

[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.

Errata below are for Protocol Document Version [V48.0 – 2015/10/16](#).

Errata Published*	Description
2016/06/27	<p>In various sections, updated the content to clarify support for SMB2 or higher over NetBT (port 139).</p> <p>In Section 2.1, Transport, changed from:</p> <p>SMB2Message (variable): The body of the SMB2 packet. The length of an SMB2Message varies based on the SMB2 command represented by the message.</p> <ul style="list-style-type: none">• SMB2 dialects 2.0.2, 2.1, 3.0, and 3.0.2 support NetBIOS over TCP [RFC1001] [RFC1002].• SMB2 dialects 3.0, 3.0.2, and 3.1.1 support operation over SMB2 RDMA Transport [MS-SMBD]. <p>Changed to:</p> <p>SMB2Message (variable): The body of the SMB2 packet. The length of an SMB2Message varies based on the SMB2 command represented by the message.</p> <ul style="list-style-type: none">• SMB2 dialects 2.0.2, 2.1, 3.0, and 3.0.2 allow NetBIOS over TCP [RFC1001] [RFC1002].• SMB2 dialects 3.0, 3.0.2, and 3.1.1 allow operation over SMB2 RDMA Transport [MS-SMBD]. <p>In Section 3.2.4.2.1, Connecting to the Target Server, changed from:</p> <p>The client MUST attempt to connect to the target server over the registered transports specified in section 2.1 and [MS-SMB] section 2.1. The ServerName and the optional TransportIdentifier provided by the caller are used to establish the connection. The client SHOULD resolve the ServerName as described in [MS-WPO] section 7.1.4, and SHOULD attempt connections to one or more of the returned addresses. The client can attempt to initiate each such SMB2 connection on all configured transports that it supports, most commonly Direct TCP and the other transports described in section 2.1.</p> <p>The client can choose to prioritize the addresses and/or transport order and try each one sequentially, or try to connect on them all and select one using any implementation-specific heuristic<102>. The client can accept the TransportIdentifier parameter from the calling application, which specifies what transport to use, and then attempt to use the transport specified. If the connection attempt is successful, a connection object MUST be created, as specified in section 3.2.1.2, with the following default parameters:</p> <p>...</p> <p>This connection MUST be inserted into ConnectionTable, and processing MUST continue, as specified in section 3.2.4.2.2.</p> <p>If the connection attempt fails, the client returns the error code to the calling application.</p> <p>If the client implements the SMB 3.x dialect family, the client MUST look up a server entry in ServerList where Server.ServerName matches the ServerName to which the connection is established. If an entry is found, the client MUST set Connection.Server to the server entry found. Otherwise the client MUST initialize a server object and MUST set Server.ServerName to ServerName and Connection.Server to NULL.</p>

