Signing_Cert_Certificate column. If the signature does not match to the public key of the signing certificate, then the CA MUST return the status 0x01 as specified in the following table. • If the signing certificate is valid and its associated key (2) was used to sign the base CRL stored in the same row, the CA MUST return 0x03 as the status for this signing certificate. Changed to: The CA MUST do the following for each one of the rows in Signing_Cert table: • The CA MUST evaluate the certificate (1) status stored in the Signing_Cert_Certificate column by building its chain based on the specification defined in [RFC3280]. • If the signing certificate is revoked, the CA MUST return the status CA_DISP_REVOKED. • If the certificate (1) index (identified by the Signing_Cert_Certificate column) does not match the key (2) index, the CA MUST return the status CA_DISP_ERROR. • If the certificate (1) index (identified by Signing_Cert_Certificate column) matches the key (2) index, the CA MUST return the status CA_DISP_VALID.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td>2020/08/17</td><td>In Section 3.2.1.4.3.2 ICertRequestD2::GetCAProperty (Opnum 7), revised property index value-descriptions for PropIDs in the last table of this section.

Changed from:

<table>
<tr><th>PropID</th><th>PropIndex MUST be</th></tr>
<tr><td>0x12</td><td>The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.</td></tr>
<tr><td>0x13</td><td>ANY</td></tr>
<tr><td>0x14</td><td>ANY</td></tr>
<tr><td>0x1b</td><td>ANY</td></tr>
<tr><td>0x1f</td><td>The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.</td></tr>
<tr><td>0x20</td><td>The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.</td></tr>
<tr><td>0x25</td><td>ANY</td></tr>
<tr><td>0x26</td><td>ANY</td></tr>
<tr><td>0x27</td><td>ANY</td></tr>
<tr><td>0x2B</td><td>The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.</td></tr>
</table>

Changed to:

<table>
<tr><th>PropID</th><th>PropIndex MUST be</th></tr>
<tr><td>0x12</td><td>The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table. An index of 0xFFFFFFFF is allowed and indicates the maximum valid index.</td></tr>
<tr><td>0x13</td><td>ANYThe minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.</td></tr>
<tr><td>0x14</td><td>ANYThe minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.</td></tr>
<tr><td>0x1b</td><td>ANYThe minimum index is 0. The maximum index is one less than value of the Config_CA_KRA_Cert_Count datum.</td></tr>
<tr><td>0x1f</td><td>The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table. An index of 0xFFFFFFFF is allowed and indicates the maximum valid index.</td></tr>
<tr><td>0x20</td><td>The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table. An index of 0xFFFFFFFF is allowed and indicates the maximum valid index.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><td>0x25</td><td>ANYThe index corresponds to a particular CA signing certificate. Since the last CA signing certificate cannot have a forward cross certificate, the minimum index is 0 and the maximum index is two less than the count of rows in the Signing_Cert table.</td></tr>
<tr><td>0x26</td><td>ANYThe index corresponds to a particular CA signing certificate. Since the first CA signing certificate cannot have a backward cross certificate, the minimum index is 1 and the maximum index is one less than the count of rows in the Signing_Cert table.</td></tr>
<tr><td>0x27</td><td>ANYThe minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.</td></tr>
<tr><td>0x2B</td><td>The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table. An index of 0xFFFFFFFF is allowed and indicates the maximum valid index.</td></tr>
</table>
</td></tr>
<tr><td>2020/08/17</td><td>In Section 3.2.1.4.3.2.20,clarified the client request for all CA signing certificates. Also corrected the processing instruction to specify the return value from the CA as 0x03 when the signing certificate is valid and its associated key was used to sign the base CRL stored in the same row.

Changed from:

"The client has requested to identify which signing certificate is associated with the key (2) used to publish CRLs.
The CA MUST do the following for each one of the rows in Signing_Cert table:"
. . .

● "If the signing certificate is valid and its associated key was used to sign the base CRL stored in the same row, the CA MUST return0x00 as the status for this signing certificate."

Changed to:

"The client has requested the CA signing certificate status for all CRLs."
The CA MUST do the following for each one of the rows in Signing_Cert table:"
. . .
● "If the signing certificate is valid and its associated key was used to sign the base CRL stored in the same row, the CA MUST return0x03 as the status for this signing certificate."</td></tr>
<tr><td>2020/08/03</td><td>In Section 3.2.1.4.3.2, ICertRequestD2::GetCAProperty (Opnum 7), added missing PropID entry value (0x2D) at the end of last Table in this section, which defines values that MUST be set for PropIndex and PropType parameters for each property value passed via the PropID parameter.

Changed from:

<table>
<tr><td>0x2C</td><td>0x00000000</td><td>0x00000004</td></tr>
</table>

Changed to:

<table>
<tr><td>0x2C</td><td>0x00000000</td><td>0x00000004</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><td>0x2D</td><td>0x00000000</td><td>0x00000004</td></tr>
</table>
</td></tr>
<tr><td>2020/08/03</td><td>In Section 3.2.1.4.3.2.29, PropID = 0x0000001D (CR_PROP_TEMPLATES) "Configured Certificate Templates", revised the processing instructions to specify an updated server return value for the PropID = 0x0000001D property in the GetCAProperty method, consisting of a string containing pairs of the template name and OID separated by new lines.

Changed from:

The client requested to know the list of certificate templates that are configured for this CA.
The server MUST return an empty CERTTRANSBLOB (section 2.2.2.2) structure.

