[MS-STANOIMAP]:

Outlook Internet Message Access Protocol (IMAP) Standards Support

This document provides a statement of standards support. It is intended for use in conjunction with the Microsoft technical specifications, publicly available standards specifications, network programming art, and Microsoft distributed systems concepts. It assumes that the reader is either familiar with the aforementioned material or has immediate access to it.

A Standards Support document does not require the use of Microsoft programming tools or programming environments in order to implement the standard. Developers who have access to Microsoft programming tools and environments are free to take advantage of them.

Open Specifications documents are intended for use in conjunction with publicly available standards specifications and network programming art and, as such, assume that the reader either is familiar with the aforementioned material or has immediate access to it.

This document describes the choices made when implementing the Internet Message Access Protocol (IMAP) Standard. It identifies ambiguities and implementer choices and indicates the approach taken in the implementation. The details of the implementation itself are described in the specifications for the relevant protocols or data structures, not in this document.

Revision Summary

Date	Revision History	Revision Class	Comments
7/15/2009	1.0.0	Major	Initial Availability.
10/1/2008	1.1.0	Minor	Updated IP notice.
4/10/2009	2.0.0	Major	Updated applicable product releases.
7/15/2009	3.0.0	Major	Revised and edited technical content.
11/4/2009	3.1.0	Minor	Updated the technical content.
2/10/2010	3.2.0	Minor	Updated the technical content.
8/4/2010	3.3	Minor	Clarified the meaning of the technical content.
11/3/2010	3.4	Minor	Clarified the meaning of the technical content.
3/18/2011	3.5	Minor	Clarified the meaning of the technical content.
8/5/2011	3.5	None	No changes to the meaning, language, or formatting of the technical content.
10/7/2011	3.5	None	No changes to the meaning, language, or formatting of the technical content.
1/20/2012	4.0	Major	Significantly changed the technical content.
4/27/2012	4.0	None	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	5.0	Major	Significantly changed the technical content.
10/8/2012	5.0	None	No changes to the meaning, language, or formatting of the technical content.
2/11/2013	5.0	None	No changes to the meaning, language, or formatting of the technical content.
7/26/2013	5.0	None	No changes to the meaning, language, or formatting of the technical content.
11/18/2013	5.0	None	No changes to the meaning, language, or formatting of the technical content.
2/10/2014	5.0	None	No changes to the meaning, language, or formatting of the technical content.
4/30/2014	5.0	None	No changes to the meaning, language, or formatting of the technical content.
7/31/2014	5.0	None	No changes to the meaning, language, or formatting of the technical content.
10/30/2014	5.0	None	No changes to the meaning, language, or formatting of the technical content.
5/26/2015	6.0	Major	Significantly changed the technical content.
9/14/2015	6.0	None	No changes to the meaning, language, or formatting of the technical content.

Date	Revision History	Revision Class	Comments
6/13/2016	6.0	None	No changes to the meaning, language, or formatting of the technical content.
9/14/2016	6.0	None	No changes to the meaning, language, or formatting of the technical content.
2/24/2017	6.0	None	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1 Introduction

This document specifies the level of support provided by Outlook for the Internet Message Access Protocol (IMAP). A client that implements IMAP accesses and manipulates electronic **mailboxes** on an IMAP server in a way that is functionally equivalent to local folders. The Outlook IMAP service component sends IMAP requests to an IMAP server.

1.1 Glossary

This document uses the following terms:

Augmented Backus-Naur Form (ABNF): A modified version of Backus-Naur Form (BNF), commonly used by Internet specifications. ABNF notation balances compactness and simplicity with reasonable representational power. ABNF differs from standard BNF in its definitions and uses of naming rules, repetition, alternatives, order-independence, and value ranges. For more information, see [RFC5234].

base64 encoding: A binary-to-text encoding scheme whereby an arbitrary sequence of bytes is converted to a sequence of printable ASCII characters, as described in [RFC4648].

mailbox: A message store that contains email, calendar items, and other Message objects for a single recipient.

SASL: The Simple Authentication and Security Layer, as described in [RFC2222]. This is an authentication (2) mechanism used by the Lightweight Directory Access Protocol (LDAP).

Transmission Control Protocol (TCP): A protocol used with the Internet Protocol (IP) to send data in the form of message units between computers over the Internet. TCP handles keeping track of the individual units of data (called packets) that a message is divided into for efficient routing through the Internet.

