

[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.

August 24, 2020 – [Download](#)

Errata below are for Protocol Document [Version V61.0 - 2020/08/26](#)

Errata Published*	Description
2021/03/22	<p>The following sections were updated for processing rules on unauthenticated session.</p> <p>Section 3.3.1.5 Global Section 3.3.1.7 Per Transport Connection Section 3.3.3 Initialization Section 3.3.5.1 Accepting an Incoming Connection Section 3.3.5.2.9 Verifying the Session Section 3.3.5.5.3 Handling GSS-API Authentication</p> <p>For details, please see the diff file.</p>
2021/03/22	<p>In Section 3.2.4.1.1 Signing the Message, the following was updated.</p> <p>Changed from:</p> <p>The client MUST sign the message under the following conditions:</p> <ul style="list-style-type: none">• If the request message being sent contains a nonzero value in the SessionId field, the session identified by the SessionId has Session.SigningRequired equal to TRUE and either the request is a TREE_CONNECT request or the tree connection identified by the TreeId field has TreeConnect.EncryptData equal to FALSE.• If Connection.Dialect is "3.1.1" and the message being sent is a TREE_CONNECT Request and the session identified by SessionId has Session.EncryptData equal to FALSE. <p>If Session.SigningRequired is FALSE, the client MAY<92> sign the request.</p> <p>Changed to:</p> <p>The client MUST sign the message if one of the following conditions is TRUE:</p> <ul style="list-style-type: none">• If Connection.Dialect is equal to "2.0.2" or "2.1", the message being sent contains a nonzero value in the SessionId field, and the session identified by the SessionId has Session.SigningRequired equal to TRUE.• If Connection.Dialect belongs to the SMB 3.x dialect family, the message being sent contains a nonzero value in the SessionId field and one of the following conditions is TRUE:

Errata Published*	Description										
	<ul style="list-style-type: none"> The session identified by SessionId has Session.EncryptData equal to FALSE. The tree connection identified by the TreeId field has TreeConnect.EncryptData equal to FALSE. <p>If Session.SigningRequired is FALSE, the client MAY<92> sign the request.</p>										
2021/03/08	<p>In section 6, Appendix A: Product Behavior, behavior note 349 was updated for allowed FSCTLs:</p> <p>Changed from:</p> <p>Windows 10 v20H2 and later and Windows Server v20H2 and later allow the additional CtlCode value, as specified in [MS-FSCC].</p> <table border="1" data-bbox="402 703 1281 856"> <thead> <tr> <th>FSCTL name</th> <th>FSCTL function number</th> </tr> </thead> <tbody> <tr> <td>FSCTL_GET_RETRIEVAL_POINTERS_AND_REFCOUNT</td> <td>0x903D3</td> </tr> <tr> <td>FSCTL_GET_RETRIEVAL_POINTER_COUNT</td> <td>0x9042B</td> </tr> </tbody> </table> <p>Changed to:</p> <p>Windows 10 v20H2 and later and Windows Server v20H2 and later allow the additional CtlCode value, as specified in [MS-FSCC].</p> <table border="1" data-bbox="402 1060 1166 1199"> <thead> <tr> <th>FSCTL name</th> <th>FSCTL function number</th> </tr> </thead> <tbody> <tr> <td>FSCTL_GET_RETRIEVAL_POINTER_COUNT</td> <td>0x9042B</td> </tr> </tbody> </table>	FSCTL name	FSCTL function number	FSCTL_GET_RETRIEVAL_POINTERS_AND_REFCOUNT	0x903D3	FSCTL_GET_RETRIEVAL_POINTER_COUNT	0x9042B	FSCTL name	FSCTL function number	FSCTL_GET_RETRIEVAL_POINTER_COUNT	0x9042B
FSCTL name	FSCTL function number										
FSCTL_GET_RETRIEVAL_POINTERS_AND_REFCOUNT	0x903D3										
FSCTL_GET_RETRIEVAL_POINTER_COUNT	0x9042B										
FSCTL name	FSCTL function number										
FSCTL_GET_RETRIEVAL_POINTER_COUNT	0x9042B										
2021/02/08	<p>In Section 2.2.3, SMB2 NEGOTIATE Request, the description of SMB2_NEGOTIATE_SIGNING_ENABLED was updated:</p> <p>Changed from:</p> <p>When set, indicates that security signatures are enabled on the client. The client MUST set this bit if the SMB2_NEGOTIATE_SIGNING_REQUIRED bit is not set, and MUST NOT set this bit if the SMB2_NEGOTIATE_SIGNING_REQUIRED bit is set. The server MUST ignore this bit.</p> <p>Changed to:</p> <p>When set, indicates that security signatures are enabled on the client. The server MUST ignore this bit.</p> <p>In Section 2.2.5, SMB2 SESSION_SETUP Request, the description of SMB2_NEGOTIATE_SIGNING_ENABLED was updated:</p> <p>Changed from:</p> <p>When set, indicates that security signatures are enabled on the client. The client MUST set this bit if the SMB2_NEGOTIATE_SIGNING_REQUIRED bit is not set, and MUST NOT set this bit if the SMB2_NEGOTIATE_SIGNING_REQUIRED bit is set. The server MUST ignore this bit.</p>										

