[MS-OXSMTP]:
Simple Mail Transfer Protocol (SMTP) Mail Submission Extensions

Revision Summary

Date	Revision History	Revision Class	Comments
04/04/2008	0.1		Initial Availability.
06/27/2008	1.0		Initial Release.
08/06/2008	1.01		Revised and edited technical content.
09/03/2008	1.02		Updated references.
12/03/2008	1.03		Updated IP notice.
04/10/2009	2.0		Updated applicable product releases.
07/15/2009	3.0	Major	Revised and edited for technical content.
11/04/2009	3.1.0	Minor	Updated the technical content.
02/10/2010	3.2.0	Minor	Updated the technical content.
05/05/2010	3.3.0	Minor	Updated the technical content.
08/04/2010	4.0	Major	Significantly changed the technical content.
11/03/2010	4.0	No change	No changes to the meaning, language, or formatting of the technical content.
03/18/2011	4.1	Minor	Clarified the meaning of the technical content.
08/05/2011	5.0	Major	Significantly changed the technical content.
10/07/2011	5.0	No change	No changes to the meaning, language, or formatting of the technical content.
01/20/2012	6.0	Major	Significantly changed the technical content.

Table of Contents

1 Introduction

The Simple Mail Transfer Protocol (SMTP) Message Submission Extensions conform to and extend SMTP standards to support the transfer of mail between a client and a server.

Sections 1.8, 2, and 3 of this specification are normative and contain RFC 2119 language. Sections 1.5 and 1.9 are also normative but cannot contain RFC 2119 language. All other sections and examples in this specification are informative.

1.1 Glossary

The following terms are defined in [MS-GLOS]:

Transmission Control Protocol (TCP)

The following terms are defined in [MS-OXGLOS]:

MIME message
Multipurpose Internet Mail Extensions (MIME)
Simple Mail Transfer Protocol (SMTP)
Transport Layer Security (TLS)

The following terms are specific to this document:

delivery status notification (DSN): A message that reports the result of an attempt to deliver a message to one or more recipients, as described in [RFC3464].

Message Submission Agent (MSA): A process that accepts messages from a Mail User Agent (MUA) and either delivers it or acts as a Simple Mail Transfer Protocol (SMTP) client to submit the messages to a Message Transfer Agent (MTA).

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as described in [RFC2119]. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 References

References to Microsoft Open Specification documents do not include a publishing year because links are to the latest version of the documents, which are updated frequently. References to other documents include a publishing year when one is available.

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information. Please check the archive site, http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624, as an additional source.

[MS-SMTP] Microsoft Corporation, "NT LAN Manager (NTLM) Authentication: Simple Mail Transfer Protocol (SMTP) Extension".

[MS-XLOGIN] Microsoft Corporation, "Simple Mail Transfer Protocol (SMTP) AUTH LOGIN Extension".

[RFC1652] Klensin, J., Freed, N., Ed., Rose, M., et al., "SMTP Service Extension for 8bit-MIMEtransport", RFC 1652, July 1994, http://www.rfc-editor.org/rfc/rfc1652.txt

[RFC1869] Klensin, J., Freed, N., Ed., Rose, M., et al., "SMTP Service Extensions", STD 10, RFC 1869, November 1995, http://www.rfc-editor.org/rfc/rfc1869.txt

[RFC1870] Klensin, J., Freed, N., Ed., and Moore, K., "SMTP Service Extension for Message Size Declaration", STD 10, RFC 1870, November 1995, http://www.rfc-editor.org/rfc/rfc1870.txt

[RFC1891] Moore, K., "SMTP Service Extension for Delivery Status Notifications", RFC 1891, January 1996, http://www.rfc-editor.org/rfc/rfc1891.txt

[RFC2034] Freed, N., "SMTP Service Extension for Returning Enhanced Error Codes", RFC 2034, October 1996, http://www.rfc-editor.org/rfc/rfc2034.txt

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, http://www.rfc-editor.org/rfc/rfc2119.txt