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

1.2 References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the Errata.

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.

[RFC2060] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", RFC 2060, December 1996, http://www.rfc-editor.org/rfc/rfc2060.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

[RFC2246] Dierks, T., and Allen, C., "The TLS Protocol Version 1.0", RFC 2246, January 1999, http://www.rfc-editor.org/rfc/rfc2246.txt

[RFC2822] Resnick, P., Ed., "Internet Message Format", RFC 2822, April 2001, http://www.ietf.org/rfc/rfc2822.txt

[RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", RFC 3501, March 2003, http://www.rfc-editor.org/rfc/rfc3501.txt

[RFC6154] Leiba, B., and Nicolson, J., "IMAP LIST Extension for Special-Use Mailboxes", RFC 6154, March 2011, http://www.rfc-editor.org/rfc/rfc6154.txt

1.2.2 Informative References

None.

1.3 Microsoft Implementations

Microsoft Office Outlook 2007

Microsoft Outlook 2010

Microsoft Outlook 2013

Microsoft Outlook 2016

1.4 Standards Support Requirements

The conformance requirements for [RFC3501] are as follows:

- All required portions of the specification are implemented according to the specification.
- Any recommended portions that are implemented are implemented according to the specification.
- Any optional portions that are implemented are implemented according to the specification.

The following table lists the sections of [RFC3501] that are considered normative and the sections that are considered informative.

Section(s)	Normative/Informative
1	Informative
2 - 7	Normative
8	Informative
9	Normative
10 - 12	Informative

1.5 Notation

The following notations are used in this specification.

Notation	Explanation
C####	This identifies a clarification of ambiguity in the target specification. This includes imprecise statements, omitted information, discrepancies, and errata. This does not include data formatting

Notation	Explanation
	clarifications.
V####	This identifies an intended point of variability in the target specification such as the use of MAY, SHOULD, or RECOMMENDED. This does not include extensibility points.
E####	Because the use of extensibility points (such as optional implementation-specific data) can impair interoperability, this profile identifies such points in the target specification.

2 Standards Support Statements

2.1 Normative Variations

The following sub-sections detail the normative variations from [RFC3501].

2.1.1 [RFC3501] Section 2.1, Port 143

The specification states: "When **TCP** is used, an IMAP4rev1 server listens on port 143."

Outlook uses port 143 for non-SSL connections and port 993 for SSL connections. However, Outlook can be configured to use any port number.

2.1.2 [RFC3501] Section 3.4, Client MUST Read the OK Response Before Closing the Connection

The specification states that the client MUST read the tagged OK response to the LOGOUT command before the client closes the connection.

Outlook can be configured to send a LOGOUT command that immediately terminates the connection.

2.1.3 [RFC3501] Section 9, ABNF Rules in General

The specification states that **ABNF** rules MUST be strictly followed.

Outlook strictly follows the rules when sending requests or data to the server, but is more forgiving when parsing responses from the server.

2.1.4 [RFC3501] Section 9, Rule Regarding Spaces

The specification states: "In all cases, SP refers to exactly one space. It is NOT permitted to substitute TAB, insert additional spaces, or otherwise treat SP as being equivalent to LWSP."

Outlook strictly follows this rule when sending requests or data to the server, but is more forgiving when parsing responses from the server.

2.1.5 [RFC3501] Section 9, Body-extension Field

The specification states that, for future expansion, a client MUST accept body-extension fields.

Outlook does not accept body-extension fields.

2.1.6 [RFC3501] Section 9, Flag-extension Field

The specification states that, for future expansion, a client MUST accept flag-extension fields.

Outlook ignores flag-extension fields.

2.1.7 [RFC3501] Section 11.1, Client MUST Implement the TLS_RSA_WITH_RC4_128_MD5 Cipher Suite

The specification states that the client MUST implement the TLS_RSA_WITH_RC4_128_MD5 cipher suite.

Outlook does not implement the TLS_RSA_WITH_RC4_128_MD5 cipher suite and, instead, relies on the operating system to provide the implementation.

2.2 Clarifications

The following sub-sections identify clarifications relative to [RFC3501] or [RFC2822].

Unless otherwise stated, the specified products conform to all SHOULD and RECOMMENDED behavior in [RFC3501] and [RFC2822]. The term "can" is used throughout [RFC3501] and [RFC2822] and is interpreted to indicate optional behavior.