Errata Published*	Description
	<p>Changed to:</p> <p>When set, indicates that security signatures are enabled on the client. The server MUST ignore this bit.</p> <p>In Section 2.2.10, SMB2 TREE_CONNECT Response, the description of SMB2_SHAREFLAG_DFS, SMB2_SHAREFLAG_DFS_ROOT and SMB2_SHARE_CAP_DFS were updated:</p> <p>Changed from:</p> <p>SMB2_SHAREFLAG_DFS: The specified share is present in a Distributed File System (DFS) tree structure. The server SHOULD set the SMB2_SHAREFLAG_DFS bit in the ShareFlags field if the per-share property Share.IsDfs is TRUE.</p> <p>SMB2_SHAREFLAG_DFS_ROOT: The specified share is present in a DFS tree structure. The server SHOULD set the</p> <p>SMB2_SHAREFLAG_DFS_ROOT bit in the ShareFlags field if the per-share property Share.IsDfs is TRUE.</p> <p>SMB2_SHARE_CAP_DFS: The specified share is present in a DFS tree structure. The server MUST set the SMB2_SHARE_CAP_DFS bit in the Capabilities field if the per-share property Share.IsDfs is TRUE.</p> <p>Changed to:</p> <p>SMB2_SHAREFLAG_DFS: The specified share is present in a Distributed File System (DFS) tree structure.</p> <p>SMB2_SHAREFLAG_DFS_ROOT: The specified share is present in a DFS Root (as specified in [MS-DFSC]) tree structure</p> <p>SMB2_SHARE_CAP_DFS: The specified share is present in a DFS tree structure.</p>
2021/01/11	<p>In Section 3.3.5.9.7 Handling the SMB2_CREATE_DURABLE_HANDLE_RECONNECT Create Context the following bullet points were added under step 6</p> <ul style="list-style-type: none"> ● Open.IsDurable is TRUE, Open.Lease is NULL, and Open.OplockLevel is not equal to SMB2_OPLOCK_LEVEL_BATCH. ● Open.IsDurable is TRUE and Open.Lease.LeaseState does not contain SMB2_LEASE_HANDLE_CACHING. <p>In Section 3.3.5.9.12 Handling the SMB2_CREATE_DURABLE_HANDLE_RECONNECT_V2 Create Context the following was updated:</p> <p>Changed from:</p> <p>There is no processing done for "Path Name Validation" or "Open Execution" as listed in section 3.3.5.9.</p> <p>Changed to:</p>