[RFC2460] Deering, S., and Hinden, R., "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998, http://www.ietf.org/rfc/rfc2460.txt

[RFC2920] Freed, N., "SMTP Service Extension for Command Pipelining", STD 60, RFC 2920, September 2000, http://www.rfc-editor.org/rfc/rfc2920.txt

[RFC3030] Vaudreuil, G., "SMTP Service Extensions for Transmission of Large and Binary MIME Messages", RFC 3030, December 2000, http://www.rfc-editor.org/rfc/rfc3030.txt

[RFC3207] Hoffman, P., "SMTP Service Extension for Secure SMTP over Transport Layer Security", RFC 3207, February 2002, http://www.rfc-editor.org/rfc/rfc3207.txt

[RFC4409] Gellens, R., and Klensin, J., "Message Submission for Mail", RFC 4409, April 2006, http://www.rfc-editor.org/rfc/rfc4409.txt

[RFC4954] Siemborski, R., and Melnikov, A., Eds., "SMTP Service Extension for Authentication", RFC 4954, July 2007, http://www.rfc-editor.org/rfc/rfc4954.txt

[RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, October 2008, http://www.rfc-editor.org/rfc/rfc5321.txt

[RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008, http://www.ietf.org/rfc/rfc5322.txt

[RFC791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981, http://www.ietf.org/rfc/rfc791.txt

1.2.2   Informative References

[MS-GLOS] Microsoft Corporation, "Windows Protocols Master Glossary".

[MS-OXGLOS] Microsoft Corporation, "Exchange Server Protocols Master Glossary".

1.3   Overview

The SMTP Mail Submission Extensions conform to [RFC5321] and the standard extensions that interoperate with [RFC5321]. This set of extensions enables additional features and communication between an **SMTP** client and server.

These extensions conform to the core SMTP specification and the following SMTP extensions:

- SMTP Service Extensions, as described in [RFC1869]: Enables a server to notify a client about the extensions that the server supports.

- 8bit-MIME transport, as described in [RFC1652]: Enables a server to use SMTP Service Extensions to inform a client that the server supports **Multipurpose Internet Mail Extensions (MIME)** transport, and allows the client to send 8-bit MIME messages.

- Transmission of Large and Binary MIME Messages, as described in [RFC3030]: Enables a server to use SMTP Service Extensions to inform a client that the server supports large binary **MIME messages**, and allows the client to send large binary MIME messages.

- Message Size Declaration, as described in [RFC1870]: Enables a server to use SMTP Service Extensions to inform a client that the server supports client message size declarations, and allows the client to declare a message size and the server to provide the maximum message size that it will accept.

- Delivery Status Notifications, as described in [RFC1891]: Enables a server to use SMTP Service Extensions to inform a client that the server supports delivery status notifications, and allows the client to indicate how it wants **delivery status notifications (DSNs)** to be sent to it.

- Returning Enhanced Error Codes, as described in [RFC2034]: Enables a server to use SMTP Service Extensions to inform a client that the server supports enhanced error codes.

- Authentication, as described in [RFC4954]: Enables a server to use SMTP Service Extensions to tell a client which authentication mechanisms the server supports, and allows a client to use the authentication extension. The authentication mechanisms are described in [MS-SMTP] and [MS-XLOGIN].

- Command Pipelining, as described in [RFC2920]: Enables a server to use SMTP Service Extensions to tell a client whether it implements SMTP command pipelining, and allows a client to use command pipelining.

- Secure SMTP over Transport Layer Security, as described in [RFC3207]: Enables a server to use SMTP Service Extensions to tell a client whether it implements **Transport Layer Security (TLS)**, and allows a client and server to use transport layer security.

- Message Submission for Mail, as described in [RFC4409]: Enables a server to use SMTP Service Extensions for message submission, and allows a client, specifically a message submission agent, to submit mail to a server.

1.4 Relationship to Other Protocols

The SMTP Mail Submission Extensions, which are based on [RFC5321] and other related extensions, use **Transmission Control Protocol (TCP)** version 4, as described in [RFC791], and TCP version 6, as described in [RFC2460], for message transfer.