Changed to:

The client requested to know the list of certificate templates that are configured for this CA.
The server MUST return a string containing the list of templates supported by this CA, with one pair of name and string OID for each template and separated by new lines, as in the format that follows:
"name1\nOID1\nname2\OID2...\nnameN\nOIDN\n\0"If the template does not have an associated OID (Win2k domain), there will be an empty string in its place.</td></tr>
<tr><td>2019/12/16</td><td>In Section 3.1.2.4.2.2.2.8 , Certificate.Template.msPKI-Private-Key-Flag, added missing 'CT_FLAG_HELLO_LOGON_KEY' flag and description to the processing rules table. Also added new informative reference [MSDOCS-WHfB] to the description for the missing flag

Changed from:

<table>
<tr><td>0x000001000
CT_FLAG_ATTEST_PREFERRED
*</td><td>This flag instructs the client to generate a certificate request as explained in section 3.1.1.4.3.4.1.1 if the Client_HardwareKeyInfo and Client_KeyAttestationStatement ADM elements are not empty (as described in section 3.1.2.4.2.2.2.2).</td></tr>
</table>

Changed to:

<table>
<tr><td>0x000001000
CT_FLAG_ATTEST_PREFERRED
*</td><td>This flag instructs the client to generate a certificate request as explained in section 3.1.1.4.3.4.1.1 if the Client_HardwareKeyInfo and Client_KeyAttestationStatement ADM elements are not empty (as described in section 3.1.2.4.2.2.2.2).</td></tr>
<tr><td>0x00200000</td><td>This flag instructs the client to generate a certificate request for the Windows Hello</td></tr>
</table>
</td></tr>
</table>

Errata Published*	Description	
	CT_FLAG_HELLO_LOGON_KEY *	Logon key. For more information about Windows Hello for Business, see [MSDOCS-WHfB].

*Date format: YYYY/MM/D

[MS-WCFESAN]: WCF-Based Encrypted Server Administration and Notification Protocol

This topic lists the Errata found in [MS-WCFESAN] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

[MS-WDSMT]: Windows Deployment Services Multicast Transport Protocol

This topic lists the Errata found in [MS-WDSMT] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-WDSOSD]: Windows Deployment Services Operation System Deployment Protocol

This topic lists the Errata found in the MS-FAX document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

September 12, 2018 - Download

[MS-WFDAA]: Wi-Fi Direct (WFD) Application to Application Protocol

This topic lists the Errata found in [MS-WFDAA] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

June 1, 2017 - Download

[MS-WFDPE]: Wi-Fi Display Protocol Extension

This topic lists the Errata found in [MS-WFDPE] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

March 20, 2017 - Download

June 1, 2017 - Download

[MS-WKST]: Workstation Service Remote Protocol

This topic lists the Errata found in [MS-WKST] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

Errata below are for Protocol Document Version V28.0 – 2018/09/12.

Errata Published*	Description
2018/11/12	In Section 3.2.4.8, NetrUseGetInfo (Opnum 9), changed from: … The server MUST fill the return structures as follows: • If the Level member is 0, the server MUST return the information about the connection by filling the USE_INFO_0_CONTAINER (section 2.2.5.25) structure in the Buffer field of the InfoStruct parameter as follows. USE_INFO_0_CONTAINER contains an array of USE_INFO_0 structures. • ui0_local set to Connection.local • ui0_remote set to Connection.Remote • If the Level member is 1, the server MUST return the information about the connection by filling the USE_INFO_1_CONTAINER (section 2.2.5.26) structure in the Buffer field of the InfoStruct parameter as follows. USE_INFO_1_CONTAINER contains an array of USE_INFO_1 structures. • ui1_local set to Connection.local • ui1_remote set to Connection.remote • ui1_password set to NULL • ui1_status set to Connection.status • ui1_asg_type set to Connection.asgtype • ui1_refcount set to Connection.refcount • ui1_usecount set to Connection.useCount • If the Level member is 2, the server MUST return the information about the connection by filling the USE_INFO_2_CONTAINER (section 2.2.5.27) structure in the Buffer field of the InfoStruct parameter as follows. USE_INFO_2_CONTAINER contains an array of USE_INFO_2 structures. • ui2_local set to Connection.local • ui2_remote set to Connection.remote • ui2_password set to NULL • ui2_status set to Connection.status • ui2_asg_type set to Connection.asgtype • ui2_refcount set to Connection.refcount • ui2_usecount set to Connection.useCount • ui2_domainname set to Connection.domain • If the Level member is 3, the server MUST return the information about the connection by filling the USE_INFO_3_CONTAINER structure in the Buffer field of the InfoStruct parameter as follows. USE_INFO_3_CONTAINER contains an array of