Errata Published*	Description
	<p>There is no processing done for "Path Name Validation" as listed in section 3.3.5.9.</p> <p>The following bullet points were added:</p> <p>...</p> <ul style="list-style-type: none"> • Open.Lease is NULL and the SMB2_CREATE_REQUEST_LEASE or SMB2_CREATE_REQUEST_LEASE_V2 create context is present. • Open.Lease is NOT NULL and the SMB2_CREATE_REQUEST_LEASE or SMB2_CREATE_REQUEST_LEASE_V2 create context is not present. • The SMB2_CREATE_REQUEST_LEASE_V2 create context is also present in the request, the server supports directory leasing, and Open.Lease.LeaseKey does not match the LeaseKey provided in the SMB2_CREATE_REQUEST_LEASE_V2 create context. <p>...</p> <p>To</p> <p>...</p> <ul style="list-style-type: none"> • Open.Lease is NULL and the SMB2_CREATE_REQUEST_LEASE or SMB2_CREATE_REQUEST_LEASE_V2 create context is present. • Open.IsDurable is TRUE, Open.Lease is NULL, and Open.OplockLevel is not equal to SMB2_OPLOCK_LEVEL_BATCH. • Open.Lease is NOT NULL and the SMB2_CREATE_REQUEST_LEASE or SMB2_CREATE_REQUEST_LEASE_V2 create context is not present. • Open.IsDurable is TRUE and Open.Lease.LeaseState does not contain SMB2_LEASE_HANDLE_CACHING. • The SMB2_CREATE_REQUEST_LEASE_V2 create context is also present in the request, the server supports directory leasing, and Open.Lease.LeaseKey does not match the LeaseKey provided in the SMB2_CREATE_REQUEST_LEASE_V2 create context. <p>...</p> <p>The following bullet point was changed from:</p> <ul style="list-style-type: none"> • The server MUST ignore the DesiredAccess, ShareAccess, and CreateOptions fields in the request. <p>To</p> <ul style="list-style-type: none"> • If the SMB2_DHANDLE_FLAG_PERSISTENT bit in the Flags field of the SMB2_CREATE_DURABLE_HANDLE_RECONNECT_V2 create context is not set, the server MUST ignore the DesiredAccess, ShareAccess, and CreateOptions fields in the request.
2020/11/23	<p>In Section 2.2.26 SMB2 LOCK Request, the description of LockSequenceNumber and LockSequenceIndex have been updated.</p> <p>Changed from:</p> <p>LSN – LockSequenceNumber (4 bits): In the SMB 2.0.2 dialect, this field is unused and MUST be 0. The client MUST set this to 0, and the server MUST ignore it on receipt. In all other dialects, a 4-bit integer value.</p> <p>LockSequenceIndex (28 bits): In the SMB 2.0.2 dialect, this field is unused and MUST be 0. The client MUST set this to 0, and the server MUST ignore it on receipt. In all other dialects, a 28-bit integer value that MUST contain a value from 0 to 64, where 0 is reserved.</p>

Errata Published*	Description
	<p>Changed to:</p> <p>LockSequenceNumber,LockSequenceIndex (4 bytes): A 32-bit unsigned integer. In the SMB 2.0.2 dialect, this field is unused and MUST be reserved. The client MUST set this to 0, and the server MUST ignore it on receipt. In all other dialects, this field is interpreted as LockSequenceNumber and LockSequenceIndex fields.</p> <p>LockSequenceNumber (4 bits): The 4 least significant bits of this field containing integer value.</p> <p>LockSequenceIndex (28 bits): A 28-bit integer value that MUST contain a value from 0 to 64, where 0 is reserved.</p>
2020/11/23	<p>The following sections were updated for chained and unchained compression:</p> <p>Section 2.2.42 SMB2_COMPRESSION_TRANSFORM_HEADER</p> <p>Section 2.2.42.1 SMB2_COMPRESSION_PAYLOAD_HEADER</p> <p>Section 3.2.5.1.1.2 Decompressing the Message</p> <p>Section 3.3.5.2.1.2 Decompressing the Message</p> <p>In Section 3.1.4.4 Compressing the Message, all instances of "SMB2_COMPRESSION_TRANSFORM_HEADER" have been changed to "SMB2_COMPRESSION_TRANSFORM_HEADER_UNCHAINED" and all instances of "SMB2_COMPRESSION_PAYLOAD_HEADER" have been changed to "SMB2_COMPRESSION_CHAINED_PAYLOAD_HEADER"</p> <p>In Section 3.1.5.3 Decompressing the Chained Message, all instances of "SMB2_COMPRESSION_PAYLOAD_HEADER" have been changed to "SMB2_COMPRESSION_CHAINED_PAYLOAD_HEADER" and all instances of "SMB2_COMPRESSION_TRANSFORM_HEADER" have been changed to "SMB2_COMPRESSION_TRANSFORM_HEADER_CHAINED".</p> <p>For details, please see the diff file.</p>
2020/11/23	<p>In Section 2.2.2.2.2 Share Redirect Error Context Response, field descriptions have been updated.</p> <p>Changed from:</p> <p>StructureSize (4 bytes): This field MUST be set to the size of the structure.</p> <p>NotificationType (4 bytes): This field MUST be set to 3.</p> <p>ResourceNameOffset (4 bytes): The offset from the start of this structure to the ResourceName field.</p> <p>ResourceNameLength (4 bytes): The length of the share name provided in the ResourceName field, in bytes.</p> <p>Flags (2 bytes): This field MUST be set to zero.</p> <p>TargetType (2 bytes): This field MUST be set to zero.</p> <p>IPAddrCount (4 bytes): The number of MOVE_DST_IPADDR structures in the IPAddrMoveList field.</p> <p>IPAddrMoveList (variable): Array of MOVE_DST_IPADDR structures, as specified in section 2.2.2.2.1.</p> <p>ResourceName (variable): Name of the share as a counted Unicode string, as specified in [MS-DTYP] section 2.3.10.</p> <p>Changed to:</p>