1.5 Prerequisites/Preconditions

The SMTP Mail Submission Extensions have to be enabled in order to operate. The mechanisms for enabling and disabling those extensions are beyond the scope of this document.

The SMTP server can be configured to accept connections at zero or more bindings. A binding is a combination of an IPv4 and IPv6 address and a TCP port number.

1.6 Applicability Statement

The SMTP Mail Submission Extensions are applicable to scenarios where clients will be submitting e-mail messages directly to a server. This specification does not cover how SMTP transport agents affect or alter messages on the server.

1.7 Versioning and Capability Negotiation

The SMTP Mail Submission Extensions introduce no new versioning mechanisms beyond those that exist in SMTP, as described in [RFC5321].

Negotiation of SMTP options is described in [RFC1869] section 4. The EHLO response, as described in section 3.2.5, indicates what capabilities are present on the server.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

Parameter	Value	Reference
Server port	Port 25	Internet Assigned Numbers Authority
Server port	Port 587	Internet Assigned Numbers Authority

Port mapping is configurable so that non-default values can be used.

2 Messages

2.1 Transport

The transport of the protocol that the SMTP Mail Submission Extensions extend is specified in [RFC5321] section 1.1. Specifically, these extensions use TCP over IPv4, as specified in [RFC791], and TCP over IPv6, as specified in [RFC2460].

2.2 Message Syntax

The syntax of the messages that are exchanged between the client and the server is specified in [RFC5321]. Some optional message strings and codes that are based on the SMTP standard are implemented by these protocol extensions. These message strings and codes are specified in section 3.

2.2.1 Namespaces

None.

3 Protocol Details

3.1 Client Details

The client role conforms to [RFC5321] for the exchange of messages with the server. The client can implement the SMTP extensions that are specified by the following: [RFC1652], [RFC1869], [RFC1870], [RFC1891], [RFC2034], [RFC2920], [RFC3207], [RFC4409], [RFC4954], [RFC5322], [MS-SMTP] and [MS-XLOGIN]. If the client does not support all or part of the SMTP extensions, the server can service clients with the most basic SMTP implementations.

3.1.1 Abstract Data Model

The client state model is specified in [RFC5321]. The client state model MUST reflect the server state model. This set of protocol extensions fully complies with the state transitions specified in [RFC5321].

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Higher-Layer Triggered Events

None.

3.1.5 Message Processing Events and Sequencing Rules

The client conforms to [RFC5321] for all message processing events and sequencing rules.

3.1.6 Timer Events

None.

3.1.7 Other Local Events

None.

3.2 Server Details

The SMTP Mail Submission Extensions are compliant with the specifications listed in section 1.2.1. Extensions to the specifications listed in section 1.2.1 for the server role are specified in this section<1>.

3.2.1 Abstract Data Model

The server state model is specified in [RFC5321]. The SMTP Mail Submission Extensions are fully compliant with the state transitions that are specified in [RFC5321].

3.2.2 Timers

ConnectionTimer: A timer that identifies how much time has elapsed since a session was initiated. If the **ConnectionTimer** lapses, the server MUST transition to the initial state, which is the equivalent of clearing the state tables and buffers, as specified in [RFC5321] section 2.3.6.

ConnectionInactivityTimer: A timer that identifies how much time has elapsed since a client provided input. If the **ConnectionInactivityTimer** lapses, the server MUST transition to the initial state, which is the equivalent of clearing the state tables and buffers, as specified in [RFC5321] section 2.3.6.

3.2.3 Initialization

The **ConnectionTimer** timer, as specified in section 3.2.2, is initialized when a session is initiated.

The **ConnectionInactivityTimer** timer, as specified in section 3.2.2, is initialized when the client and server start a session. The **ConnectionInactivityTimer** timer is restarted when the client sends input to the server.

3.2.4 Higher-Layer Triggered Events

None.

3.2.5 Message Processing Events and Sequencing Rules

The server role is compliant with the message processing and sequencing rules that are specified in [RFC5321].