Errata Published*	Description
	USE_INFO_3 structures. ● ui2_local set to Connection.local ● ui2_remote set to Connection.remote ● ui2_password set to NULL ● ui2_status set to Connection.status ● ui2_asg_type set to Connection.asgtype ● ui2_refcount set to Connection.refcount ● ui2_usecount set to Connection.useCount ● ui2_domainname set to Connection.domain ● ui2_flag set to 0 The server MUST invoke the event to end the client impersonation ([MS-RPCE] section 3.3.3.4.3.3). Changed to: … The server MUST fill the return structures as follows: ● If the Level member is 0, the server MUST return the information about the connection by filling the USE_INFO_0_CONTAINER (section 2.2.5.25) structure in the Buffer field of the InfoStruct parameter as follows. USE_INFO_0_CONTAINER contains an array of USE_INFO_0 structures. ● ui0_local set to Connection.local ● ui0_remote set to Connection.Remote ● If the Level member is 1, the server MUST return the information about the connection by filling the USE_INFO_1_CONTAINER (section 2.2.5.26) structure in the Buffer field of the InfoStruct parameter as follows. USE_INFO_1_CONTAINER contains an array of USE_INFO_1 structures. ● ui1_local set to Connection.local ● ui1_remote set to Connection.remote ● ui1_password set to NULL ● ui1_status set to Connection.status ● ui1_asg_type set to Connection.asgtype ● ui1_refcount set to Connection.refcount ● ui1_usecount set to Connection.usecount ● If the Level member is 2 or 3, the server MUST return the information about the connection by filling the USE_INFO_2_CONTAINER (section 2.2.5.27) structure in the Buffer field of the InfoStruct parameter as follows. USE_INFO_2_CONTAINER contains an array of USE_INFO_2 structures. ● ui2_local set to Connection.local ● ui2_remote set to Connection.remote ● ui2_password set to NULL ● ui2_status set to Connection.status ● ui2_asg_type set to Connection.asgtype ● ui2_refcount set to Connection.refcount ● ui2_usecount set to Connection.usecount ● ui2_username set to Connection.username ● ui2_domainname set to Connection.domain The server MUST invoke the event to end the client impersonation ([MS-RPCE] section 3.3.3.4.3.3).
2018/11/12	In Section 3.2.4.13, NetrJoinDomain2 (Opnum 22), changed from:

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th>Value/code</th><th>Meaning</th></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td>NETSETUP_MACHINE_PWD_PASSED
0x00000080</td><td>Indicates that the Password parameter SHOULD<58> specify the password for the machine joining the domain. This flag is valid only for unsecured joins, which MUST be indicated by setting the NETSETUP_JOIN_UNSECURE flag. If this flag is set, the value of Password determines the value stored for the computer password during the join process.</td></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td>NETSETUP_INSTALL_INVOCATION
0x00040000</td><td>Indicates that the protocol method was invoked during installation</td></tr>
</table>
Changed to:
<table>
<tr><th>Value/code</th><th>Meaning</th></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td>NETSETUP_MACHINE_PWD_PASSED
0x00000080</td><td>Indicates that the Password parameter SHOULD<58> specify the password for the machine joining the domain. This flag is valid only for unsecured joins, which MUST be indicated by setting the NETSETUP_JOIN_UNSECURE flag, or read-only joins, which MUST be indicated by setting the NETSETUP_JOIN_READONLY flag. If this flag is set, the value of Password determines the value stored for the computer password during the join process.</td></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td>NETSETUP_JOIN_READONLY
0x00000800</td><td>Specifies that the join SHOULD <121> be performed in a read-only manner against an existing account object. This option is intended to enable the server to join a domain using a read-only domain controller.</td></tr>
<tr><td>NETSETUP_INSTALL_INVOCATION
0x00040000</td><td>Indicates that the protocol method was invoked during installation</td></tr>
</table>
<121> Section 3.2.4.13: Windows NT, Windows 2000, Windows XP, Windows Server 2003, and Windows Server 2003 R2 do not implement this option.
In Section 3.2.4.13.3, Domain Join Specific Message Processing, changed from:
</td></tr>
</table>

Errata Published*	Description
	The following statements define the sequence of message-processing operations: 1. If the NETSETUP_MACHINE_PWD_PASSED bit is set in Options, and the NETSETUP_JOIN_UNSECURE bit is not set in Options, the server MUST return ERROR_INVALID_PARAMETER. Otherwise, message processing continues. 2. If the NETSETUP_MACHINE_PWD_PASSED bit is set in Options, and AccountName is not NULL, the server MUST return ERROR_INVALID_PARAMETER. Otherwise, message processing continues. 3. If the NETSETUP_MACHINE_PWD_PASSED bit is set in Options, and either Password is NULL or the length of the PasswordString is zero, the server MUST return ERROR_PASSWORD_RESTRICTION. Otherwise, message processing continues. 4. If the NETSETUP_MACHINE_PWD_PASSED bit is set in Options, the value of PasswordString MUST be copied to the value of ComputerPasswordString, and PasswordString MUST be set to NULL. 5. If the server processing the message is already joined to a domain, and the NETSETUP_DOMAIN_JOIN_IF_JOINED bit is not set in Options, the server MUST return NERR_SetupAlreadyJoined. Otherwise, message processing continues. … 6. If DomainNameString contains the character "\",… The specified domain controller MUST be validated by invoking the DsrGetDcNameEx2 method ([MS-NRPC] section 3.5.4.3.1) on the DomainControllerString computer, specifying the following parameters: … ● Flags = B | J | R … If the call fails, or the returned domain controller name does not match DomainControllerString, the server MUST invoke the DsrGetDcNameEx2 method ([MS-NRPC] section 3.5.4.3.1) on the DomainControllerString computer, specifying the following parameters: … ● Flags = B | J | S … 29. The following LDAP attributes… Changed to: The following statements define the sequence of message-processing operations: 1. If the NETSETUP_MACHINE_PWD_PASSED bit is set in Options, and the NETSETUP_JOIN_UNSECURE bit is not set in Options, the server MUST return ERROR_INVALID_PARAMETER. Otherwise, message processing continues. 2. If the NETSETUP_MACHINE_PWD_PASSED bit is set in Options, and AccountName is not NULL, the server MUST return ERROR_INVALID_PARAMETER. Otherwise, message processing continues. 3. If the NETSETUP_MACHINE_PWD_PASSED bit is set in Options, and either Password is NULL or the length of the PasswordString is zero, the server MUST return ERROR_PASSWORD_RESTRICTION. Otherwise, message processing continues. 4. If the NETSETUP_MACHINE_PWD_PASSED bit is set in Options, the value of PasswordString MUST be copied to the value of ComputerPasswordString, and PasswordString MUST be set to NULL. 5. If the NETSETUP_JOIN_READONLY bit is set in Options, and NETSETUP_MACHINE_PWD_PASSED bit is not set in Options, the server MUST return