2.2.1 [RFC3501] Section 2.2.1, Client Protocol Sender and Server Protocol Receiver

C0001:

The specification states that each client command is prefixed with an identifier, called a tag, but does not make a specific requirement on format. Later in the specification (section 9), the syntax is explicitly stated.

Microsoft Office Outlook 2007, Microsoft Outlook 2010, Microsoft Outlook 2013, Microsoft Outlook 2016

Outlook randomly generates a sequence of four characters of lowercase letters "a" through "z", and/or digits 0 through 9.

2.2.2 [RFC3501] Section 2.2.2, Server Protocol Sender and Client Protocol Receiver

V0001:

The specification states: "Server data MAY be sent as a result of a client command, or MAY be sent unilaterally by the server. There is no syntactic difference between server data that resulted from a specific command and server data that were sent unilaterally."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook can handle unrequested responses.

C0002:

The specification states: "Server data SHOULD be recorded, so that the client can reference its recorded copy rather than sending a command to the server to request the data. In the case of certain server data, the data MUST be recorded."

This part of the specification does not state which server data MUST be recorded. A thorough reading of the specification reveals that the client MUST record the following server data:

- **Mailbox** size updates (section 5.2 of the specification)
- Updates from the FLAGS response (section 7.2.6 of the specification)
- Updates from the EXISTS response (section 7.3.1 of the specification)
- Updates from the RECENT response (section 7.3.2 of the specification)
- Updates from the EXPUNGE response (section 7.4.1 of the specification)

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Required recording of certain server data: Outlook records only three of the five server data that are required to be recorded. The following are not recorded:

- Updates from the FLAGS response — Outlook ignores the FLAGS response, as specified in section 2.2.46 of this document.
- Updates from the RECENT response — Outlook ignores the RECENT response, as specified in section 2.2.47 of this document.

Recommended recording of other server data: Outlook maintains a local cache of all user data and some meta-data (UIDs, hierarchy characters, etc.), thereby reducing the likelihood of the same data being repeatedly fetched from the server.

2.2.3 [RFC3501] Section 2.3, Message Attributes

C0003:

The specification states: "In addition to message text, each message has several attributes associated with it." It is unclear whether "each message has several attributes associated with it" is a normative requirement.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not use some of the message attributes. For more details, see sections 2.2.4 through 2.2.9.

2.2.4 [RFC3501] Section 2.3.1.1, Unique Identifier (UID) Message Attribute

V0002:

The specification describes a next unique identifier.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not use the next unique identifier for any purpose.

V0003:

The specification states that unique identifiers SHOULD persist at all times.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

If a unique identifier does not persist, then Outlook fully resynchronizes the contents of each folder upon each visit in which the unique identifier was not persisted.

2.2.5 [RFC3501] Section 2.3.2, Flags Message Attribute

V0004:

The specification defines the \Recent flag.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook ignores the \Recent flag.

V0005:

The specification states: "Servers MAY permit the client to define new keywords in the **mailbox**."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not define new keywords under any circumstances.

2.2.6 [RFC3501] Section 2.3.3, Internal Date Message Attribute

V0006:

The specification states that, in the case of messages delivered via SMTP, the value of the Internal Date message attribute SHOULD be the date and time of final delivery of the message as defined by SMTP.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook can be configured to use either the value of the Internal Date message attribute or the value of the message header as the date and time of final message delivery.

2.2.7 [RFC3501] Section 2.3.4, [RFC2822] Size Message Attribute

V0007:

The specification defines the [RFC2822] Size message attribute, which specifies the number of octets in the message.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook uses the [RFC2822] Size value only for message-size display purposes; Outlook does not use the [RFC2822] Size value for any logical processing.

2.2.8 [RFC3501] Section 2.3.5, Envelope Structure Message Attribute

V0008:

The specification defines the Envelope Structure message attribute.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook never requests the Envelope Structure message attribute.

2.2.9 [RFC3501] Section 2.3.6, Body Structure Message Attribute

V0009:

The specification defines the Body Structure message attribute.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook never requests the Body Structure message attribute.

2.2.10 [RFC3501] Section 2.4, Message Texts

V0010:

The specification states: "In addition to being able to fetch the full [RFC2822] text of a message, IMAP4rev1 permits the fetching of portions of the full message text."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook can fetch headers and bodies separately but cannot fetch individual body parts separately.