The server is fully compliant with [RFC1869] section 4 for the **EHLO** command. The server MUST use the following **EHLO** extension responses:

As specified in [RFC5321]:

ServerResponse = "250-" dot-atom-text SP "Hello" SP IPv4address/IPv6address CR LF

dot-atom-text = 1*atext *("." 1*atext)

atext = ALPHA / DIGIT / ; Printable US-ASCII

"!" / "#" /; characters not including

"$" / "%" /; specials. Used for atoms.

"&" / "'" /

"*" / "+" /

"-" / "/" /

"=" / "?" /

"^" / "_" /

"`" / "{" /

"|" / "}" /

"~"

IPv4address = d8 "." d8 "." d8 "." d8

d8 = DIGIT ; 0-9

/ %x31-39 DIGIT ; 10-99

/ "1" 2DIGIT ; 100-199

/ "2" %x30-34 DIGIT ; 200-249

/ "25" %x30-35 ; 250-255

IPv6address = 6(h16 ":") ls32

/ "::" 5(h16 ":") ls32

/ [h16] "::" 4(h16 ":") ls32

/ [*1(h16 ":") h16] "::" 3(h16 ":") ls32

/ [*2(h16 ":") h16] "::" 2(h16 ":") ls32

/ [*3(h16 ":") h16] "::" h16 ":" ls32

/ [*4(h16 ":") h16] "::" ls32

/ [*5(h16 ":") h16] "::" h16

/ [*6(h16 ":") h16] "::"

ls32 = h16 ":" h16 / IPv4address

h16 = 1*4HEXDIG

As specified in [RFC1870]:

Size = "250-SIZE" CR LF

As specified in [RFC1891]:

dsn = "250-DSN" CR LF

As specified in [RFC2034]:

enhancedStatusCodes = "250-ENHANCEDSTATUSCODES" CR LF

The server role by default MUST return the following codes in response to an **EHLO** command. The following codes can optionally be returned if the server changes the default configuration:

As specified in [RFC2920]:

pipelining = "250-PIPELINING" CR LF

As specified in [RFC3207]:

starttls = "250-STARTTLS" CR LF

The framework for authentication is specified in [RFC4954]. The specific responses that correspond to authentication schemes are specified in [MS-SMTP] and [MS-XLOGIN]. The server MUST be compliant with [MS-SMTP] and [MS-XLOGIN] for authentication. The server code is as follows:

Release: Sunday, January 22, 2012

auth = "250-AUTH" SP "NTLM" SP "LOGIN" CR LF

As specified in [RFC1652]:

8bitmime = "250-8BITMIME" CR LF

As specified in [RFC3030]:

binarymime = "250-BINARYMIME" CR LF

As specified in [RFC3030]:

chunking = "250-CHUNKING" CR LF

The following table lists the error code extensions to [RFC4409] section 4.1 that MUST be returned if the error condition is met.

Command	Error condition	Response code	Extended status code	Response text
MAIL	No initial **EHLO** command issued.	503	5.5.2	Send hello first
MAIL	**MAIL** command issued after **BDAT** command.	503	5.5.1	Bad sequence of commands
MAIL	A **STARTTLS** command must be issued first.	451	5.7.3	Must issue a **STARTTLS** command first
MAIL	A second **MAIL From** field is specified.	503	5.5.2	Sender already specified
MAIL	The **From** field is specified without a ":" character.	501	5.5.4	Unrecognized parameter
MAIL	System resources are not available.	452	4.3.1	Insufficient system resources
MAIL	Invalid extension.	501	5.5.4	Invalid arguments
RCPT	No initial **EHLO** command issued.	503	5.5.2	Send hello first
RCPT	**RCPT** command issued after **BDAT** command.	503	5.5.1	Bad sequence of commands
RCPT	Command format is missing a ":" character.	501	5.5.4	Unrecognized parameter
RCPT	Invalid address.	501	5.1.3	Invalid address
RCPT	The maximum number of recipients has been exceeded.	452	4.5.3	Too many recipients
RCPT	A null reverse path is given as the address.	501	5.1.3	Invalid address
RCPT	A relay is not available for	550	5.7.1	Unable to relay

Command	Error condition	Response code	Extended status code	Response text
	the recipient.			