Errata Published*	Description
	ERROR_INVALID_PARAMETER. Otherwise, message processing continues. 6. If the NETSETUP_JOIN_READONLY bit is set in Options, and the NETSETUP_ACCT_CREATE bit is set in Options, the server MUST return ERROR_INVALID_PARAMETER. Otherwise, message processing continues. 7. If the NETSETUP_JOIN_READONLY bit is set in Options, the server MUST perform all subsequent message processing as if NETSETUP_DEFER_SPN_SET and NETSETUP_JOIN_UNSECURE bits are set in Options. 8. If the server processing the message is already joined to a domain, and the NETSETUP_DOMAIN_JOIN_IF_JOINED bit is not set in Options, the server MUST return NERR_SetupAlreadyJoined. Otherwise, message processing continues.… 9. If DomainNameString contains the character "\",… The specified domain controller MUST be validated by invoking the DsrGetDcNameEx2 method ([MS-NRPC] section 3.5.4.3.1) on the DomainControllerString computer, specifying the following parameters: … ● Flags : if NETSETUP_JOIN_READONLY bit is set in Options, set Flags = (B | R); otherwise set Flags to (B | J | R) … If the call fails, or the returned domain controller name does not match DomainControllerString, the server MUST invoke the DsrGetDcNameEx2 method ([MS-NRPC] section 3.5.4.3.1) on the DomainControllerString computer, specifying the following parameters: … ● Flags : if NETSETUP_JOIN_READONLY bit is set in Options, set Flags = (B | S); otherwise set Flags to (B | J | S) … 32. If the NETSETUP_JOIN_READONLY bit is not set in Options, the following LDAP attributes…

*Date format: YYYY/MM/DD

[MS-WMIO]: Windows Management Instrumentation Encoding Version 1.0 Protocol

This topic lists the Errata found in [MS-WMIO] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

RSS
Atom

Errata are subject to the same terms as the Open Specifications documentation referenced.

Errata below are for Protocol Document Version V15.0 – 2018/09/12.

Errata Published*	Description
2019/06/10	In Section 3 Structure Examples, we revised the octet value of PropertyInfoRef. Changed from: A0 00 00 00 Changed to: 0A 00 00 00

*Date format: YYYY/MM/DD

[MS-WMF]: Windows Metafile Format

This topic lists the Errata found in [MS-WMF] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-WPO]: Windows Protocols Overview

This topic lists the Errata found in [MS-WPO] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

[MS-WSDS]: WS-Enumeration Directory Services Protocol Extensions

This topic lists the Errata found in [MS-WSDS] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

March 20, 2017 - Download

June 1, 2017 - Download

[MS-WSMV]: Web Services Management Protocol Extensions for Windows Vista

This topic lists the Errata found in [MS-WSMV] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

December 1, 2017 - Download

September 12, 2018 - Download

[MS-WSP]: Windows Search Protocol

This topic lists the Errata found in [MS-WSP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

September 23, 2019 - Download

March 4, 2020 - Download

Errata below are for Protocol Document Version V34.0 – 2020/03/04.

Errata Published*	Description
2020/08/17	In Section 2.2.3.11 CPMGetRowsIn, the valid values of the eType field have been updated: Value Meaning 0x00000000 There is no SeekDescription; the SeekDescription field is omitted. eRowSeekNext 0x00000001 SeekDescription contains a CRowSeekNext structure. eRowSeekAt 0x00000002 SeekDescription contains a CRowSeekAt structure. eRowSeekAtRatio 0x00000003 SeekDescription contains a CRowSeekAtRatio structure. eRowSeekByBookmark 0x00000004 SeekDescription contains a CRowSeekByBookmark structure. To Value Meaning eRowSeekNext

Errata Published*	Description
	0x00000001 SeekDescription contains a CRowSeekNext structure. eRowSeekAt 0x00000002 SeekDescription contains a CRowSeekAt structure. eRowSeekAtRatio 0x00000003 SeekDescription contains a CRowSeekAtRatio structure. eRowSeekByBookmark 0x00000004 SeekDescription contains a CRowSeekByBookmark structure. In Section 3.1.5.2.6 Receiving a CPMGetRowsIn Request, the following Changed from: ● If eType == 0x00000000, then no work needs to be done to reposition the next row's read index within the rowset. Changed to: ● If eType == 0x00000000, then set status = STATUS_INVALID_PARAMETER(0xC000000D).
2020/08/17	In section 2.2.3.5 CPMCreateQueryOut the size of the aCursors field was changed from 4 bytes to variable. Changed from: aCursors (4 bytes): An array of 32-bit unsigned integers representing the handles to cursors with the number of elements equal to the number of categories in the CategorizationSet field of CPMCreateQueryIn message plus one element, representing an uncategorized cursor. Changed to: aCursors (variable): An array of 32-bit unsigned integers representing the handles to cursors with the number of elements equal to the number of categories in the CategorizationSet field of CPMCreateQueryIn message plus one element, representing an uncategorized cursor.
2020/06/08	In Section 2.2.4, Errors, added 4 error codes. Changed from: DB_S_ENDOFROWSET (0x00040EC6) ... Changed to:

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>DB_S_ENDOFROWSET (0x00040EC6)
CI_E_SHUTDOWN (0x80041812)
CI_E_NOT_INITIALIZED (0x8004180B)
DB_E_BADBINDINFO (0x80040E08)
MSS_E_CATALOGNOTFOUND (0x80042103)

In Section 3.1.5.2.1, Receiving a CPMConnectIn Request, changed from:

8. Report any errors encountered during message preparation or during any abstract interface call to the GSS. Errors that are specific to this request:
...
STATUS_INVALID_PARAMETER_MIX: Generated when the client version as passed in this message is smaller than CI_VERSION_WDS30 (0x102).

Changed to:

8. Report any errors encountered during message preparation or during any abstract interface call to the GSS. Errors that are specific to this request:
...
STATUS_INVALID_PARAMETER_MIX: Generated when the client version as passed in this message is smaller than 0x102.

In Section 3.1.5.2.2, Receiving a CPMCreateQueryIn Request, changed from:

4. Report any errors encountered during message preparation or during any abstract interface call to the GSS. The following errors are specific to this request:
...
STATUS_INVALID_PARAMETER_MIX: generated when the client version, as passed in this message, is smaller than CI_VERSION_WDS30 (0x102).

Changed to:
4. Report any errors encountered during message preparation or during any abstract interface call to the GSS. The following errors are specific to this request:
...
STATUS_INVALID_PARAMETER_MIX: generated when the client version, as passed in this message, is smaller than 0x102.</td></tr>
<tr><td>2020/06/08</td><td>In Section 2.2.1.1, CBaseStorageVariant, vType value tablewas changed to add VT_CF.
Changed from:

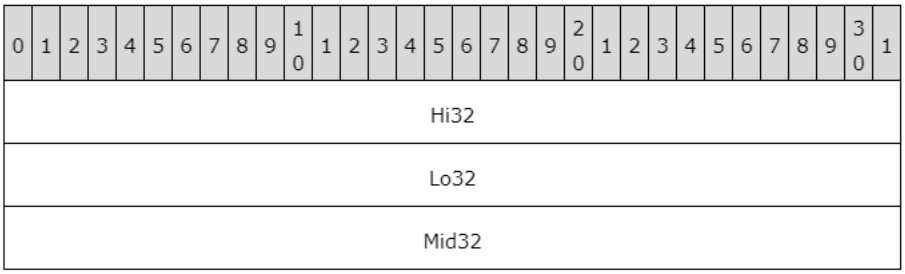
<table><tr><td>VT_BSTR
0x0008</td><td>vValue</td></tr><tr><td>VT_LPSTR
0x001E</td><td>A null-terminated string using the system code page.</td></tr></table></td></tr>
</table>

Errata Published*	Description

Changed to:

VT_BSTR 0x0008	vValue
VT_CF 0x0011	A VT_CF structure as specified in section 2.2.1.1.1.7.
VT_LPSTR 0x001E	A null-terminated string using the system code page.

In Section 2.2.1.1.1.1, DECIMAL, bit table with definitions were revised.
Changed from:

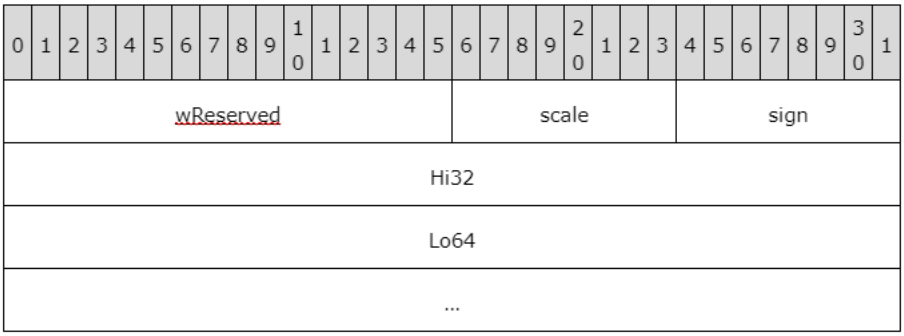


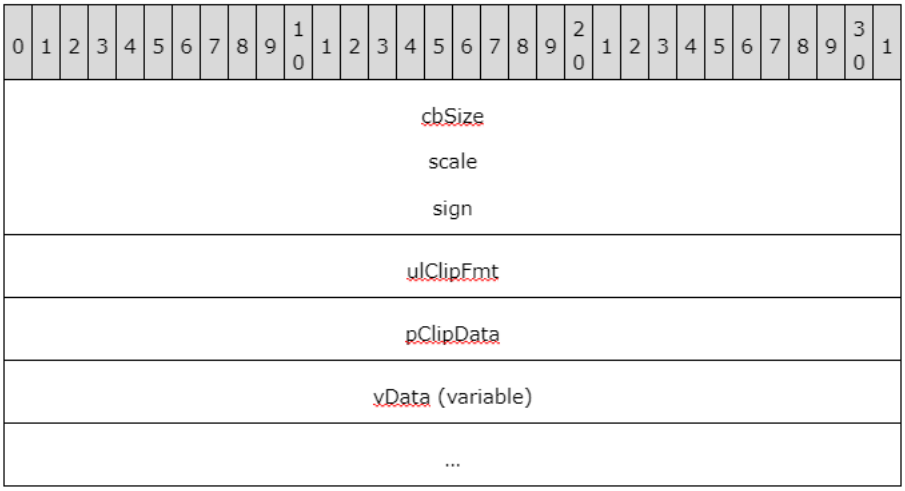
Hi32 (4 bytes): The highest 32 bits of the 96-bit integer.

