[MS-XLOGIN]:
Simple Mail Transfer Protocol (SMTP) AUTH LOGIN Extension

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 Open Specification Promise or the Community Promise. 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.

Preliminary

Copyright © 2012 Microsoft Corporation.

Preliminary Documentation. This Open Specification provides documentation for past and current releases and/or for the pre-release (beta) version of this technology. This Open Specification is final documentation for past or current releases as specifically noted in the document, as applicable; it is preliminary documentation for the pre-release (beta) versions. Microsoft will release final documentation in connection with the commercial release of the updated or new version of this technology. As the documentation may change between this preliminary version and the final version of this technology, there are risks in relying on preliminary documentation. To the extent that you incur additional development obligations or any other costs as a result of relying on this preliminary documentation, you do so at your own risk.

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	4.0.0	Major	Updated and revised the technical content.
02/10/2010	4.1.0	Minor	Updated the technical content.
05/05/2010	4.1.1	Editorial	Revised and edited the technical content.
08/04/2010	5.0	Major	Significantly changed the technical content.
11/03/2010	5.0	No change	No changes to the meaning, language, or formatting of the technical content.
03/18/2011	5.0	No change	No changes to the meaning, language, or formatting of the technical content.
08/05/2011	5.1	Minor	Clarified the meaning of the technical content.
10/07/2011	5.1	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.

Preliminary

Copyright © 2012 Microsoft Corporation.

Table of Contents

1 Introduction 5
1.1 Glossary 5
1.2 References 5
1.2.1 Normative References 5
1.2.2 Informative References 6
1.3 Overview 6
1.4 Relationship to Other Protocols 6
1.5 Prerequisites/Preconditions 6
1.6 Applicability Statement 6
1.7 Versioning and Capability Negotiation 6
1.8 Vendor-Extensible Fields 6
1.9 Standards Assignments 7

2 Messages 8
2.1 Transport 8
2.2 Message Syntax 8
2.2.1 EHLO Response 8
2.2.2 Command and Response ABNF Grammar 8

3 Protocol Details 9
3.1 Client Details 9
3.1.1 Abstract Data Model 9
3.1.2 Timers 11
3.1.3 Initialization 11
3.1.4 Higher-Layer Triggered Events 11
3.1.5 Message Processing Events and Sequencing Rules 11
3.1.5.1 Sending EHLO 11
3.1.5.2 Requesting Simple Login Authentication 11
3.1.5.3 Requesting Login Authentication with Username 12
3.1.6 Timer Events 12
3.1.7 Other Local Events 12
3.2 Server Details 12
3.2.1 Abstract Data Model 12
3.2.2 Timers 15
3.2.3 Initialization 15
3.2.4 Higher-Layer Triggered Events 15
3.2.5 Message Processing Events and Sequencing Rules 15
3.2.5.1 Processing EHLO 15
3.2.5.2 Processing AUTH LOGIN 15
3.2.5.3 Processing Username Response 15
3.2.5.4 Processing Password Response 15
3.2.6 Timer Events 16
3.2.7 Other Local Events 16

4 Protocol Example 17

5 Security 18
5.1 Security Considerations for Implementers 18
5.2 Index of Security Parameters 18

6 Appendix A: Product Behavior 19

Preliminary

Copyright © 2012 Microsoft Corporation.

7 Change Tracking.....20

8 Index 22

Preliminary

Copyright © 2012 Microsoft Corporation.

1 Introduction

The Simple Mail Transfer Protocol (SMTP) AUTH LOGIN Extension is an authentication mechanism that provides an easily implemented method for clients to authenticate to **SMTP** servers over a standard SMTP connection. This extension uses the SMTP Service Extension for Authentication, as described in [RFC4954], to extend SMTP.

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]:

Augmented Backus-Naur Form (ABNF)
NT LAN Manager (NTLM) Authentication Protocol
Secure Sockets Layer (SSL)

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

base64 encoding
Simple Mail Transfer Protocol (SMTP)
Transport Layer Security (TLS)

The following terms are specific to this document:

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.