2.2.11 [RFC3501] Section 3.4, Logout State

V0011:

The specification states: "A client SHOULD NOT unilaterally close the connection, and instead SHOULD issue a LOGOUT command."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Under certain error conditions, Outlook drops the connection without sending a LOGOUT command.

2.2.12 [RFC3501] Section 4.3, String

V0012:

The specification states: "The empty string is represented as either "" (a quoted string with zero characters between double quotes) or as {0} followed by CRLF (a literal with an octet count of 0)."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook correctly generates and handles both representations.

2.2.13 [RFC3501] Section 4.3.1, 8-bit and Binary Strings

V0013:

The specification states that implementations MAY transmit 8-bit or multi-octet characters in literals, but SHOULD do so only when the IANA-registered character set is identified.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not transmit 8-bit or multi-octet characters.

V0014:

The specification states that implementations MUST encode binary data into a textual form, such as base64, before transmitting the data.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook uses **base64 encoding**.

V0015:

The specification states: "A string with an excessive amount of CTL characters MAY also be considered to be binary."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not modify the string based on number of CTL characters.

2.2.14 [RFC3501] Section 5.1, Mailbox Naming

V0016:

The specification does not state a requirement for case sensitivity in non-INBOX **mailbox** names.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook allows folders to have names that differ only in case in the local IMAP store. Outlook functions correctly against a store that is case-insensitive.

V0017:

The specification states: "Any character which is one of the atom-specials (see the Formal Syntax) will require that the mailbox name be represented as a quoted string or literal."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook generates a quoted string, for a short name, and a literal, for a longer name, when encoding atom-specials. Outlook consumes both representations correctly.

V0018:

The specification states that CTL and other non-graphic characters are difficult to represent in a user interface and are best avoided.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook accepts only standard keyboard input.

V0019:

The specification states: "Although the list-wildcard characters ('%' and '*') are valid in a mailbox name, it is difficult to use such mailbox names with the LIST and LSUB commands due to the conflict with wildcard interpretation."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook allows a folder name to contain the "%" and "*" characters. If a server does not allow these characters, then Outlook handles error responses that result from the CREATE command and displays the error messages to the user.

V0020:

The specification states: "Usually, a character (determined by the server implementation) is reserved to delimit levels of hierarchy."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not allow a folder name to contain the hierarchy delimiter.

V0021:

The specification states: "Two characters, '#' and '&', have meanings by convention, and should be avoided except when used in that convention."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook allows a folder name to contain the "#" and "&" characters. The "&" character is encoded as "&-". For example, the folder name "a&b" is transmitted as "a&-b". A folder name that contains the "#" character is transmitted without any escaping.

2.2.15 [RFC3501] Section 5.1.2, Mailbox Namespace Naming Convention

V0022:

The specification describes how the namespace identifier is used in the name of a **mailbox** namespace.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not perform special handling of the namespace identifier. The "#" character can be used as part of the root-folder path in the Outlook configuration user interface (this allows a user to map multiple namespaces into the same Outlook profile). Outlook has no intrinsic knowledge of a namespace, and, therefore, treats a root-folder path that specifies a namespace the same as one that does not specify a namespace.

2.2.16 [RFC3501] Section 5.2, Mailbox Size and Message Status Updates

V0023:

The specification states: "At any time, a server can send data that the client did not request. Sometimes, such behavior is REQUIRED. For example, agents other than the server MAY add messages to the **mailbox**."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook handles unrequested data from the server and from agents other than the server.

2.2.17 [RFC3501] Section 5.3, Response When No Command in Progress

V0024:

The specification states that server implementations are permitted to send an untagged response (except for EXPUNGE) while there is no command in progress.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook handles unilateral responses of all types.

2.2.18 [RFC3501] Section 5.4, Autologout Timer

V0025:

The specification describes an optional inactivity timer on the server.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook automatically generates NOOP commands in 10-minute (or less) intervals to avoid having its connections dropped due to inactivity.

2.2.19 [RFC3501] Section 5.5, Multiple Commands in Progress

V0026:

The specification states: "The client MAY send another command without waiting for the completion result response of a command."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook is capable of sending another command without waiting for the completion-result response of the previous command.