The following table lists the error code extensions to [RFC4409] section 4.2 that MUST be returned if the error condition is met.

Command	Error condition	Response code	Extended status code	Response text
MAIL	Invalid address.	501	5.1.7	Invalid address

The following table lists the error code extensions to [RFC4409] section 4.3 that MUST be returned if the error condition is met.

Command	Error condition	Response code	Extended status code	Response text
MAIL	Session was not authenticated by using the SMTP **AUTH** command.	530	5.7.1	Not authenticated
MAIL	Session was not authenticated by using the SMTP **AUTH** command.	530	5.7.1	Client was not authenticated

The following table lists the error code extensions to [RFC4409] sections 3.2, 4.2, and 5.1 that SHOULD be returned if the error condition is met.

Command	Error condition	Response code	Extended status code	Response text
MAIL	If the **Message Submission Agent (MSA)** is not able to determine a return path to the submitting user, from a valid **MAIL FROM** command, a valid source IP address, or based on authenticated identity, then the MSA SHOULD immediately reject the message.	550	5.7.1	Client does not have permissions to submit to this server
MAIL or **RCPT**	If domain expansion fails.	550	5.7.1	Unable to relay
MAIL or **RCPT**	Improper address.	501	5.1.3	Invalid address

The following table contains error code extensions to [RFC4409] sections 6.1 and 6.2 that MAY be returned if the error condition is met.

Command	Error condition	Response code	Extended status code	Response text
MAIL	The address in the **FROM** field has insufficient submission rights.	550	5.7.1	Client does not have permissions to submit to this server
RCPT	The recipient does not have	550	5.7.1	Unable to relay

Command	Error condition	Response code	Extended status code	Response text
	sufficient privileges.			

3.2.6 Timer Events

ConnectionTimeOut timer event: Occurs when the **ConnectionTimer**, as specified in section 3.2.2, exceeds the connection timer limit. The connection timer limit MUST be 5 minutes, as specified in [RFC5321] section 4.5.3.2.7, for a gateway server. The connection timer limit MUST be 10 minutes for a relay server. The SMTP response that is sent after this event occurs MUST be "421 4.4.1 Connection timed out". The server MUST end the session as specified in [RFC5321] section 3.8.

ConnectionInactivityTimeOut timer event: Occurs when the **ConnectionInactivityTimer**, as specified in section 3.2.2, exceeds the connection inactivity timer limit. The connection inactivity timer limit MUST be 1 minute for a gateway server. The connection inactivity timer limit MUST be 5 minutes for a relay server. The SMTP response that is sent after this event occurs MUST be "451 4.7.0 Timeout waiting for client input". The server MUST end the session as specified in [RFC5321] section 3.8.

3.2.7 Other Local Events

MaxHopCount event: Occurs when the maximum hop count is exceeded. The default value for the edge and hub SMTP server maximum hop count MUST be 60. The SMTP response that is sent after this event occurs MUST be "554 5.4.6 Hop count exceeded – possible mail loop". This response MUST be sent when the **Received:** header fields, as specified in [RFC5321] section 6.3, length is more than the configured maximum hop count. This response is sent at the end of a **DATA** or **BDAT** command. The server MUST end the session.

MaxLocalHopCount event: Occurs when the maximum local hop count is exceeded. The default value for the edge or hub SMTP server maximum local hop count MUST be 12.The SMTP response that is sent after this event occurs MUST be "554 5.4.6 Hop count exceeded – possible mail loop". This response MUST be sent when the server has received the message more than the configured maximum local hop count. This response is sent at the end of a **DATA** or **BDAT** command. The server MUST end the session.

TooManyRecipients event: Occurs when the maximum recipients count is exceeded on a message. The default value for the edge or hub SMTP server maximum recipients count MUST be 200. The SMTP response that is sent after this event occurs MUST be "452 4.5.3 Too many recipients". This response MUST be sent at the end of a **RCPT TO** command. The server MUST end the session.