[RFC2045] Freed, N., and Borenstein, N., "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, November 1996, http://ietf.org/rfc/rfc2045.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

[RFC4346] Dierks, T., and Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.1", RFC 4346, April 2006, http://www.ietf.org/rfc/rfc4346.txt

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

Preliminary

Copyright © 2012 Microsoft Corporation.

[RFC5234] Crocker, D., Ed., and Overell, P., "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008, http://www.rfc-editor.org/rfc/rfc5234.txt

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

Client applications use SMTP to transfer mail to a server for submission. Client applications that connect to an SMTP server can use a number of different authentication mechanisms. In some scenarios, clients can use existing authentication mechanisms to authenticate with the SMTP server, such as the **NT LAN Manager (NTLM) Authentication Protocol**. However, in other scenarios, existing authentication mechanisms are unavailable or clients may not implement them. This extension provides an authentication mechanism for SMTP clients that is simple to implement.

The SMTP Service Extension for Authentication, as described in [RFC4954], defines a service extension to SMTP, as described in [RFC5321], where a client specifies an authentication method to the server and performs an authentication protocol exchange. This extension is one such authentication method for SMTP. It allows clients to authenticate to SMTP servers over a standard SMTP connection by passing authentication information in SMTP commands and responses.

1.4 Relationship to Other Protocols

This extension uses the methods provided by the SMTP Service for Authentication, as described in [RFC4954], to extend SMTP, as described in [RFC5321], by providing a new authentication method. This extension relies on SMTP to provide the transport for the authentication commands and responses.

1.5 Prerequisites/Preconditions

This extension conforms to all of the prerequisites and preconditions of SMTP, as described in [RFC5321], and the extension to SMTP provided by the SMTP Service for Authentication, as described in [RFC4954].

1.6 Applicability Statement

This extension is used by clients to support authentication to SMTP servers that implement the AUTH LOGIN extension. This extension is used by SMTP servers to provide an authentication method to control access to the SMTP service.

1.7 Versioning and Capability Negotiation

None.

1.8 Vendor-Extensible Fields

None.

Preliminary

Copyright © 2012 Microsoft Corporation.

1.9 Standards Assignments

None.

Preliminary

Copyright © 2012 Microsoft Corporation.

2 Messages

2.1 Transport

This extension does not change the base transport specified by [RFC5321], or its extension specified by [RFC4954].

2.2 Message Syntax

2.2.1 EHLO Response

Server support for the AUTH LOGIN extension is identified by the AUTH EHLO keyword in the EHLO response, as specified in [RFC4954] section 3. In the following example, "C:" and "S:" indicate lines sent by the client and server respectively.

```
S: 220 SMTP.example.com
C: EHLO client.example.com
S: 250-smtp.example.com  Hello client.example.com
S: 250 AUTH LOGIN
```

The server response of AUTH LOGIN as part of the responses to EHLO indicates support for AUTH LOGIN.

2.2.2 Command and Response ABNF Grammar

This section uses **Augmented Backus-Naur Form (ABNF)** (as specified in [RFC5234]) to define the format of commands and responses used by this extension. Note that the values of *username* and *password* are encoded using **base64 encoding**, as specified in [RFC2045], before being transmitted.

```
CR                              = %x0D
LF                              = %x0A
SP                              = %x20

username                        = 1*CHAR                        ; Base64-encoded username
password                        = 1*CHAR                        ; Base64-encoded password

auth_login_command              = "AUTH LOGIN" CR LF
auth_login_command_user         = "AUTH LOGIN" SP username CR LF
auth_login_username_challenge   = "334 VXNlcm5hbWU6" CR LF
auth_login_username_response    = username CR LF
auth_login_password_challenge   = "334 UGFzc3dvcmQ6" CR LF
auth_login_password_response    = password CR LF
```

Preliminary

Copyright © 2012 Microsoft Corporation.