Errata Published*	Description
	<p>Changed to:</p> <p>The ServerName and the optional TransportIdentifier provided by the caller are used to establish the connection. The client SHOULD resolve the ServerName as described in [MS-WPO] section 7.1.4, and SHOULD attempt connections to one or more of the returned addresses. The client can attempt to initiate each such SMB2 connection on all configured transports that it allows<102>, most commonly Direct TCP and the other transports described in section 2.1.</p> <p><102> Section 3.2.4.2.1: Windows clients initiate new transport connections to the server with Direct TCP and NetBIOS over TCP. Windows Server 2012, Windows Server 2012 R2 operating system, Windows Server 2016, and Windows 10 v1511 Enterprise operating system do not initiate a new transport connection with RDMA, but do after a multichannel exchange if a suitable interface is available.</p> <p>The client can choose to prioritize the addresses and/or transport order and try each one sequentially, or try to connect on them all and select one using any implementation-specific heuristic<103>. The client can accept the TransportIdentifier parameter from the calling application, which specifies what transport to use, and then attempt to use the transport specified. If the connection attempt is successful, a connection object MUST be created, as specified in section 3.2.1.2, with the following default parameters:</p> <p>...</p> <p>This connection MUST be inserted into ConnectionTable, and processing MUST continue, as specified in section 3.2.4.2.2.</p> <p>If the connection attempt fails, the client returns the error code to the calling application.</p> <p>In Section 3.2.5.2, Receiving an SMB2 NEGOTIATE Response, changed from:</p> <p>...</p> <p>If the client implements the SMB 3.x dialect family and Connection.Server is not NULL, the client MUST disconnect the connection if any of the following conditions is satisfied:</p> <p>...</p> <p>If the client implements the SMB 3.x dialect family and Connection.Server is NULL, the client MUST set the following values:</p> <ul style="list-style-type: none"> • Connection.Server.ServerGUID to ServerGUID in the response <p>Changed to:</p> <p>...</p> <p>If the client implements the SMB 3.x dialect family, the client MUST look up the server entry in ServerList where Server.ServerName matches the Connection.ServerName. If an entry is found, the client MUST set Connection.Server to the server entry found. Otherwise, the client MUST initialize a server object and MUST set Server.ServerName to Connection.ServerName and Connection.Server to NULL. The client MUST add the Server entry to ServerList.</p> <p>If the client implements the SMB 3.x dialect family and Connection.Server is not NULL, the client MUST disconnect the connection if any of the following conditions is satisfied:</p> <p>...</p> <p>If the client implements the SMB 3.x dialect family and Connection.Server is NULL, the client MUST set the following values:</p> <ul style="list-style-type: none"> • Connection.Server to the server entry in ServerList where Server.ServerName matches the Connection.ServerName. • Connection.Server.ServerGUID to ServerGUID in the response.
2016/06/27	<p>In Section 3.3.5.9, Receiving an SMB2 CREATE Request, corrected the processing rules.</p> <p>Changed from:</p>

Errata Published*	Description
	<ul style="list-style-type: none"> • If Open.IsDurable is TRUE or Open.IsResilient is TRUE, Open.DurableFileId is set to a generated value that uniquely identifies this open in GlobalOpenTable. Otherwise, Open.DurableFileId is set to a generated value that uniquely identifies this Open in Session.OpenTable. <p>Changed to:</p> <ul style="list-style-type: none"> • Open.DurableFileId is set to a generated value that uniquely identifies this open in GlobalOpenTable.
2016/06/27	<p>In Section 3.3.4.7, Object Store Indicates a Lease Break, corrected the processing rules.</p> <p>Changed from:</p> <p>If Open.Connection is NULL, Open.IsResilient is FALSE, and Open.IsPersistent is FALSE, the server SHOULD close the Open as specified in section 3.3.4.17.</p> <p>Otherwise, if Lease.BreakToLeaseState does not contain SMB2_LEASE_HANDLE_CACHING and Open.IsDurable is TRUE, the server MUST close the Open as specified in section 3.3.4.17.</p> <p>Changed to:</p> <p>If Open.Connection is NULL, the server MUST close the Open as specified in section 3.3.4.17 for the following cases:</p> <ul style="list-style-type: none"> • Open.IsResilient is FALSE, Open.IsDurable is FALSE, and Open.IsPersistent is FALSE. • Lease.BreakToLeaseState does not contain SMB2_LEASE_HANDLE_CACHING and Open.IsDurable is TRUE.
2016/05/16	<p>In 4 sections, updated the text to remove the implication that resiliency requires leasing.</p> <p>In Section 3.3.1.5, Global, changed from:</p> <p>If the server implements the SMB 2.1 or SMB 3.x dialect family and supports leasing, it MUST implement the following:</p> <ul style="list-style-type: none"> ▪ GlobalLeaseTableList: A list of all the lease tables as described in 3.3.1.11, indexed by the ClientGuid. ▪ MaxResiliencyTimeout: The maximum resiliency time-out in milliseconds, for the TimeOut field of NETWORK_RESILIENCY_REQUEST Request, as specified in section 2.2.31.3. <p>Changed to:</p> <p>If the server implements the SMB 2.1 or SMB 3.x dialect family and supports leasing, it MUST implement the following:</p> <ul style="list-style-type: none"> ▪ GlobalLeaseTableList: A list of all the lease tables as described in 3.3.1.11, indexed by the ClientGuid. <p>If the server implements the SMB 2.1 or SMB 3.x dialect family and supports resiliency, it MUST implement the following:</p> <ul style="list-style-type: none"> ▪ MaxResiliencyTimeout: The maximum resiliency time-out in milliseconds, for the TimeOut field of NETWORK_RESILIENCY_REQUEST Request, as specified in section 2.2.31.3. <p>In Section 3.3.3, Initialization, changed from:</p> <p>If the server implements the SMB 2.1 or 3.x dialect family and supports leasing, the server MUST initialize the following:</p>