MessageRateLimitExceeded event: Occurs when the message submission rate for a client has exceeded the configured limit. The default value for an edge SMTP server MUST be 600 messages per minute and MUST be based on a unique IP address. The default value for a hub SMTP server MUST be 5 messages per minute and MUST be based on a unique user. The SMTP response sent after this event occurs MUST be "421 4.4.2 Message submission rate for this client has exceeded the configured limit". This response MUST be sent at the end of a **MAIL FROM** command. The server MUST end the session.

HeaderSizeExceeded event: Occurs when the message header size exceeds 64 KB. The SMTP response that is sent after this event occurs MUST be "552 5.3.4 Header size exceeds fixed maximum size". This response MUST be sent at the end of a **DATA** or **BDAT** command. The server MUST end the session.

MessageSizeExceeded event: Occurs when the message size exceeds 10 MB. The SMTP response that is sent after this event occurs MUST be "552 5.3.4 Message size exceeds fixed maximum message size". This response MUST be sent at the end of a **DATA** or **BDAT** command. The server MUST end the session.

ProtocolViolationCount event: Occurs when the maximum number of logon or protocol errors is exceeded. The value for the maximum logon error count MUST be 3. The value for the maximum protocol error count MUST be 5. The SMTP response that is sent after this event occurs MUST be "421 4.7.0 Too many errors on this connection, closing transmission channel". The server MUST end the session.

OutOfResources event: Occurs when a client initiates a TCP connection to the server and the server is low on memory or disk space. The SMTP response sent after this event occurs MUST be "452 4.3.1 Insufficient system resources". The server MUST end the session.

NewConnectionNotAvailable event: Occurs when an SMTP server cannot process a new connection. It indicates that the process is in a hanging or crashed condition. The SMTP response that is sent after this event occurs MUST be "421 4.4.2 Connection dropped". The server MUST end the session.

BindingNotConfigured event: Occurs when an SMTP server is not configured to accept connections from a client at a specific IP address or user. The SMTP response that is sent after this event occurs SHOULD be "421 4.3.2 Service not available"<2>. The server MUST end the session.

ConnectionCountExceeded event: Occurs when an SMTP server has exceeded 5000 concurrent inbound connections. The SMTP response that is sent after this event occurs MUST be "421 4.3.2 The maximum number of concurrent server connections has exceeded a limit, closing transmission channel". The server MUST end the session.

ConnectionCountPerSource event: Occurs when an SMTP server has exceeded 20 inbound connections for an IP address. The SMTP response that is sent after this event occurs MUST be "421 4.3.2 The maximum number of concurrent connections has exceeded a limit, closing transmission channel". The server MUST end the session.

IPAddressNotAllowed event: Occurs when a gateway SMTP server binding receives a connection from an IP address that is not allowed. For a static range of blocked IP addresses, the SMTP response that is sent after this event occurs MUST be "550 5.7.1 External client with IP address {0} does not have permissions to submit to this server. Visit http://support.microsoft.com/kb/928123 for more information". For a dynamic range of blocked IP addresses, the SMTP response MUST be "550 5.7.1 External client with IP address {0} does not have permissions to submit to this server". The {0} input is the IP address that is not allowed. If the client IP address is found on a blocked list of IP addresses, the SMTP server response sent after this event MUST be "550 5.7.1 Recipient not authorized, your IP has been found on a block list". The server MUST end the session.

AcknowledgementDelay event: Occurs when the server waits 30 seconds for a mail item to be delivered to the next hop. This event occurs after the end of **DATA** or **BDAT LAST** command, as specified in [RFC3030] section 2. If the **AcknowledgementDelay** event occurs, the server MUST send acknowledgment of receiving the mail item even if transport has not delivered the item to the next hop. The server sends the response as specified in [RFC5321] and processes the next command. The server state does not change.