3 Protocol Details

3.1 Client Details

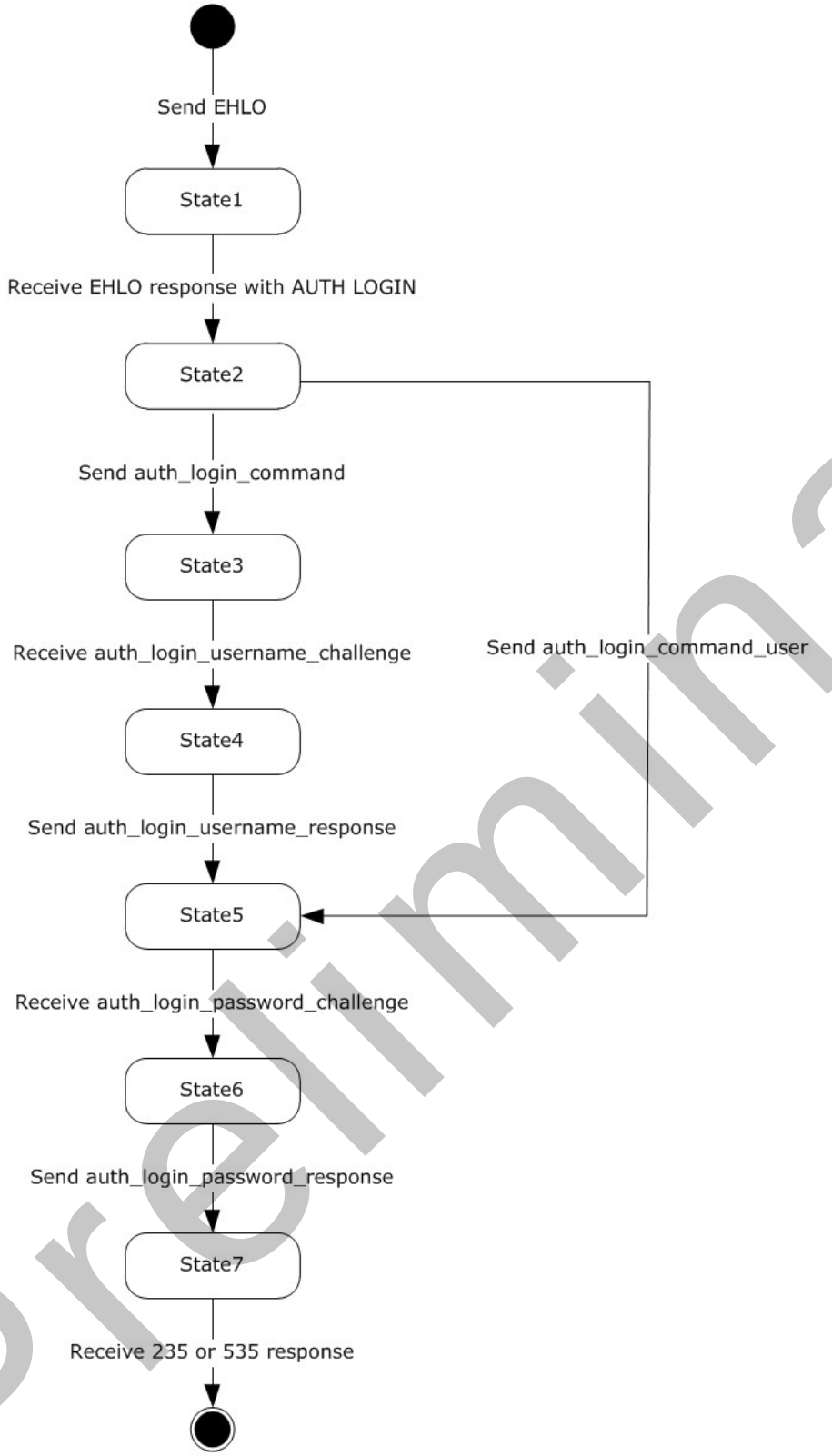
3.1.1 Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. The described organization is provided to facilitate the explanation of how the protocol behaves. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that described in this document.

The abstract data model shown below represents the sequence of messages relative to the client.

Preliminary

Copyright © 2012 Microsoft Corporation.



Preliminary

Copyright © 2012 Microsoft Corporation.

Figure 1: Client abstract data model

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

This extension does not change the message processing events or sequencing rules of messages specified in [RFC4954]. Clients SHOULD ignore any enhanced status codes returned by the server.

3.1.5.1 Sending EHLO

Before requesting authentication, the client SHOULD send an EHLO command and examine the response. If the response does not contain the AUTH LOGIN keyword specified in section 2.2.1, the client SHOULD NOT attempt to use LOGIN authentication.

