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

---

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft's Open Specification Promise (available here: http://www.microsoft.com/interop/osp) or the Community Promise (available here: http://www.microsoft.com/interop/cp/default.mspx). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.
- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

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.

Table of Contents

1 Introduction 4
1.1 Glossary 4
1.2 References 4
1.2.1 Normative References 4
1.2.2 Informative References 5
1.3 Overview 5
1.4 Relationship to Other Protocols 5
1.5 Prerequisites/Preconditions 5
1.6 Applicability Statement 5
1.7 Versioning and Capability Negotiation 5
1.8 Vendor-Extensible Fields 5
1.9 Standards Assignments 5

2 Messages 6
2.1 Transport 6
2.2 Message Syntax 6

3 Protocol Details 7
3.1 Client Details 7
3.1.1 Abstract Data Model 8
3.1.2 Timers 8
3.1.3 Initialization 8
3.1.4 Higher-Layer Triggered Events 8
3.1.5 Message Processing Events and Sequencing Rules 8
3.1.6 Timer Events 8
3.1.7 Other Local Events 8
3.2 Server Details 8
3.2.1 Abstract Data Model 11
3.2.2 Timers 11
3.2.3 Initialization 11
3.2.4 Higher-Layer Triggered Events 11
3.2.5 Message Processing Events and Sequencing Rules 11
3.2.6 Timer Events 11
3.2.7 Other Local Events 11

4 Protocol Examples 12

5 Security 13
5.1 Security Considerations for Implementers 13
5.2 Index of Security Parameters 13

6 Appendix A: Product Behavior 14

7 Change Tracking 15

8 Index 17

1 Introduction

The **Simple Mail Transport Protocol (SMTP)** Message Submission for Mail protocol, as specified in [RFC4409], profiles SMTP mechanisms specified in [RFC2821] and others to provide mail submission mechanisms for client mail systems. SMTP was originally defined to provide for mail transfer between mail servers. It is now widely used as a message submission mechanism, where client messaging systems introduce new messages into the mail routing network.

This specification profiles [RFC4409], identifying the elements necessary to conform to Exchange Server protocols.

1.1 Glossary

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

conditions
Mail User Agent (MUA)
NTLM
port
Simple Mail Transfer Protocol (SMTP)

The following terms are specific to this document:

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)**.

Message Transfer Agent (MTA): An **SMTP** server that accepts mail from a **Mail Submission Agent (MSA)** or another **MTA** and delivers the mail or relays it to another **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

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-OXGLOS] Microsoft Corporation, "Exchange Server Protocols Master Glossary", April 2008.

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

[RFC2554] Myers, J., "SMTP Service Extension for Authentication", RFC 2554, March 1999, http://www.ietf.org/rfc/rfc2554.txt

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

1.2.2 Informative References

[RFC2821] Klensin, J., Ed., "Simple Mail Transfer Protocol", RFC 2821, April 2001, http://www.ietf.org/rfc/rfc2821.txt

[RFC821] Postel, J., "SIMPLE MAIL TRANSFER PROTOCOL", RFC 821, August 1982, http://www.ietf.org/rfc/rfc821.txt

1.3 Overview

[RFC4409] describes a profile of SMTP[RFC2821] and others that defines how clients submit mail to a server. This document specifies which parts of [RFC4409] are necessary for message submission.

1.4 Relationship to Other Protocols

This specification profiles [RFC4409] to define the protocol for message submission. [RFC4409] is based on SMTP as specified in [RFC2821].

1.5 Prerequisites/Preconditions

None.

1.6 Applicability Statement

This protocol is applicable to scenarios where clients will be submitting e-mail messages directly to a server.

1.7 Versioning and Capability Negotiation

This specification introduces no new versioning mechanisms beyond those that exist in SMTP.

Negotiation of SMTP options is part of [RFC4409]. The EHLO response indicates what capabilities are present on the server.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

None.

2.2 Message Syntax

None.

3 Protocol Details

This protocol describes the process for a client submitting e-mail messages to a **Message Transfer Agent (MTA)**. In [RFC4409], three components are described to represent the interaction: a **Mail User Agent (MUA)**, a **Message Submission Agent (MSA)**, and an MTA. These components describe an architecture, shown in Figure 1, where the MUA creates the message, the MSA picks it up and, via SMTP, submits it to the MTA for transmission to the recipient.

Figure 1: Message transmission

Most modern e-mail clients combine the MUA and the MSA, such that there is no perceptible handoff between the MUA and MSA functions. As a result, the MSA provides SMTP client capabilities while the MTA provides the SMTP server capabilities. This protocol only covers the path between the MSA and the MTA. Thus the profile of [RFC4409] provided is broken into two roles — the client and the server — where the client performs SMTP submissions and the server accepts the SMTP submissions. Aspects of [RFC4409] that describe the interaction between the MUA and the MSA are specifically excluded from this protocol.

3.1 Client Details

The MUA/MSA process is the client. The MUA functionality is used to acquire or create the message and the MSA functionality performs an SMTP client submission. The client, therefore, only acts as an SMTP client; it does not have the ability to receive **messages** via SMTP as an SMTP server.

The client protocol for message submission conforms to [RFC4409] except as identified in the following table.

[RFC4409] section	Description
3.1	The client SHOULD submit messages on port 587, port 25 or any port that a site chooses for message submission.
3.2	This section is not included in the client profile because the MUA and MSA functionality are integrated into the same process.
3.3	The client SHOULD authenticate using mechanisms defined in [RFC2554]. The client SHOULD use AUTH LOGIN or AUTH **NTLM**.
4.1	This section is not included in the client profile because the MUA and MSA functionality are integrated into the same process.
4.2	This section is not included in the client profile because the MUA and MSA functionality are integrated into the same process.
4.3	This section is not included in the client profile because the MUA and MSA functionality are integrated into the same process.
5.1	This section is not included in the client profile because the MUA and MSA functionality are integrated into the same process.