Errata Published*	Description																				
	<p>StructureSize (4 bytes): This field MUST be set 48, indicating the size of this structure with a single MOVE_DST_IPADDR structure. This value is set regardless of the number of MOVE_DST_IPADDR structures returned and the length of ResourceName field.</p> <p>NotificationType (4 bytes): This field indicates the notification type and MUST be set to 3 (SHARE_MOVE_NOTIFICATION) defined in [MS-SWN] section 2.2.2.4.</p> <p>ResourceNameOffset (4 bytes): The offset from the start of this structure to the ResourceName field.</p> <p>ResourceNameLength (4 bytes): The length of the share name provided in the ResourceName field, in bytes.</p> <p>Reserved (2 bytes): This field MUST NOT be used and MUST be reserved. This field MUST be set to zero and MUST be ignored on receipt.</p> <p>TargetType (2 bytes): This field indicates the target is an IP address MUST be set to zero.</p> <p>IPAddrCount (4 bytes): The number of MOVE_DST_IPADDR structures in the IPAddrMoveList field.</p> <p>IPAddrMoveList (variable): Array of MOVE_DST_IPADDR structures, as specified in section 2.2.2.2.1.</p> <p>ResourceName (variable): A Unicode string containing the share name.</p> <p>In Section 2.2.2.2.1 MOVE_DST_IPADDR structure, the description of the values in the Type field have been updated.</p> <p>Changed from:</p> <table border="0"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>MOVE_DST_IPADDR_V4</td> <td></td> </tr> <tr> <td>0x00000001</td> <td>The type of destination IP address in this structure is IPv4 address. The fields after Reserved field in this structure are interpreted as IPv4Address followed by Reserved2 as described below.</td> </tr> <tr> <td>MOVE_DST_IPADDR_V6</td> <td></td> </tr> <tr> <td>0x00000002</td> <td>The type of destination IP address in this structure is IPv6 address. The field after Reserved field in this structure is interpreted as IPv6Address as described below.</td> </tr> </tbody> </table> <p>Changed to:</p> <table border="0"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>MOVE_DST_IPADDR_V4</td> <td></td> </tr> <tr> <td>0x00000001</td> <td>The type of destination IP address in this structure is IPv4 address.</td> </tr> <tr> <td>MOVE_DST_IPADDR_V6</td> <td></td> </tr> <tr> <td>0x00000002</td> <td>The type of destination IP address in this structure is IPv6 address.</td> </tr> </tbody> </table>	Value	Meaning	MOVE_DST_IPADDR_V4		0x00000001	The type of destination IP address in this structure is IPv4 address. The fields after Reserved field in this structure are interpreted as IPv4Address followed by Reserved2 as described below.	MOVE_DST_IPADDR_V6		0x00000002	The type of destination IP address in this structure is IPv6 address. The field after Reserved field in this structure is interpreted as IPv6Address as described below.	Value	Meaning	MOVE_DST_IPADDR_V4		0x00000001	The type of destination IP address in this structure is IPv4 address.	MOVE_DST_IPADDR_V6		0x00000002	The type of destination IP address in this structure is IPv6 address.
Value	Meaning																				
MOVE_DST_IPADDR_V4																					
0x00000001	The type of destination IP address in this structure is IPv4 address. The fields after Reserved field in this structure are interpreted as IPv4Address followed by Reserved2 as described below.																				
MOVE_DST_IPADDR_V6																					
0x00000002	The type of destination IP address in this structure is IPv6 address. The field after Reserved field in this structure is interpreted as IPv6Address as described below.																				
Value	Meaning																				
MOVE_DST_IPADDR_V4																					
0x00000001	The type of destination IP address in this structure is IPv4 address.																				
MOVE_DST_IPADDR_V6																					
0x00000002	The type of destination IP address in this structure is IPv6 address.																				
2020/11/10	<p>In Section 3.2.1.1 Global, added ShareList element.</p> <p>In Section 3.2.1.3 Per Session, added IsAnonymous and IsGuest Boolean session elements.</p> <p>In Section 3.2.1.10 Per Share, added Section.</p> <p>In Section 3.2.3 Initialization, added ShareList element setting.</p> <p>In Section 3.2.4.1.8 Encrypting the Message, added processing for IsAnonymous and IsGuest user sessions.</p>																				