3.1.5.2 Requesting Simple Login Authentication

To request LOGIN authentication, the client issues the AUTH command with the parameter LOGIN (*auth_login_command* as specified in section 2.2.2).

```
AUTH LOGIN<CR><LF>
```

If AUTH LOGIN is not supported, the server responds with a 504 error message as specified in [RFC4954] section 4.

If AUTH LOGIN is supported on the server, the server responds with *auth_login_username_challenge* as specified in section 2.2.2.

```
334 VXNlcm5hbWU6<CR><LF>
```

The client then responds with *auth_login_username_response* (as specified in section 2.2.2) with the user name to be used for authentication, encoded with base64 encoding, as specified in [RFC2045]. For example, if the client's user name is "Charlie", the client responds with the following:

```
Q2hhcmxpZQ==<CR><LF>
```

The server then responds with *auth_login_password_challenge* (as specified in section 2.2.2).

```
334 UGFzc3dvcmQ6<CR><LF>
```

Preliminary

Copyright © 2012 Microsoft Corporation.

The client then responds with *auth_login_password_response* (as specified in section 2.2.2) with the password to be used for authentication, encoded with base64 encoding. For example, if the client's password is "password", the client responds with the following:

```
cGFzc3dvcmQ=<CR><LF>
```

If the authentication is complete, the server issues a 235 reply for success or a 535 reply for failure, as specified in [RFC4954].

3.1.5.3 Requesting Login Authentication with Username

To request LOGIN authentication, the client issues the AUTH command with the parameter LOGIN and the user name to be used for authentication, encoded with base64 encoding, as specified in [RFC2045]. For example, if the client's user name is "Charlie", the client initiates AUTH LOGIN as follows (*auth_login_command_user* as specified in section 2.2.2):

```
AUTH LOGIN Q2hhcmxpZQ==<CR><LF>
```

If AUTH LOGIN is not supported, the server responds with a 504 error message as specified in [RFC4954] section 4.

If AUTH LOGIN is supported on the server, the server responds with *auth_login_password_challenge* (as specified in section 2.2.2).

```
334 UGFzc3dvcmQ6<CR><LF>
```

The client then responds with *auth_login_password_response* (as specified in section 2.2.2) with the password to be used for authentication, encoded with base64 encoding. For example, if the client's password is "password", the client responds with the following:

```
cGFzc3dvcmQ=<CR><LF>
```

If the authentication is complete, the server issues a 235 reply for success or a 535 reply for failure, as specified in [RFC4954].

3.1.6 Timer Events

None.

3.1.7 Other Local Events

None.

3.2 Server Details

3.2.1 Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. The described organization is provided to facilitate the explanation of how the protocol behaves. This document does not mandate that implementations