Errata Published*	Description
	<ul style="list-style-type: none"> ▪ GlobalLeaseTableList MUST be set to an empty list. ▪ MaxResiliencyTimeout SHOULD<185> be set to an implementation-specific default value. <p>Changed to:</p> <p>If the server implements the SMB 2.1 or 3.x dialect family and supports leasing, the server MUST initialize the following:</p> <ul style="list-style-type: none"> ▪ GlobalLeaseTableList MUST be set to an empty list. <p>If the server implements the SMB 2.1 or SMB 3.x dialect family and supports resiliency, it MUST implement the following:</p> <ul style="list-style-type: none"> ▪ MaxResiliencyTimeout SHOULD<185> be set to an implementation-specific default value. <p>In Section 3.3.5.14.1, Processing Unlocks, changed from:</p> <p>If the unlock operation succeeds and there are no remaining entries in the Locks array, Connection.Dialect is "2.1" or belongs to the SMB 3.x dialect family, the server supports leasing, and Open.IsResilient is TRUE, the server MUST set the lock sequence number in Open.LockSequenceArray through the following step to indicate that the unlock request with LockSequence has been successfully processed by the server:</p> <p>Changed to:</p> <p>If the unlock operation succeeds and there are no remaining entries in the Locks array, Connection.Dialect is "2.1" or belongs to the SMB 3.x dialect family, and Open.IsResilient or Open.IsPersistent is TRUE, the server MUST set the lock sequence number in Open.LockSequenceArray through the following step to indicate that the unlock request with LockSequence has been successfully processed by the server:</p> <p>In Section 3.3.5.14.2, Processing Locks, changed from:</p> <p>If the lock operation succeeds and there are no remaining entries in the Locks array, Connection.Dialect is "2.1" or belongs to the SMB 3.x dialect family, the server supports leasing, and Open.IsResilient is TRUE, the server MUST set the lock sequence number in Open.LockSequenceArray through the following step to indicate that the lock request with LockSequence has been successfully processed by the server:</p> <p>Changed to:</p> <p>If the lock operation succeeds and there are no remaining entries in the Locks array, Connection.Dialect is "2.1" or belongs to the SMB 3.x dialect family, and Open.IsResilient or Open.IsPersistent is TRUE, the server MUST set the lock sequence number in Open.LockSequenceArray through the following step to indicate that the lock request with LockSequence has been successfully processed by the server:</p>
2016/04/18	<p>In Section 2.2.13, SMB2 CREATE Request, clarified the use of "can" in the description of the NameOffset field.</p> <p>Changed from:</p> <p>NameOffset (2 bytes): The offset, in bytes, from the beginning of the SMB2 header to the 8-byte aligned file name. If SMB2_FLAGS_DFS_OPERATIONS is set in the Flags field of the SMB2 header, the file name can be prefixed with Distributed File System (DFS) link information that will be removed during DFS name normalization as specified in section 3.3.5.9. Otherwise, the</p>