Lo32 (4 bytes): The lowest 32 bits of the 96-bit integer.

Mid32 (4 bytes): The middle 32 bits of the 96-bit integer.

Changed to:



Errata Published*	Description
	wReserved (2 bytes): MUST be set to zero and MUST be ignored. scale (1 byte): The number of decimal places for the number. Valid values are from 0 to 28. sign (1 byte): Indicates the sign; 0 for positive numbers or DECIMAL_NEG(0x80) for negative numbers. Hi32 (4 bytes): The high 32 bits of the number. Lo64 (8 bytes): The low 64 bits of the number. Added new Section 2.2.1.1.1.7, VT_CF: **2.2.1.1.1.7 VT_CF** VT_CF is used for clipboard format.  cbSize (4 bytes): A 32-bit unsigned integer that specif ulClipFmt (4 bytes): A 32-bit unsigned integer that specifies the clipboard format. pClipData (4 bytes): A 32-bit unsigned integer that specifies the offset to clipboard format data. vData (variable): The clipboard format data. In Section 2.2.1.13, CRelDocRestriction, added description of 'docid'. Changed from: In the example, "Windows" is the application name,"SystemIndex" is the catalog name, and "153"

Errata Published*	Description
	is thedocument ID for the document in decimal notation.<3> Changed to: In the example, "docid" is a uniquie 32-bit unsigned integer, "Windows" is the application name, "SystemIndex" is the catalog name, and "153" is the document ID for the document in decimal notation.<3> In Section 3.1.5.2.4, Receiving a CPMGetQueryStatusExInRequest, revised CFilteredDocuments and CTotalDocuments. Changed from: Use Output parameters: cFilteredDocuments= CFilteredDocuments SetcDocumentsToFilter to: CTotalDocuments - CFilteredDocuments Changed to: Use Output parameters: cFilteredDocuments = cFilteredDocuments Set cDocumentsToFilter to: cTotalDocuments -cFilteredDocuments In Section 3.1.5.2.6, Receiving a CPMGetRowsIn Request, revised _chapter to _chapt. Changed from: Call the GetBookmarkPosition abstract interface to the GSS with QueryIdentifier, CursorHandle, and the bookmark handle value as arguments. Use the bmkIndex (section 3.1.7) returned as an argument to SetNextGetRowsPosition (section 3.1.7), along with QueryIdentifier, CursorHandle, and _chapter. Call the GetRows abstract interface with QueryIdentifier, CursorHandle, 1, and _fBwdFetch as arguments. 5.After the position is set, retrieve the desired row from the GSS by calling the GetRows abstract interface with the HANDLE of the named pipe over which the server has received the CPMGetRowsIn message as its QueryIdentifier argument, with the _hCursor handle as its CursorHandle argument, with _chapter as its chapter argument, with _cRowsToTransfer as its NumRowsRequested argument, and with _fBwdFetch as its FetchForward argument. Do this in all cases, except for step 4 bullet 3. 6. Copy as many rows as fit in a buffer, the size of which is indicated by _cbReadBuffer (section 2.2.3.11), but not more than indicated by _cRowsToTransfer (section 2.2.3.11). Thereafter, reposition the cursor to reflect the actual number of returned rows by calling the SetNextGetRowsPosition abstract interface with QueryIdentifier, CursorHandle and _chapter as arguments and with Index set to the old index (as obtained by calling GetNextRowsPosition(QueryIdentifier, _hCursor, _chapter)) plus the number of rows that fit in the buffer.

Errata Published*	Description
	Changed to: Call the GetBookmarkPosition abstract interface to the GSS with QueryIdentifier, CursorHandle, and the bookmark handle value as arguments. Use the bmkIndex (section 3.1.7) returned as an argument to SetNextGetRowsPosition (section 3.1.7), along with QueryIdentifier, CursorHandle, and _chapt. Call the GetRows abstract interface with QueryIdentifier, CursorHandle, 1, and _fBwdFetch as arguments. 5. After the position is set, retrieve the desired row from the GSS by calling the GetRows abstract interface with the HANDLE of the named pipe over which the server has received the CPMGetRowsIn message as its QueryIdentifier argument, with the _hCursor handle as its CursorHandle argument, with _chapt as its chapter argument, with _cRowsToTransfer as its NumRowsRequested argument, and with _fBwdFetch as its FetchForward argument. Do this in all cases, except for step 4 bullet 3. 6. Copy as many rows as fit in a buffer, the size of which is indicated by _cbReadBuffer (section 2.2.3.11), but not more than indicated by _cRowsToTransfer (section 2.2.3.11). Thereafter, reposition the cursor to reflect the actual number of returned rows by calling the SetNextGetRowsPosition abstract interface with QueryIdentifier, CursorHandle and _chapt as arguments and with Index set to the old index (as obtained by calling GetNextRowsPosition(QueryIdentifier, _hCursor, _chapt)) plus the number of rows that fit in the buffer. In Section 3.1.7, Other Local Events, added definition forWorkid. Changed from: FetchForward, a 32-bit unsigned integer identifying whether the rows are to be fetched in forward order or in reverse. (0x00000000 for forward, 0x00000001 for reverse) Changed to: FetchForward, a 32-bit unsigned integer identifying whether the rows are to be fetched in forward order or in reverse. (0x00000000 for forward, 0x00000001 for reverse) Workid, a 32-bit unsigned integer representing the document ID identifying the document for which a property is to be fetched.
2020/05/25	In Sections 2 and 3, field and structure names were corrected (e.g., DBRANGEBOUND_EXACT changed to DBRANGEBOUNDTTYPE_EXACT and VT_Vector changed to VT_VECTOR). See the PDF doc here for details.