2.2.20 [RFC3501] Section 6.1.1, CAPABILITY Command

V0027:

The specification describes the "AUTH=" capability name, which specifies an authentication mechanism that the server supports.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook recognizes "AUTH=NTLM" and "AUTH= DIGEST-MD5" in the CAPABILITY response.

2.2.21 [RFC3501] Section 6.1.2, NOOP Command

V0028:

The specification states that the NOOP command can be used as a periodic poll for new messages or message status updates during a period of inactivity and can also be used to reset any inactivity autologout timer on the server.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook uses the NOOP command as follows:

- If the server does not support the IDLE extension, then Outlook uses NOOP to poll.
- Regardless of whether the server supports the IDLE extension, then Outlook uses NOOP to reset the auto-logout timer.

2.2.22 [RFC3501] Section 6.2, Access Without Establishing Authentication

V0029:

The specification describes how the client can access certain **mailboxes** without establishing authentication.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not support the Anonymous **SASL** authenticator and does not have any method for logging on as "anonymous".

2.2.23 [RFC3501] Section 6.2.2, AUTHENTICATE Command

V0030:

The specification states: "If the server supports the requested authentication mechanism, it performs an authentication protocol exchange to authenticate and identify the client. It MAY also negotiate an OPTIONAL security layer for subsequent protocol interactions."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not perform out-of-band AUTHENTICATE negotiation.

E0001:

The specification states that the client is not required to implement any authentication mechanisms other than the PLAIN authentication mechanism.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

In addition to the PLAIN authentication mechanism, Outlook implements the following authentication mechanisms:

- NTLM

- DIGEST-MD5

V0031:

The specification states: "A server implementation MUST implement a configuration in which it does NOT permit any plaintext password mechanisms, unless either the STARTTLS command has been negotiated or some other mechanism that protects the session from password snooping has been provided."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not require protection against password snooping.

E0002:

The specification states that the client SHOULD implement additional **SASL** mechanisms that do not use plaintext passwords.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

For more details, see E0001 in this section.

V0032:

The specification states: "The server SHOULD list its supported authentication mechanisms in the response to the CAPABILITY command so that the client knows which authentication mechanisms to use."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook uses only authentication mechanisms that are advertised by the server. If the server does not advertise any authentication mechanisms, then Outlook uses the LOGIN command.

V0033:

The specification states: "A server MAY include a CAPABILITY response code in the tagged OK response of a successful AUTHENTICATE command in order to send capabilities automatically. It is unnecessary for a client to send a separate CAPABILITY command if it recognizes these automatic capabilities."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook sends a CAPABILITY command regardless of whether it receives a CAPABILITY response code in response to a successful AUTHENTICATE command.

V0034:

The specification states: "If an AUTHENTICATE command fails with a NO response, the client MAY try another authentication mechanism by issuing another AUTHENTICATE command. It MAY also attempt to authenticate by using the LOGIN command. In other words, the client MAY request authentication types in decreasing order of preference, with the LOGIN command as a last resort."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Depending on user/administrator provided policies, Outlook successively tries decreasingly strong methods of authentication until it finds one that works. Outlook uses the LOGIN command to authenticate if it does not find an authentication method that works.

Outlook also uses the LOGIN command to authenticate when the server does not advertise any authentication mechanisms and when the LOGIN command is not disabled by a user/administrator policy.

2.2.24 [RFC3501] Section 6.2.3, LOGIN Command

V0035:

The specification states: "A server MAY include a CAPABILITY response code in the tagged OK response to a successful LOGIN command in order to send capabilities automatically. It is unnecessary for a client to send a separate CAPABILITY command if it recognizes these automatic capabilities."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook sends a CAPABILITY command regardless of whether the server sends the capabilities automatically.

2.2.25 [RFC3501] Section 6.3.1, SELECT Command

V0036:

The specification states that the server MUST send the following untagged data to the client before returning a tagged OK for the SELECT command response: FLAGS, EXISTS, RECENT, UNSEEN, PERMANENTFLAGS, UIDNEXT, and UIDVALIDITY. The specification also states that the client implementation SHOULD have default behavior for handling a SELECT response that is missing the required untagged data.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook requires the EXISTS untagged data and the UIDVALIDITY untagged data for proper operation. Outlook is unaffected if any of the following untagged data are missing: FLAGS, RECENT, UNSEEN, PERMANENTFLAGS, or UIDNEXT.