Errata Published*	Description
	<p>file name is relative to the share that is identified by the TreeId in the SMB2 header. The NameOffset field SHOULD be set to the offset of the Buffer field from the beginning of the SMB2 header. The file name (after DFS normalization if needed) MUST conform to the specification of a relative pathname in [MS-FSCC] section 2.1.5. A zero length file name indicates a request to open the root of the share.</p> <p>Changed to:</p> <p>NameOffset (2 bytes): The offset, in bytes, from the beginning of the SMB2 header to the 8-byte aligned file name. If SMB2_FLAGS_DFS_OPERATIONS is set in the Flags field of the SMB2 header, the file name includes a prefix that will be processed during DFS name normalization as specified in section 3.3.5.9. Otherwise, the file name is relative to the share that is identified by the TreeId in the SMB2 header. The NameOffset field SHOULD be set to the offset of the Buffer field from the beginning of the SMB2 header. The file name (after DFS normalization if needed) MUST conform to the specification of a relative pathname in [MS-FSCC] section 2.1.5. A zero length file name indicates a request to open the root of the share.</p>
2016/04/18	<p>In Section 3.3.5.9.14, Handling the SVHDX_OPEN_DEVICE_CONTEXT Create Context, corrected the processing rules by adding a new paragraph and removing a second.</p> <p>Changed from:</p> <p>This section applies only to servers that implement the SMB 3.0.2 or SMB 3.1.1 dialect. If the create request has any other created contexts, the server MUST process those create contexts before processing the SVHDX_OPEN_DEVICE_CONTEXT.</p> <p>If IsSharedVHDSupported is FALSE, the server MUST fail the request with STATUS_INVALID_DEVICE_REQUEST.</p> <p>Changed to:</p> <p>This section applies only to servers that implement the SMB 3.0.2 or SMB 3.1.1 dialect. If IsSharedVHDSupported is FALSE, the server MUST ignore the create context. If the create request has any other create contexts, the server MUST process those create contexts before processing the SVHDX_OPEN_DEVICE_CONTEXT.</p>
2016/04/04	<p>In Section 3.3.5.9.7, Handling the SMB2_CREATE_DURABLE_HANDLE_RECONNECT Create Context, updated the product behavior note in processing step 3 to clarify behavior for establishing a durable handle reconnect when looking up an existing open.</p> <p>Changed from:</p> <p>The processing changes involved for this create context are:</p> <p>...</p> <p>3. The server MUST look up an existing open in the GlobalOpenTable by doing a lookup with the FileId.Persistent portion of the create context. If the lookup fails, the server SHOULD<275> fail the request with STATUS_OBJECT_NAME_NOT_FOUND and proceed as specified in "Failed Open Handling" in section 3.3.5.9.</p> <p><275> Section 3.3.5.9.7: Windows Vista SP1, Windows Server 2008, Windows 7 and Windows Server 2008 R2 ignore undefined create contexts.</p> <p>...</p> <p>Changed to:</p> <p>The processing changes involved for this create context are:</p> <p>...</p>

Errata Published*	Description
	<p>3. The server MUST look up an existing open in the GlobalOpenTable by doing a lookup with the FileId.Persistent portion of the create context. If the lookup fails, the server SHOULD<275> fail the request with STATUS_OBJECT_NAME_NOT_FOUND and proceed as specified in "Failed Open Handling" in section 3.3.5.9.</p> <p><275> Section 3.3.5.9.7: If the Session was established by invalidating the previous session by specifying PreviousSessionId in the SMB2 SESSION_SETUP request, Windows 8.1 and Windows Server 2012 R2 close the durable opens established on the previous session.</p>
2016/04/04	<p>In Section 3.2.4.4, Re-establishing a Durable Open, corrected "Open.ShareAccess" to "Open.ShareMode".</p> <p>Changed from: The SMB2 CREATE Request MUST be initialized as follows: If Connection.Dialect belongs to the SMB 3.x dialect family, the client MUST set the following: ... <ul style="list-style-type: none"> ▪ The client sets the ShareAccess field to Open.ShareAccess. </p> <p>Changed to: The SMB2 CREATE Request MUST be initialized as follows: If Connection.Dialect belongs to the SMB 3.x dialect family, the client MUST set the following: ... <ul style="list-style-type: none"> ▪ The client sets the ShareAccess field to Open.ShareMode. </p>
2016/04/04	<p>In Section 3.3.5.9.14, Handling the SVHDX_OPEN_DEVICE_CONTEXT Create Context, clarified the processing rules for Open.IsSharedVHDX and ":SharedVirtualDisk".</p> <p>Changed from:</p> <p>If IsSharedVHDSupported is TRUE, the processing changes involved for this create context are: ... <ul style="list-style-type: none"> ▪ In the "Response Construction" phase: <ul style="list-style-type: none"> ▪ If the RSVD server has returned a response create context, as specified in [MS-RSVD] sections 2.2.4.31 and 2.2.4.33, the server MUST include it in the buffer described by the response CreateContextLength and CreateContextOffset fields. </p> <p>Changed to:</p> <p>If IsSharedVHDSupported is TRUE and the file name in the Buffer field ends with ":SharedVirtualDisk", the processing changes involved for this create context are: ... <ul style="list-style-type: none"> ▪ In the "Response Construction" phase: <ul style="list-style-type: none"> ▪ If the RSVD server has returned a response create context, as specified in [MS-RSVD] sections 2.2.4.31 and 2.2.4.33, the server MUST include it in the buffer described by the response CreateContextLength and CreateContextOffset fields. </p> <p>If IsSharedVHDSupported is TRUE and the file name in the Buffer field does not end with ":SharedVirtualDisk", the processing changes involved for this create context are: <ul style="list-style-type: none"> ▪ The server MUST set Open.IsSharedVHDX to FALSE. ▪ If OriginatorFlags in SVHDX_OPEN_DEVICE_CONTEXT is set to SVHDX_ORIGINATOR_VHDMP, the server MUST fail the request with STATUS_VHD_SHARED. Otherwise, the create operation MUST be ignored </p>