Errata Published*	Description
	<p>In Section 3.2.5.3.1 Handling a New Authentication, added processing for IsAnonymous and IsGuest user sessions.</p> <p>In Section 3.2.5.5 Receiving an SMB2 TREE_CONNECT Response, added processing for the ShareList element.</p> <p>In Section 3.3.5.2.1.1 Decrypting the Message, added processing for IsAnonymous and IsGuest user sessions.</p> <p>Please see the diff pdf file for details.</p>
2020/10/26	<p>In Section 3.3.4.1.5 Compressing the Message, the processing rules for Connection.CompressionIds was updated.</p> <p>Changed from:</p> <p>If Connection.Dialect is 3.1.1, IsCompressionSupported is TRUE, Connection.CompressionIds is not empty, and Request.CompressReply is TRUE, the server MUST process the message as specified in section 3.1.4.4, before sending it to the client.</p> <p>Changed to:</p> <p>If Connection.Dialect is 3.1.1, IsCompressionSupported is TRUE, Connection.CompressionIds is not empty, and Request.CompressReply is TRUE, the server SHOULD<206> process the message as specified in section 3.1.4.4, before sending it to the client.</p> <p><206> Section 3.3.4.1.5: Windows 10 v2004, Windows 10 v20H2, Windows Server v2004, and Windows Server v20H2 do not compress the message if Connection.CompressionIds does not include LZNT1, LZ77 and LZ77+Huffman algorithms.</p>
2020/10/12	<p>In section 3.3.5.4 Receiving an SMB2 NEGOTIATE Request the processing rules for CompressionAlgorithmCount and CompressionAlgorithms were updated.</p> <p>Changed from:</p> <ul style="list-style-type: none"> • Set CompressionAlgorithmCount to 1. • Set CompressionAlgorithms to "NONE". <p>Changed to:</p> <ul style="list-style-type: none"> • The server SHOULD<251> set CompressionAlgorithmCount to 1. • The server SHOULD<252> set CompressionAlgorithms to "NONE". <p><251> Section 3.3.5.4: Windows 10 v2004 and Windows Server v2004 set CompressionAlgorithmCount to 0.</p> <p><252> Section 3.3.5.4: Windows 10 v2004 and Windows Server v2004 set CompressionAlgorithms to empty.</p>
2020/09/28	<p>In Section 2.2.13 SMB2 CREATE Request, the Buffer field was changed.</p>

Errata Published*	Description
	<p>Changed from:</p> <p>Buffer (variable): A variable-length buffer that contains the Unicode file name and create context list, as defined by NameOffset, NameLength, CreateContextsOffset, and CreateContextsLength. In the request, the Buffer field MUST be at least one byte in length. The file name (after DFS normalization if needed) MUST conform to the specification of a relative pathname in [MS-FSCC] section 2.1.5.</p> <p>Changed to:</p> <p>Buffer (variable): A variable-length buffer that contains the Unicode file name and create context list, as defined by NameOffset, NameLength, CreateContextsOffset, and CreateContextsLength. In the request, the Buffer field MUST be at least one byte in length.</p> <p>In Section 3.2.4.3 Application Requests Opening a File, was updated with the following processing rule:</p> <p>The client MUST conform to the specification in [MS-FSCC] section 2.1.5 for the application-supplied path name.</p> <p>In Section 3.3.5.9 Receiving an SMB2 CREATE Request, was updated with the following processing rules:</p> <p>The server MUST fail the request with STATUS_INVALID_PARAMETER in the following cases:</p> <ul style="list-style-type: none"> • If NameLength number of bytes in the Buffer field extends beyond the CREATE request received. • If NameLength is not a multiple of 2. • If NameOffset is less than the Buffer field offset. <p>Product behavior note 266 was updated.</p> <p>Changed from:</p> <p><266> Section 3.3.5.9: Windows-based servers fail the CREATE request with STATUS_INVALID_PARAMETER if the file name in the Buffer field of the request begins in the form "subfolder\..\\", for example "x\..\y.txt".</p> <p>Changed to:</p> <p><266> Section 3.3.5.9: Windows-based servers accept the path names containing Dot Directory Names specified in [MS-FSCC]: section 2.1.5.1 and attempt to normalize the path name by removing the pathname components of "." and "..". Windows-based servers fail the CREATE request with STATUS_INVALID_PARAMETER if the file name in the Buffer field of the request begins in the form "subfolder\..\\", for example "x\..\y.txt".</p>
2020/09/28	<p>In Section 3.2.5.2 Receiving an SMB2 NEGOTIATE Response, the processing rules for compression were updated.</p> <p>Changed from:</p>