V0037:

The specification notes the following consequences for the client when the SELECT command response is missing certain untagged data:

- UNSEEN — If this data is missing, then the client cannot make any assumptions about the first unseen message in the **mailbox**, and needs to issue a SEARCH command if it wants to find it.
- PERMANENTFLAGS — If this data is missing, then the client should assume that all flags can be changed permanently.
- UIDNEXT — If this data is missing, then the client cannot make any assumptions about the next unique identifier value.
- UIDVALIDITY — If this data is missing, then the server does not support unique identifiers.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook's behavior for each of these missing untagged data items is as follows:

- UNSEEN — Outlook ignores this data and, therefore, takes no special action if it is missing.
- PERMANENTFLAGS — Outlook always assumes that all flags can be changed permanently.
- UIDNEXT — Outlook ignores this data and, therefore, takes no special action if it is missing.
- UIDVALIDITY — Outlook does not function properly against a server that does not support unique identifiers.

V0038:

The specification states: "If the client is permitted to modify the mailbox, the server SHOULD prefix the text of the tagged OK response with the READ-WRITE response code."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook ignores the READ-WRITE response code and, therefore, takes no special action if a folder is advertised as READ-WRITE.

V0039:

The specification states: "If the client is not permitted to modify the mailbox but is permitted read access, the mailbox is selected as read-only and the server MUST prefix the text of the tagged OK response to SELECT with the READ-ONLY response code."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook ignores the READ-ONLY response code and, therefore, takes no special action if a folder is advertised as READ-ONLY.

2.2.26 [RFC3501] Section 6.3.2, EXAMINE Command

V0040:

The specification defines the EXAMINE command.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not use the EXAMINE command.

2.2.27 [RFC3501] Section 6.3.8, LIST Command

V0041:

The specification states that server implementations are permitted to hide otherwise accessible mailboxes from the wildcard characters.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

When Outlook is in "show all folders" mode, Outlook requires that the server expose the hidden folders for wildcards, thereby making the full hierarchy discoverable.

V0042:

The specification states: "The special name INBOX is included in the output from LIST, if INBOX is supported by this server for this user and if the uppercase string 'INBOX' matches the interpreted reference and **mailbox** name arguments with wildcards. The criteria for omitting INBOX is whether SELECT INBOX will return failure."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook interoperates properly with a server that does not expose INBOX.

2.2.28 [RFC3501] Section 6.3.9, LSUB Command

V0043:

The specification states: "The returned untagged LSUB response MAY contain different **mailbox** flags from a LIST untagged response. If this should happen, the flags in the untagged LIST are considered more authoritative."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook uses the flags supplied with the most recent response to either LSUB or LIST. In other words, the authority is neither LSUB nor LIST, but the most recently received response to either of these two commands.

2.2.29 [RFC3501] Section 6.3.10, STATUS Command

V0044:

The specification states that clients SHOULD NOT expect to be able to issue many consecutive STATUS commands and obtain reasonable performance.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook can be configured to send a large number of these commands consecutively, but does so on a background thread to avoid delaying user operations.

V0045:

The specification states that the following status data items can be requested: MESSAGES, RECENT, UIDNEXT, UIDVALIDITY, and UNSEEN.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook requests only the UNSEEN status data item.

2.2.30 [RFC3501] Section 6.3.11, APPEND Command

V0046:

The specification states that the "message literal" argument SHOULD be in the format of an [RFC2822] message, but there MAY be exceptions, such as draft messages, in which the required [RFC2822] header lines are omitted from the message literal argument.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook is capable of uploading incomplete draft messages.

V0047:

The specification states: "Unless it is certain that the destination **mailbox** cannot be created, the server MUST send the response code TRYCREATE as the prefix of the text of the tagged NO response."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook ignores the TRYCREATE response code.

V0048:

The specification states: "If the mailbox is currently selected, the normal new message actions SHOULD occur. Specifically, the server SHOULD notify the client immediately via an untagged EXISTS response. If the server does not do so, the client MAY issue a NOOP command (or failing that, a CHECK command) after one or more APPEND commands."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

The absence of an untagged EXISTS response does not determine whether Outlook sends a NOOP command after an APPEND command. In other words, any generation of a NOOP in this case is independent of the untagged EXISTS response. Outlook does not use the CHECK command.