Errata Published*	Description
2016/03/21	<p>In Section 3.3.3, Initialization, changed "implement" and "perform" to "initialize" in keeping with the purpose of the section.</p> <p>Changed from:</p> <p>The server MUST implement the following:</p> <ul style="list-style-type: none"> ▪ All the members in ServerStatistics MUST be set to zero. <p>...</p> <p>If the server implements the SMB 2.1 or 3.x dialect family and supports leasing, the server MUST implement the following:</p> <ul style="list-style-type: none"> ▪ GlobalLeaseTableList MUST be set to an empty list. <p>...</p> <p>If the server implements the SMB 3.x dialect family, the server MUST implement the following:</p> <ul style="list-style-type: none"> ▪ EncryptData MUST be set in an implementation-specific manner. <p>...</p> <p>If the server implements the SMB 3.0.2 or SMB 3.1.1 dialect, the server MUST implement the following:</p> <ul style="list-style-type: none"> ▪ IsSharedVHDSupported: MUST be set to FALSE. <p>If the server implements the SMB 3.1.1 dialect, the server MUST perform the following:</p> <ul style="list-style-type: none"> ▪ MaxClusterDialect MUST be set in an implementation-specific manner. <p>...</p> <p>Changed to:</p> <p>The server MUST initialize the following:</p> <ul style="list-style-type: none"> ▪ All the members in ServerStatistics MUST be set to zero. <p>...</p> <p>If the server implements the SMB 2.1 or 3.x dialect family and supports leasing, the server MUST initialize the following:</p> <ul style="list-style-type: none"> ▪ GlobalLeaseTableList MUST be set to an empty list. <p>...</p> <p>If the server implements the SMB 3.x dialect family, the server MUST initialize the following:</p> <ul style="list-style-type: none"> ▪ EncryptData MUST be set in an implementation-specific manner. <p>...</p> <p>If the server implements the SMB 3.0.2 or SMB 3.1.1 dialect, the server MUST initialize the following:</p> <ul style="list-style-type: none"> ▪ IsSharedVHDSupported: MUST be set to FALSE. <p>If the server implements the SMB 3.1.1 dialect, the server MUST initialize the following:</p> <ul style="list-style-type: none"> ▪ MaxClusterDialect MUST be set in an implementation-specific manner. <p>...</p>
2016/01/25	<p>In Section 3.3.5.9, Receiving an SMB2 CREATE Request, the first paragraph under Open Execution has been changed from:</p> <p>If the FILE_DELETE_ON_CLOSE flag is set in CreateOptions and any of the following conditions is TRUE, the server SHOULD<248> fail the request with STATUS_ACCESS_DENIED.</p> <ul style="list-style-type: none"> ▪ DesiredAccess does not include DELETE or GENERIC_ALL. ▪ Treeconnect.MaximalAccess does not include DELETE or GENERIC_ALL. <p>Changed to:</p>