Errata Published*	Description
	<p>...</p> <ul style="list-style-type: none"> • For each algorithm in CompressionAlgorithms, if the value of algorithm is greater than 32, the client MUST return an error to the calling application. • If there is a duplicate value in CompressionAlgorithms, the client MUST return an error to the calling application. • If CompressionAlgorithmCount is 1 and CompressionAlgorithms contains "NONE", the client MUST set Connection.CompressionIds to an empty list. <p>...</p> <p>Changed to:</p> <p>...</p> <ul style="list-style-type: none"> • For each algorithm in CompressionAlgorithms, if the value of algorithm is greater than or equal to 32, the client MUST return an error to the calling application. • If there is a duplicate value in CompressionAlgorithms, the client MUST return an error to the calling application. • If CompressionAlgorithmCount is 1 and CompressionAlgorithms contains "NONE", the client SHOULD<162> set Connection.CompressionIds to an empty list. <p><162> Section 3.2.5.2: Windows 10 v1903 through Windows 10 v20H2 and Windows Server v1903 through Windows Server v20H2 will disconnect the connection.</p> <p>...</p>
2020/09/28	<p>In Section 2.2.19 SMB2 READ Request, the following was changed from:</p> <p>Changed from:</p> <p>MinimumCount (4 bytes): The minimum number of bytes to be read for this operation to be successful. If fewer than the minimum number of bytes are read by the server, the server MUST return an error rather than the bytes read.</p> <p>Changed to:</p> <p>MinimumCount (4 bytes): The minimum number of bytes to be read for this operation to be successful.</p> <p>In Section 3.2.4.6 Application Requests Reading from a File or Named Pipe, the following was changed from:</p> <p>Changed from:</p>

Errata Published*	Description
	<ul style="list-style-type: none"> • The MinimumCount field is set to the value that is provided by the application. If no value is provided by the application, the client MUST set this field to 0. <p>Changed to:</p> <ul style="list-style-type: none"> • The client SHOULD<127> set MinimumCount field to the value that is provided by the application. If no value is provided by the application, the client SHOULD set this field to 0. <p><127> Section 3.2.4.6: Windows-based clients set MinimumCount field to 0.</p> <p>In Section 3.3.5.12 Receiving an SMB2 READ Request, the following was changed from:</p> <p>Changed from:</p> <ul style="list-style-type: none"> • DataOffset MUST be set to the offset into the response, in bytes, from the beginning of the SMB2 header where the data is located. • The data MUST be copied into the response. • DataLength MUST be set to the number of bytes returned. • DataRemaining MUST be set to zero. <p>...</p> <ul style="list-style-type: none"> • The data MUST be sent via the processing specified in [MS-SMBD] section 3.1.4.5 RDMA Write to Peer Buffer, providing the Connection, the data, and the array of SMB_DIRECT_BUFFER_DESCRIPTOR_V1 structures passed in the request at offset ReadChannelInfoOffset and of length ReadChannelInfoLength fields. • The DataOffset field MUST be set to the offset into the response, in bytes, from the beginning of the SMB2 header to the Buffer field. • The data MUST NOT be copied into the response. • DataRemaining MUST be set to the number of bytes returned via RDMA. <p><317> Section 3.3.5.12: Windows reads from a file with Server Requests a Read [MS-FSA] section 2.1.5.2.</p> <p>Object Store parameter SMB2 parameter</p> <p>ByteOffset ByteOffset</p> <p>ByteCount ByteCount</p> <p>Open Open.Local</p> <p>Key 0</p>