[RFC4409] section	Description
5.2	This section is not included in the client profile because the MUA and MSA functionality are integrated into the same process.
6	This section is not included in the client profile because the MUA and MSA functionality are integrated into the same process.
7	The client SHOULD support DSN, AUTH, and STARTTLS. AUTH SHOULD be supported as specified in section 3.3 of this table.
8	This section is not included in the client profile because the MUA and MSA functionality are integrated into the same process.

3.1.1 Abstract Data Model

None.

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

None.

3.1.6 Timer Events

None.

3.1.7 Other Local Events

None.

3.2 Server Details

The server component of message submission conforms to [RFC4409] except as identified in the following table.

[RFC4409] Section	Description
3.1	The server SHOULD allow messages to be submitted on **port** 587, port 25, or any port that a site chooses for message submission.
3.2	The server SHOULD offer authenticate mechanisms as defined in [RFC2554].

[RFC4409] Section	Description
3.3	The server SHOULD offer authenticate mechanisms as defined in [RFC2554]. The client SHOULD offer AUTH LOGIN or AUTH NTLM.
4.1	The server SHOULD respond with a 554 response code for the MAIL, RCPT, or DATA commands when they contain improper information, unless a more specific response code is provided.
4.2	The server SHOULD respond with a 554 response code for the MAIL, RCPT, or DATA commands when they contain improper domain references, unless a more specific response code is provided.
4.3	The server SHOULD require client authentication. If the server does not require client authentication, the server MUST NOT issue an error response to an unauthenticated MAIL command.
7	The server SHOULD implement DSN, AUTH, and STARTTLS. The server SHOULD offer AUTH LOGIN and AUTH NTLM.

Error Conditions

The following table contains 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	RCPT command issued after XECH50 command.	503	5.5.1	Bad sequence of commands
RCPT	Command format is missing	501	5.5.4	Unrecognized

Command	Error condition	Response code	Extended Status Code	Response text
	a ":" character.			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 the recipient.	550	5.7.1	Unable to relay

The following table contains 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
MAIL	Invalid address.	501	5.1.7	Invalid address

The following table contains 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 using SMTP AUTH.	530	5.7.1	Not authenticated
MAIL	Session was not authenticated using SMTP AUTH.	530	5.7.1	Client was not authenticated

The following table contains 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	MSA is not able to determine a return path to the submitting user, from a valid MAIL FROM, 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, RCPT, or DATA	If domain expansion fails.	501	5.5.4	Invalid arguments
MAIL or RCPT	Improper address.	501	5.1.7	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 sufficient privileges.	550	5.7.1	Unable to relay

3.2.1 Abstract Data Model

None.

3.2.2 Timers

None.

3.2.3 Initialization

None.

3.2.4 Higher-Layer Triggered Events

None.

3.2.5 Message Processing Events and Sequencing Rules

None.

3.2.6 Timer Events

None.

3.2.7 Other Local Events

None.

4 Protocol Examples

None.

5 Security

5.1 Security Considerations for Implementers

None.

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 product versions. References to product versions include released service packs.

- Microsoft® Office Outlook® 2003
- Microsoft® Exchange Server 2003
- Microsoft® Office Outlook® 2007
- Microsoft® Exchange Server 2007
- Microsoft® Outlook® 2010
- Microsoft® Exchange Server 2010

Exceptions, if any, are noted below. If a service pack number appears with the product version, behavior changed in that service pack. The new behavior also applies to subsequent service packs of the product unless otherwise specified.

Unless otherwise specified, any statement of optional behavior in this specification 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 product does not follow the prescription.

7 Change Tracking

This section identifies changes made to [MS-OXSMTP] protocol documentation between February 2010 and May 2010 releases. Changes are classed as major, minor, or editorial.

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.
- A protocol is deprecated.
- The removal of a document from the documentation set.
- Changes made for template compliance.

Minor changes do not affect protocol interoperability or implementation. Examples are updates to fix technical accuracy or ambiguity at the sentence, paragraph, or table level.

Editorial changes apply to grammatical, formatting, and style issues.

No changes means that the document is identical to its last release.

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

- New content added.
- Content update.
- 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 always have the revision type "Editorially updated."

Some important terms used in revision 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.

Changes are listed in the following table. If you need further information, please contact protocol@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Revision Type
1.2.1 Normative References	55085 Added [RFC2554] reference information.	N	Content update.
1.3 Overview	Updated the section title.	N	Content updated for template compliance.
3.1 Client Details	Added reference to [RFC2554] in the table header.	N	Content update.
3.2 Server Details	55087 Updated the descriptions for sections 4.1 and 4.2 of [RFC4409], and added error conditions information.	N	Content update.

8 Index

A

Applicability 5

C

Capability negotiation 5
Change tracking 15
Client
  overview 7

E

Examples
  overview 12

F

Fields – vendor-extensible 5

G

Glossary 4

I

Implementer – security considerations 13
Index of security parameters 13
Informative references 5
Introduction 4

M

Messages
  overview 6
Messaging
  transport 6

N

Normative references 4

O

Overview 5

P

Parameters – security index 13
Preconditions 5
Prerequisites 5
Product behavior 14

R

References
  informative 5
  normative 4
Relationship to other protocols 5

S

Security
  implementer considerations 13
  overview 13
  parameter index 13
Server
  overview 8
Standards Assignments 5

T

Tracking changes 15
Transport 6

V

Vendor-extensible fields 5
Versioning 5