Errata Published*	Description
	<p>If the FILE_DELETE_ON_CLOSE flag is set in CreateOptions and Treeconnect.MaximalAccess does not include DELETE or GENERIC, the server SHOULD<248> fail the request with STATUS_ACCESS_DENIED.</p>
2016/01/11	<p>In Section 3.3.5.10, Receiving an SMB2 CLOSE Request, modified the processing rules to account for Windows behavior. The 6 and 7th paragraphs have been changed from:</p> <p>The server MUST locate the Request in Connection.RequestList for which Request.MessageId matches the MessageId value in the SMB2 header, set Request.Open to the Open, and close the Open as specified in section 3.3.4.17.</p> <p>If SMB2_CLOSE_FLAG_POSTQUERY_ATTRIB is set in the Flags field of the request, the server MUST query the attributes of the file after the close.<291> This gives the client the attributes that have been updated to take into account any cached writes or extends that may have happened. The attributes that MUST be queried are the creation time, last access time, last write time, change time, allocation size in bytes, end of file in bytes, and file attributes.</p> <p><291> Section 3.3.5.10: Windows obtains attributes and end of file from the object store FileBasicInformation [MS-FSA] section 2.1.5.11.6 and [MS-FSCC] section 2.4.7.</p> <p>Changed to:</p> <p>The server MUST locate the Request in Connection.RequestList for which Request.MessageId matches the MessageId value in the SMB2 header and set Request.Open to the Open.</p> <p>If SMB2_CLOSE_FLAG_POSTQUERY_ATTRIB is set in the Flags field of the request, the server MUST query the creation time, last access time, last write time, change time, allocation size in bytes, end of file in bytes, and file attributes of the file from the underlying object store in an implementation-specific manner<290>.</p> <p>The server MUST close the Open as specified in section 3.3.4.17.</p> <p><290> Section 3.3.5.10: Windows obtains FileNetworkOpenInformation from the object store as described in [MS-FSA] section 2.1.5.11.21 and [MS-FSCC] section 2.4.27.</p> <p>Windows servers do not return an updated ChangeTime unless Open.GrantedAccess includes FILE_WRITE_DATA, FILE_WRITE_ATTRIBUTES, FILE_WRITE_EA, or FILE_APPEND_DATA and any prior WRITE/SET_INFO operations were performed on that Open.</p>
2016/01/11	<p>In Section 3.2.7.1, Handling a Network Disconnect, modified a processing rule.</p> <p>Changed from:</p> <ul style="list-style-type: none"> ▪ If Connection.Dialect belongs to the SMB 3.x dialect family, and if the Session has more than one channel in Session.ChannelList, the client MUST perform the following actions: ... <p>Changed to:</p> <ul style="list-style-type: none"> ▪ If Connection.Dialect belongs to the SMB 3.x dialect family, and if Connection.SupportsMultiChannel or Connection.SupportsPersistentHandles is TRUE, the client MUST perform the following actions: ...
2016/01/11	<p>In various sections, added information about the disconnect behavior of Windows clients.</p>

Errata Published*	Description
	<p>In Section 3.2.5.2, Receiving an SMB2 NEGOTIATE Response, a new paragraph was added:</p> <p>The client SHOULD<154> disconnect the connection if the size, in bytes, received in MaxTransactSize, MaxReadSize, or MaxWriteSize is less than 65536.</p> <p><154> Section 3.2.5.2: Windows Vista SP1, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2 disconnect the connection if MaxTransactSize, MaxReadSize, or MaxWriteSize is less than 4096.</p> <p>In Section 3.3.5.3.1, SMB 2.1 or SMB 3.x Support, the 6th, 7th, and 8th bullet points of the first list were changed from:</p> <ul style="list-style-type: none"> ▪ MaxTransactSize is set to the maximum buffer size<220>, in bytes, that the server will accept on this connection for QUERY_INFO, QUERY_DIRECTORY, SET_INFO, and CHANGE_NOTIFY operations. This field is applicable only for buffers sent by the client in SET_INFO requests, or returned from the server in QUERY_INFO, QUERY_DIRECTORY, and CHANGE_NOTIFY responses. Connection.MaxTransactSize MUST be set to MaxTransactSize. ▪ MaxReadSize is set to the maximum size, in bytes, of the Length in an SMB2 READ Request (2.2.19) that the server will accept on the transport that established this connection.<221> Connection.MaxReadSize MUST be set to MaxReadSize. ▪ MaxWriteSize is set to the maximum size, in bytes, of the Length in an SMB2 Write Request (2.2.21) that the server will accept on the transport that established this connection.<222> Connection.MaxWriteSize MUST be set to MaxWriteSize. <p>Changed to:</p> <ul style="list-style-type: none"> ▪ MaxTransactSize is set to the maximum buffer size, in bytes, that the server will accept on this connection for QUERY_INFO, QUERY_DIRECTORY, SET_INFO, and CHANGE_NOTIFY operations. This field is applicable only for buffers sent by the client in SET_INFO requests, or returned from the server in QUERY_INFO, QUERY_DIRECTORY, and CHANGE_NOTIFY responses. This value SHOULD<221> be greater than or equal to 65536. Connection.MaxTransactSize MUST be set to MaxTransactSize. ▪ MaxReadSize is set to the maximum size, in bytes, of the Length in an SMB2 READ Request (2.2.19) that the server will accept on the transport that established this connection. This value SHOULD<222> be greater than or equal to 65536. Connection.MaxReadSize MUST be set to MaxReadSize. ▪ MaxWriteSize is set to the maximum size, in bytes, of the Length in an SMB2 Write Request (2.2.21) that the server will accept on the transport that established this connection. This value SHOULD<223> be greater than or equal to 65536. Connection.MaxWriteSize MUST be set to MaxWriteSize. <p>In Section 3.3.5.4, Receiving an SMB2 NEGOTIATE Request, the 6th, 7th, and 8th bullet points of the second list were changed from:</p> <ul style="list-style-type: none"> ▪ MaxTransactSize is set to the maximum buffer size,<228> in bytes, that the server will accept on this connection for QUERY_INFO, QUERY_DIRECTORY, SET_INFO and CHANGE_NOTIFY operations. This field is applicable only for buffers sent by the client in SET_INFO requests, or returned from the server in QUERY_INFO, QUERY_DIRECTORY,