*Date format: YYYY/MM/DD

[MS-WSTEP]: WS-Trust X.509v3 Token Enrollment Extensions

This topic lists the Errata found in [MS-WSTEP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-WSUSAR]: Windows Server Update Services: Administrative API Remoting Protocol

This topic lists the Errata found in the MS-WSUSAR document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

June 1, 2017 - Download

March 4, 2020 - Download

Errata below are for Protocol Document Version V7.0 – 2020/03/04.

Errata Published*	Description
2020/07/06	Throughout this document, changed all occurrences of "computerId", "eulaFailed", and "updateRevisionId" and variations of those terms to "ComputerId", "EULAFailed", and "UpdateRevisionId".

*Date format: YYYY/MM/DD

[MS-WSUSOD]: Windows Server Update Services Protocols Overview

This topic lists the Errata found in [MS-WSUSOD] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

June 1, 2017 - Download

[MS-WSUSSS]: Windows Update Services: Server-Server Protocol

This topic lists the Errata found in the MS-WSUSSS document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

July 18, 2016 - Download

June 1, 2017 - Download

[MS-WUSP]: Windows Update Services: Client-Server Protocol

This topic lists the Errata found in [MS-WMF] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download

July 18, 2016 - Download

June 1, 2017 - Download

March 4, 2020 - Download

[MS-XCA]: Xpress Compression Algorithm

This topic lists the Errata found in [MS-XCA] since it was last published.

Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

March 4, 2020 - Download

Errata below are for Protocol Document Version V6.0 – 2020/03/04.

Errata Published*	Description

2020/08/17	In Section 2.2.4 Processing, we corrected the pseudocode to remove extraneous implementation-specific processing.
	
	Changed from:
	
	`Loop until a decompression terminating condition`
	`Check for EOF`
	`Build the decoding table`
	`CurrentPosition += 256 // start at the end of the Huffman table`
	`NextBits = Read16Bits(InputBuffer + CurrentPosition)`
	`CurrentPosition += 2`
	`NextBits <<= 16`
	`NextBits |= Read16Bits(InputBuffer + CurrentPosition)`
	`CurrentPosition += 2`
	`ExtraBits = 16`
	`BlockEnd = OutputPosition + 65536`
	
	`Loop until a block terminating condition`
	`Loop until a literal processing terminating condition`
	`If OutputPosition >= BlockEnd then terminate block processing`
	`Next15Bits = NextBits >> (32 - 15)`
	`HuffmanSymbol = DecodingTable[Next15Bits]`
	`HuffmanSymbolBitLength = the bit length of HuffmanSymbol, from the table in the input buffer`
	`If HuffmanSymbol <= 0`
	`NextBits <<= HuffmanSymbolBitLength`
	`ExtraBits -= HuffmanSymbolBitLength`
	
	`Do`
	`HuffmanSymbol = - HuffmanSymbol`
	`HuffmanSymbol += (NextBits >> 31)`
	`NextBits *= 2`
	`ExtraBits = ExtraBits - 1`
	`HuffmanSymbol = DecodingTable[HuffmanSymbol]`
	`While HuffmanSymbol <= 0`
	`Else`
	`DecodedBitCount = HuffmanSymbol & 15`
	`NextBits <<= DecodedBitCount`
	`ExtraBits -= DedcodedBitCount`
	`HuffmanSymbol >>= 4 // Shift by 4 bits to get the symbol value // (the lower 4 bits are the bit length of the symbol)`

Errata Published*	Description

```
                        HuffmanSymbol -= 256
                        If ExtraBits < 0
                            NextBits |= Read16Bits(InputBuffer + CurrentPosition)
    << (-ExtraBits)
                            ExtraBits += 16
                            CurrentPosition += 2
                        If HuffmanSymbol >= 0
                            If HuffmanSymbol == 0
                                If the entire input buffer has been read and
                                the expected decompressed size has been written to
    the output buffer
                                    Decompression is complete.  Return with
    success.
                            Terminate literal processing
                        Else
                            Output the byte value of HuffmanSymbol to the output
    stream
                End of literal processing Loop

                MatchLength = HuffmanSymbol mod 16
                MatchOffsetBitLength = HuffmanSymbol / 16
                If MatchLength == 15
                    MatchLength = ReadByte(InputBuffer + CurrentPosition)
                    CurrentPosition += 1
                    If MatchLength == 255
                        MatchLength = Read16Bits(InputBuffer + CurrentPosition)
                        CurrentPosition += 2
                        If MatchLength < 15
                            The compressed data is invalid. Return error.
                        MatchLength = MatchLength - 15
                    MatchLength = MatchLength + 15
                MatchLength = MatchLength + 3
                MatchOffset = NextBits >> (32 - MatchOffsetBitLength)
                MatchOffset += (1 << MatchOffsetBitLength)
                NextBits <<= MatchOffsetBitLength
                ExtraBits -= MatchOffsetBitLength
                If ExtraBits < 0
                    NextBits |= Read16Bits(InputBuffer + CurrentPosition) << (-
    ExtraBits)
                    ExtraBits += 16
                    CurrentPosition += 2
                For i = 0 to MatchLength - 1
                    Output OutputBuffer[OutputPosition - MatchOffset + i]
             End of block loop
        End of decoding loop
```