Tarpit event: Occurs at the end of a command when the server sends an error message to an unauthenticated user, and once again if the same client connects to the server. The server MUST ignore connection attempts for 5 seconds and then send the response to the client. The server sends the response as specified in [RFC5321] and processes the next command. The server state does not change.

4 Protocol Examples

None.

5 Security

5.1 Security Considerations for Implementers

The specifications listed in section 1.2.1 describe security considerations for implementers.

5.2 Index of Security Parameters

Security parameters for message submission authentication are described in [RFC4409].

6   Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Microsoft® Exchange Server 2003

- Microsoft® Exchange Server 2007

- Microsoft® Exchange Server 2010

- Microsoft® Exchange Server 15 Technical Preview

- Microsoft® Office Outlook® 2003

- Microsoft® Office Outlook® 2007

- Microsoft® Outlook® 2010

- Microsoft® Outlook® 15 Technical Preview

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

<1> Section 3.2: Exchange 2003, Exchange 2007, and Exchange 2010 do not support the obsolete features specified in [RFC5321], section F.

<2> Section 3.2.7: Exchange 2007 returns "421 4.3.2 Service not available, closing transmission channel" for the SMTP response to **BindingNotConfigured event**.

7   Change Tracking

This section identifies changes that were made to the [MS-OXSMTP] protocol document between the October 2011 and January 2012 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

The revision class **New** means that a new document is being released.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.

- An extensive rewrite, addition, or deletion of major portions of content.

- The removal of a document from the documentation set.

- Changes made for template compliance.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **Editorial** means that the language and formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical or language changes were introduced. The technical content of the document is identical to the last released version, but minor editorial and formatting changes, as well as updates to the header and footer information, and to the revision summary, may have been made.

Major and minor changes can be described further using the following change types:

- New content added.

- Content updated.

- Content removed.

- New product behavior note added.

- Product behavior note updated.

- Product behavior note removed.

- New protocol syntax added.

- Protocol syntax updated.

- Protocol syntax removed.

- New content added due to protocol revision.

- Content updated due to protocol revision.

- Content removed due to protocol revision.

- New protocol syntax added due to protocol revision.

- Protocol syntax updated due to protocol revision.

- Protocol syntax removed due to protocol revision.

- New content added for template compliance.

- Content updated for template compliance.

- Content removed for template compliance.

- Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated.**

Some important terms used in the change type descriptions are defined as follows:

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.

- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

The changes made to this document are listed in the following table. For more information, please contact protocol@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
6 Appendix A: Product Behavior	Added Exchange 15 Technical Preview and Outlook 15 Technical Preview to the list of applicable product versions.	Y	Content updated.

8 Index

A

Abstract data model
   client 9
   server 9
Applicability 7

C

Capability negotiation 7
Change tracking 19
Client
   abstract data model 9
   higher-layer triggered events 9
   initialization 9
   message processing 9
   other local events 9
   overview 9
   sequencing rules 9
   timer events 9
   timers 9

D

Data model - abstract
   client 9
   server 9

F

Fields - vendor-extensible 7

G

Glossary 4

H

Higher-layer triggered events
   client 9
   server 10

I

Implementer - security considerations 17
Index of security parameters 17
Informative references 5
Initialization
   client 9
   server 10
Introduction 4

M

Message processing
   client 9
   server 10
Messages

Namespaces 8
   transport 8

N

Namespaces message 8
Normative references 4

O

Other local events
   client 9
   server 14
Overview (synopsis) 5

P

Parameters - security index 17
Preconditions 6
Prerequisites 6
Product behavior 18

R

References
   informative 5
   normative 4
Relationship to other protocols 6

S

Security
   implementer considerations 17
   parameter index 17
Sequencing rules
   client 9
   server 10
Server
   abstract data model 9
   higher-layer triggered events 10
   initialization 10
   message processing 10
   other local events 14
   overview 9
   sequencing rules 10
   timer events 14
   timers 10
Standards assignments 7

T

Timer events
   client 9
   server 14
Timers
   client 9
   server 10
Tracking changes 19

V