2.2.31 [RFC3501] Section 6.4.1, CHECK Command

V0049:

The specification defines the CHECK command.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not use the CHECK command.

2.2.32 [RFC3501] Section 6.4.2, CLOSE Command

V0050:

The specification states: "Even if a **mailbox** is selected, a SELECT, EXAMINE, or LOGOUT command MAY be issued without previously issuing a CLOSE command."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook can issue a SELECT command or a LOGOUT command without previously issuing a CLOSE command. Outlook does not use the EXAMINE command.

2.2.33 [RFC3501] Section 6.4.4, SEARCH Command

V0051:

The specification defines the SEARCH command.

Office Outlook 2007, Outlook 2010

Outlook does not use the SEARCH command.

Outlook 2013, Outlook 2016

Outlook uses the SEARCH command.

2.2.34 [RFC3501] Section 6.4.5, FETCH Command

V0052:

The specification states: "There are three macros which specify commonly-used sets of data items, and can be used instead of data items."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not use these macros.

V0053:

The specification defines BODY[<section>] <<partial>> as one of the data items that can be fetched.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not issue a FETCH command with the BODY[<section>] <<partial>> data item. The data items that Outlook can fetch are specified in V0054 and V0055 in this section.

V0054:

The specification defines BODY.PEEK[<section>] <<partial>> as one of the data items that can be fetched, where <section> is a set of zero or more part specifiers delimited by periods. The specification defines a part specifier as either a part number or one of the following: HEADER, HEADER.FIELDS, HEADER.FIELDS.NOT, MIME, and TEXT.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook uses either the HEADER part specifier or zero part specifiers; Outlook does not use a part number and does not use <partial>. Therefore, the BODY.PEEK data item has the following forms in an Outlook fetch:

- BODY.PEEK[]
- BODY.PEEK[HEADER]

V0055:

The specification defines data items that can be fetched.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook issues a FETCH command with data items as follows. For more details about the BODY.PEEK data item, see V0054 in this section.

To all servers:

- (UID)
- (UID FLAGS)

Only to IMAP4 servers that do not send "IMAP4rev1" as a listed capability in a CAPABILITY response as specified in [RFC3501] section 6.1.1:

- (UID FLAGS RFC822.SIZE RFC822.HEADER INTERNALDATE)
- (UID FLAGS RFC822)
- (UID FLAGS RFC822.SIZE RFC822.HEADER)
- (UID FLAGS RFC822.PEEK)

Note: The RFC822.PEEK data item was obsoleted by [RFC2060]. However, Outlook still uses the RFC822.PEEK data item, as specified here.

Only to IMAP4 servers that send "IMAP4rev1" as a listed capability in a CAPABILITY response:

- (UID FLAGS RFC822.SIZE BODY.PEEK[HEADER] INTERNALDATE)
- (UID FLAGS BODY.PEEK[])
- (UID FLAGS RFC822.SIZE BODY.PEEK[HEADER])
- (UID FLAGS BODY.PEEK[])

2.2.35 [RFC3501] Section 6.4.6, STORE Command

V0056:

The specification defines data items that can be stored.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook can issue the STORE command with the data items as follows:

- FLAGS (<flags>)

where <flags> can be any combination of the following: \Seen, \Draft, \Flagged, \Answered, \Deleted

- +FLAGS (\Deleted \Seen)
- -FLAGS (\Deleted)
- +FLAGS (\Seen)
- -FLAGS (\Seen)
- +FLAGS.SILENT (\Deleted)
- -FLAGS.SILENT (\Deleted)
- +FLAGS.SILENT (\Seen)
- -FLAGS.SILENT (\Seen)
- +FLAGS.SILENT (\Flagged)
- -FLAGS.SILENT (\Flagged)
- +FLAGS.SILENT (\Answered)
- -FLAGS.SILENT (\Answered)

2.2.36 [RFC3501] Section 6.4.7, COPY Command

V0057:

The specification states: "The flags and internal date of the message(s) SHOULD be preserved, and the \Recent flag SHOULD be set, in the copy."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook might display the wrong date on the target message if the server does not preserve the flags and internal date of the message(s). Outlook is unaffected by the setting of the \Recent flag.

V0058:

The specification states: "Unless it is certain that the destination **mailbox** cannot be created, the server MUST send the response code TRYCREATE as the prefix of the text of the tagged NO response."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook ignores the TRYCREATE response code.