Errata Published*	Description
	<p>and CHANGE_NOTIFY responses. Connection.MaxTransactSize MUST be set to MaxTransactSize.</p> <ul style="list-style-type: none"> ▪ MaxReadSize is set to the maximum size, <229> in bytes, of the Length in an SMB2 READ Request (section 2.2.19) that the server will accept on the transport that established this connection. Connection.MaxReadSize MUST be set to MaxReadSize. ▪ MaxWriteSize is set to the maximum size, <230> in bytes, of the Length in an SMB2 WRITE Request (section 2.2.21) that the server will accept on the transport that established this connection. Connection.MaxWriteSize MUST be set to MaxWriteSize. <p>Changed to:</p> <ul style="list-style-type: none"> ▪ MaxTransactSize is set to the maximum buffer size, in bytes, that the server will accept on this connection for QUERY_INFO, QUERY_DIRECTORY, SET_INFO and CHANGE_NOTIFY operations. This field is applicable only for buffers sent by the client in SET_INFO requests, or returned from the server in QUERY_INFO, QUERY_DIRECTORY, and CHANGE_NOTIFY responses. This value SHOULD<229> be greater than or equal to 65536. Connection.MaxTransactSize MUST be set to MaxTransactSize. ▪ MaxReadSize is set to the maximum size, in bytes, of the Length in an SMB2 READ Request (section 2.2.19) that the server will accept on the transport that established this connection. This value SHOULD<230> be greater than or equal to 65536. Connection.MaxReadSize MUST be set to MaxReadSize. ▪ MaxWriteSize is set to the maximum size, in bytes, of the Length in an SMB2 WRITE Request (section 2.2.21) that the server will accept on the transport that established this connection. This value SHOULD<231> be greater than or equal to 65536. Connection.MaxWriteSize MUST be set to MaxWriteSize.
2015/12/11	<p>In section 3.3.5.9 Receiving an SMB2 CREATE Request, the first paragraph under Create Context Validation has been changed.</p> <p>Changed from:</p> <p>The server SHOULD<247> fail any request having a create context not specified in section 2.2.13.2 with a STATUS_INVALID_PARAMETER error.</p> <p><247> Section 3.3.5.9: Windows Vista SP1, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, and Windows Server 2012 ignore create contexts having a NameLength greater than 4 and ignore create contexts with a length of 4 that are not specified in section 2.2.13.2.</p> <p>Changed to:</p> <p>The server MUST fail create contexts having a NameLength less than 4 with a STATUS_INVALID_PARAMETER error.</p>

*Date format: YYYY/MM/DD