Preliminary

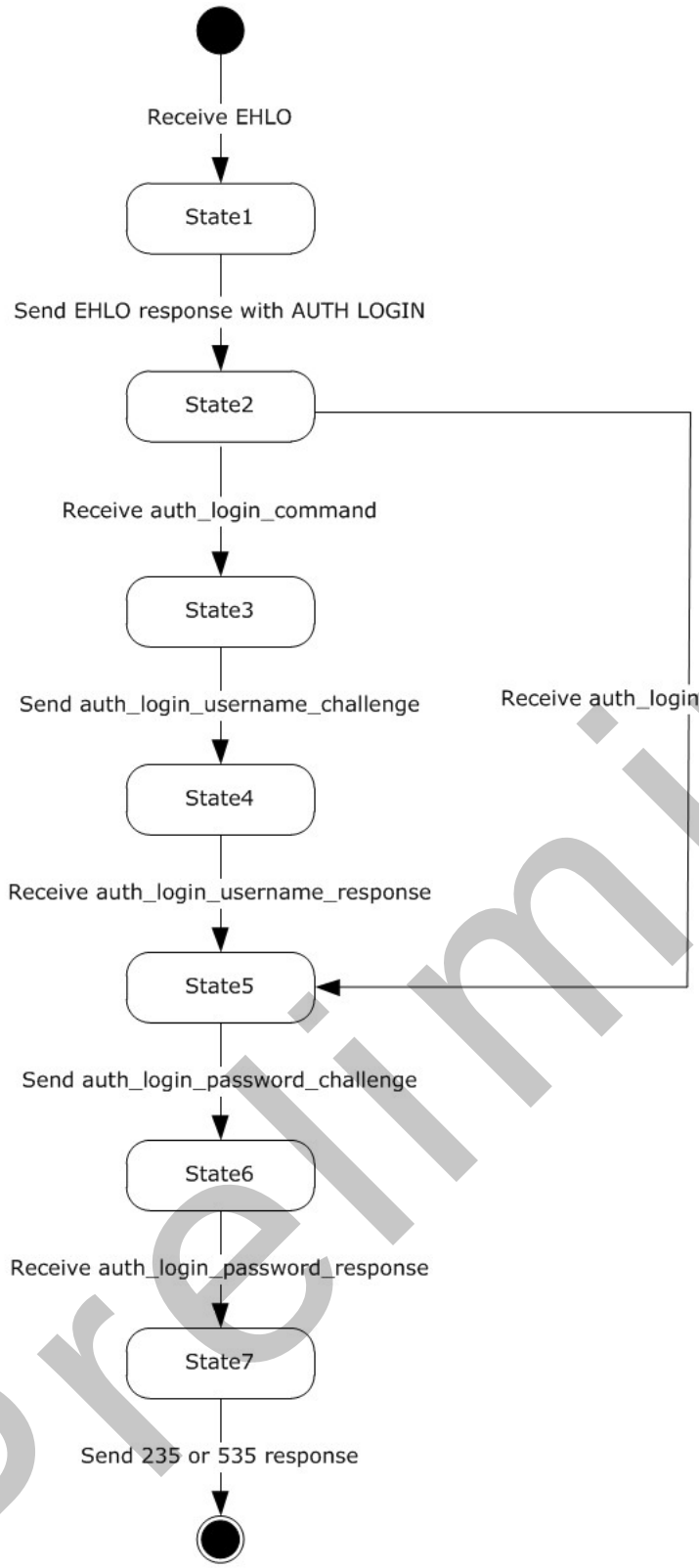
Copyright © 2012 Microsoft Corporation.

adhere to this model as long as their external behavior is consistent with that described in this document.

The abstract data model shown below represents the sequence of messages relative to the server.

Preliminary

Copyright © 2012 Microsoft Corporation.



Preliminary

Copyright © 2012 Microsoft Corporation.

Figure 2: Server abstract data model

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

3.2.5.1 Processing EHLO

If the server supports AUTH LOGIN, it MUST include the AUTH keyword with the LOGIN parameter in its response to the EHLO command, as specified in section 2.2.1.

3.2.5.2 Processing AUTH LOGIN

If the server supports AUTH LOGIN, it MUST respond to the *auth_login_command* request with an *auth_login_username_challenge* response, and it MUST respond to the *auth_login_command_user* request with an *auth_login_password_challenge* response.

If the server does not support AUTH LOGIN, it SHOULD respond with a 504 error message, as specified in [RFC4954] section 4.

3.2.5.3 Processing Username Response

In order to process an *auth_login_username_response* request, the server MUST have sent an *auth_login_username_challenge* response. If the server receives an *auth_login_username_response* request without having sent an *auth_login_username_challenge* response, the server SHOULD respond with a 500 error message.

If the server receives an *auth_login_username_response* request after sending an *auth_login_username_challenge* response, the server SHOULD defer validating the username and SHOULD send an *auth_login_password_challenge* response.

3.2.5.4 Processing Password Response

In order to process an *auth_login_password_response* request, the server MUST have sent an *auth_login_password_challenge* response. If the server receives an *auth_login_password_response* request without having sent an *auth_login_password_challenge* response, the server SHOULD respond with a 500 error message.

The server SHOULD check that the user indicated by the previous *auth_login_username_response* request is a valid user and that the password sent in the *auth_login_password_response* request is a valid password for that user. If the user and password are valid, the server SHOULD respond with a 235 response. If the user or password are invalid, the server SHOULD respond with a 535 response.

Preliminary

Copyright © 2012 Microsoft Corporation.

3.2.6 Timer Events

None.