Changed to:

```
        Loop until a decompression terminating condition
            Build the decoding table
            CurrentPosition = 256              // start at the end of the Huffman
    table
            NextBits = Read16Bits(InputBuffer + CurrentPosition)
            CurrentPosition += 2
            NextBits <<= 16
            NextBits |= Read16Bits(InputBuffer + CurrentPosition)
            CurrentPosition += 2
            ExtraBitCount = 16
            BlockEnd = OutputPosition + 65536
            Loop until a block terminating condition
                If the OutputPosition >= BlockEnd then terminate block processing
                Next15Bits = NextBits >> (32 - 15)
                HuffmanSymbol = DecodingTable[Next15Bits]
```

Errata Published*	Description
	HuffmanSymbolBitLength = the bit length of HuffmanSymbol, from the table in the input buffer NextBits <<= HuffmanSymbolBitLength ExtraBitCount -= HuffmanSymbolBitLength If ExtraBitCount < 0 NextBits |= Read16Bits(InputBuffer + CurrentPosition) << (-ExtraBitCount) ExtraBitCount += 16 CurrentPosition += 2 If HuffmanSymbol < 256 Output the byte value HuffmanSymbol to the output stream. Else If HuffmanSymbol == 256 and the entire input buffer has been read and the expected decompressed size has been written to the output buffer Decompression is complete. Return with success. Else HuffmanSymbol = HuffmanSymbol - 256 MatchLength = HuffmanSymbol mod 16 MatchOffsetBitLength = HuffmanSymbol / 16 If MatchLength == 15 MatchLength = ReadByte(InputBuffer + CurrentPosition) CurrentPosition += 1 If MatchLength == 255 MatchLength = Read16Bits(InputBuffer + CurrentPosition) CurrentPosition += 2 If MatchLength < 15 The compressed data is invalid. Return error. MatchLength = MatchLength - 15 MatchLength = MatchLength + 15 MatchLength = MatchLength + 3 MatchOffset = NextBits >> (32 – MatchOffsetBitLength) MatchOffset += (1 << MatchOffsetBitLength) NextBits <<= MatchOffsetBitLength ExtraBitCount -= MatchOffsetBitLength If ExtraBitCount < 0 Read the next 2 bytes the same as the preceding (ExtraBitCount < 0) case For i = 0 to MatchLength - 1 Output OutputBuffer[CurrentOutputPosition – MatchOffset + i] End of block loop End of decoding loop
2020/06/08	In Section 2.2.4 Processing, we clarified when and how implementations must check for the EOF condition during decompression. We modified the pseudocode and added explanatory text. Changed from: The compression stream is designed to be read in (mostly) 16-bit chunks, with a 32-bit register maintaining at least the next 16 bits of input. This strategy allows the code to seamlessly handle the bytes for long match lengths, which would otherwise be awkward. The following pseudocode demonstrates this method. Loop until a decompression terminating condition Build the decoding table ...

Errata Published*	Description
	Changed to: The compression stream is designed to be read in (mostly) 16-bit chunks, with a 32-bit register maintaining at least the next 16 bits of input. This strategy allows the code to seamlessly handle the bytes for long match lengths, which would otherwise be awkward. The following pseudocode demonstrates this method. During the beginning of processing each block for decompression, an implementation MUST check for EOF. An implementation can do this by comparing the block size against the required space for a Huffman table ” if this condition is met and all output has been written, then processing stops and success is returned. Alternately, an implementation can explicitly examine the input buffer using the Huffman table from the previous block. Loop until a decompression terminating condition Check for EOF Build the decoding table ...
2020/04/27	In Section 2.2.4, Processing, we replaced CurrentOutputPosition with OutputPosition for simplicity and clarity of the pseudocode. Changed from: For i = 0 to MatchLength - 1 Output OutputBuffer[CurrentOutputPosition – MatchOffset + i] Changed to: For i = 0 to MatchLength - 1 Output OutputBuffer[OutputPosition – MatchOffset + i]
2020/04/27	In Section 2.2.4, Processing, we clarified the nesting and termination conditions of the loops in the pseudocode. Changed from: Loop until a block terminating condition If OutputPosition >= BlockEnd then terminate block processing Loop until a literal processing terminating condition Changed to: Loop until a block terminating condition Loop until a literal processing terminating condition

Errata Published*	Description
	If OutputPosition >= BlockEnd then terminate block processing
2020/04/27	In Section 2.2.4, Processing, we altered the pseudocode to advance the CurrentPosition by 256 rather than assigning a fixed value of 256. Changed from: CurrentPosition = 256 // start at the end of the Huffman table Changed to: CurrentPosition += 256 // start at the end of the Huffman table

*Date format: YYYY/MM/DD

[MS-XCEP]: X.509 Certificate Enrollment Policy Protocol

This topic lists the Errata found in [MS-XCEP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS
Atom

No errata are available for the latest version of this Windows Protocols document. To view a PDF file of the errata for the previous versions of this document, see the following ERRATA Archives:

October 16, 2015 - Download

June 30, 2015 - Download