Errata Published*	Description
	<p>Unbuffered Set to TRUE if SMB2_READFLAG_READ_UNBUFFERED is set in the Flags field of the request, otherwise set to FALSE.</p> <p>Changed to:</p> <p>If the read fails, the server MUST fail the request using the error code received from the read operation. If the underlying object store returns fewer bytes than specified by the MinimumCount field of the request, the server MUST fail the request with STATUS_END_OF_FILE.</p> <p>If the read succeeds, the server MUST construct a read response using the syntax specified in section 2.2.20 with the following values.</p> <p>If the request Channel field contains the value SMB2_CHANNEL_NONE, then:</p> <ul style="list-style-type: none"> • DataOffset MUST be set to the offset into the response, in bytes, from the beginning of the SMB2 header where the data is located. • If the number of bytes returned from the underlying object store is more than the Length field in the request, DataLength MUST be set to the Length field of the request. Otherwise, DataLength MUST be set to the number of bytes returned from the underlying object store. • The data MUST be copied into the response. • DataRemaining MUST be set to zero. <p>...</p> <ul style="list-style-type: none"> • The DataOffset field MUST be set to the offset into the response, in bytes, from the beginning of the SMB2 header to the Buffer field. • If the number of bytes returned from the underlying object store is more than the Length field in the request, DataRemaining MUST be set to the Length field of the request. Otherwise, DataRemaining MUST be set to the number of bytes returned from the underlying object store. • The data MUST NOT be copied into the response. • The data MUST be sent via the processing specified in [MS-SMBD] section 3.1.4.5, providing the Connection, the data, and the array of SMB_DIRECT_BUFFER_DESCRIPTOR_V1 structures passed in the request at offset ReadChannelInfoOffset and of length ReadChannelInfoLength fields. <p><317> Section 3.3.5.12: Windows reads from a file with Server Requests a Read [MS-FSA] section 2.1.5.2.</p> <p>Object Store parameter SMB2 parameter</p> <p>ByteOffset Offset</p> <p>ByteCount Length</p>

Errata Published*	Description
	<p>Open Open.Local</p> <p>Key 0</p> <p>Unbuffered Set to TRUE if SMB2_READFLAG_READ_UNBUFFERED is set in the Flags field of the request, otherwise set to FALSE.</p>
2020/08/31	<p>In MS-SMB2 Section 6 Appendix A: Product Behavior, note 133 has been changed.</p> <p>Changed from:</p> <p><133> Section 3.2.4.9: In a SET_INFO request where FileInfoClass is set to FileRenameInformation, and the size of the buffer is less than 24, Windows clients pad the buffer to 24 bytes. These padding bytes are set to arbitrary values. Windows Vista SP1, Windows Server 2008, Windows 7, and Windows Server 2008 R2 clients append up to 4 additional padding bytes set to arbitrary values.</p> <p>Changed to:</p> <p><133> Section 3.2.4.9: In a SET_INFO request where FileInfoClass is set to FileRenameInformation, Windows Vista SP1, Windows Server 2008, Windows 7, and Windows Server 2008 R2 clients append up to 4 additional padding bytes set to arbitrary values.</p>
2020/08/31	<p>In Section 3.3.5.9 Receiving an SMB2 CREATE Request, the following was updated</p> <p>Changed from:</p> <p>If the file name length is greater than zero and the first character is a path separator character, the server MUST fail the request with STATUS_INVALID_PARAMETER. If the file name fails to conform with the specification of a relative pathname in [MS-FSCC] section 2.1.5, the server MUST fail the request with STATUS_OBJECT_NAME_INVALID.</p> <p>Changed to:</p> <p>If the file name in the Buffer field of the request fails to resolve the pathname components as specified in [MS-FSCC] section 2.1.5.1, the server SHOULD<262> fail the request with STATUS_INVALID_PARAMETER.</p> <p><262> Section 3.3.5.9: Windows-based servers fail the CREATE request with STATUS_INVALID_PARAMETER if the file name in the Buffer field of the request begins in the form "subfolder\..\\", for example "x\..\y.txt".</p>

*Date format: YYYY/MM/DD