3.2.7 Other Local Events

None.

Preliminary

Copyright © 2012 Microsoft Corporation.

4 Protocol Example

The following is an example of the use of the AUTH LOGIN extension. The example demonstrates SMTP authentication using the AUTH LOGIN extension.

```
S: 220 SMTP.example.com
C: EHLO client.example.com
S: 250-smtp.example.com  Hello client.example.com
S: 250 AUTH LOGIN
C: AUTH LOGIN
S: 334 VXNlcm5hbWU6
C: Q2hhcmxpZQ==
S: 334 UGFzc3dvcmQ6
C: cGFzc3dvcmQ=
S: 235 authentication successful
```

In this example, a client connects to an SMTP server. The initial response by the server ("220 SMTP.example.com") is the greeting by the server. The client responds with the EHLO command, which is required to activate the extended command set. The server responds with, among other things, an indication of support for AUTH LOGIN. The client then issues the AUTH LOGIN command. The server responds with the first challenge, encoded with base64 encoding. The client responds with "Q2hhcmxpZQ==", which is the name "Charlie", encoded with base64 encoding. The server then issues the second challenge, encoded with base64 encoding, and the client responds with "cGFzc3dvcmQ=", which is the password "password", encoded with base64 encoding. Successful authentication then results in a server response of "235".

Preliminary

Copyright © 2012 Microsoft Corporation.

5 Security

5.1 Security Considerations for Implementers

This extension offers no inherent security mechanisms to protect user credentials during authentication. Because of this, it is extremely important to only use this extension when also using a secure communication channel such as **Secure Sockets Layer (SSL)** or **Transport Layer Security (TLS)**. However, even when using SSL or TLS, the authentication credentials are available to the SMTP server, where it is possible that they could be recorded for future use. The recommendation, therefore, is to only use this extension when no other option is available.

In environments where the use of TLS or other external security is mandated, it is strongly recommended that the AUTH LOGIN advertisement be suppressed until a secure channel is negotiated. TLS in particular exhibits this behavior where the SMTP session is restarted after TLS is negotiated. For more information about TLS, see [RFC4346].

5.2 Index of Security Parameters

None.

Preliminary

Copyright © 2012 Microsoft Corporation.

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.

Preliminary

Copyright © 2012 Microsoft Corporation.

7 Change Tracking

This section identifies changes that were made to the [MS-XLOGIN] 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.

Preliminary

Copyright © 2012 Microsoft Corporation.

- 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.

Preliminary

Copyright © 2012 Microsoft Corporation.

8 Index

A

Abstract data model
  client 9
  server 12
Applicability 6

C

Capability negotiation 6
Change tracking 20
Client
  abstract data model 9
  higher-layer triggered events 11
  initialization 11
  message processing 11
  other local events 12
  sequencing rules 11
  timer events 12
  timers 11
Command and Response ABNF Grammar message 8

D

Data model - abstract
  client 9
  server 12

E

EHLO Response message 8

F

Fields - vendor-extensible 6

G

Glossary 5

H

Higher-layer triggered events
  client 11
  server 15

I

Implementer - security considerations 18
Index of security parameters 18
Informative references 6
Initialization
  client 11
  server 15
Introduction 5

M

Message processing
  client 11
Messages
  Command and Response ABNF Grammar 8
  EHLO Response 8
  transport 8

N

Normative references 5

O

Other local events
  client 12
  server 16
Overview (synopsis) 6

P

Parameters - security index 18
Preconditions 6
Prerequisites 6
Product behavior 19

R

References
  informative 6
  normative 5
Relationship to other protocols 6

S

Security
  implementer considerations 18
  parameter index 18
Sequencing rules
  client 11
Server
  abstract data model 12
  higher-layer triggered events 15
  initialization 15
  other local events 16
  timer events 16
  timers 15
Standards assignments 7

T

Timer events
  client 12
  server 16
Timers
  client 11
  server 15
Tracking changes 20
Transport 8

Preliminary

Copyright © 2012 Microsoft Corporation.

Triggered events - higher-layer
client 11
server 15

V

Vendor-extensible fields 6
Versioning 6

Preliminary

Copyright © 2012 Microsoft Corporation.