2.2.37 [RFC3501] Section 6.4.8, UID Command

V0059:

The specification states that the UID command takes the following commands as arguments: COPY, FETCH, STORE, or SEARCH.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook can issue a UID COPY, UID FETCH, or UID STORE, but cannot issue a UID SEARCH.

V0060:

The specification states that, in regards to a UID COPY, UID FETCH, or UID STORE, the numbers in the sequence set argument are unique identifiers instead of message sequence numbers and sequence set ranges are permitted, but there is no guarantee that unique identifiers will be contiguous.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook can issue UID COPY, UID FETCH, or UID STORE with sequence set ranges.

2.2.38 [RFC3501] Section 6.5, Client Commands — Experimental/Expansion

V0061:

The specification describes how to define an experimental command or any command that is not part of the specification.

Office Outlook 2007, Outlook 2010

Outlook does not define any such commands.

Outlook 2013, Outlook 2016

Outlook implements the IMAP LIST extension specified in [RFC6154] as the **XLIST** command.

2.2.39 [RFC3501] Section 7, Server Responses

C0004:

The specification states: "Certain server data MUST be recorded by the client when it is received;" and "Other server data SHOULD be recorded for later reference; if the client does not need to record the data, or if recording the data has no obvious purpose (e.g., a SEARCH response when no SEARCH command is in progress), the data SHOULD be ignored."

This part of the specification does not state which server data MUST be recorded. A thorough reading of the specification reveals that the client MUST record the following server data:

- **Mailbox** size updates (section 5.2 of the specification)
- Updates from the FLAGS response (section 7.2.6 of the specification)
- Updates from the EXISTS response (section 7.3.1of the specification)
- Updates from the RECENT response (section 7.3.2 of the specification)
- Updates from the EXPUNGE response (section 7.4.1 of the specification)

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

For details, see C0002 in section 2.2.2 of this document.

2.2.40 [RFC3501] Section 7.1, Server Responses — Status Responses

V0062:

The specification states that status responses MAY include an OPTIONAL response code.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook processes the following response codes: ALERT, UIDVALIDITY, and UNSEEN. Outlook sometimes caches the UNSEEN response code and presents it to the user.

Outlook ignores the following response codes:

- BADCHARSET
- CAPABILITY
- PARSE
- PERMANENTFLAGS
- READ-ONLY
- READ-WRITE
- TRYCREATE
- UIDNEXT

2.2.41 [RFC3501] Section 7.1.1, OK Response

V0063:

The specification states that the human-readable text contained in the OK response MAY be presented to the user as an information message.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook presents the human readable text to the user only when an ALERT response code is also included in the OK response.

2.2.42 [RFC3501] Section 7.1.5, BYE Response

V0064:

The specification states that the human-readable text contained in the BYE response MAY be presented to the user in a status report.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook never displays this human-readable text to the user.

2.2.43 [RFC3501] Section 7.2.1, CAPABILITY Response

V0065:

The specification states: "A server MAY send capabilities automatically, by using the CAPABILITY response code in the initial PREAUTH or OK responses, and by sending an updated CAPABILITY response code in the tagged OK response as part of a successful authentication. It is unnecessary for a client to send a separate CAPABILITY command if it recognizes these automatic capabilities."

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook parses the **CAPABILITY** response code but sends a separate **CAPABILITY** command to retrieve the capabilities and merges the results.

2.2.44 [RFC3501] Section 7.2.2, LIST Response

V0066:

The specification defines four name attributes for the LIST response.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook recognizes the \Noinferiors and \Noselect name attributes. Outlook ignores the \Marked and \Unmarked name attributes.

2.2.45 [RFC3501] Section 7.2.5, SEARCH Response

V0067:

The specification describes the response to a **SEARCH** command.

Office Outlook 2007, Outlook 2010

Outlook does not use the **SEARCH** command and ignores all **SEARCH** responses.

Outlook 2013, Outlook 2016

Outlook uses the **SEARCH** command and does not ignore the **SEARCH** response.

2.2.46 [RFC3501] Section 7.2.6, FLAGS Response

V0068:

The specification describes the FLAGS response, which occurs as a result of a SELECT or EXAMINE command.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook ignores the FLAGS response.

2.2.47 [RFC3501] Section 7.3.2, RECENT Response

V0069:

The specification describes the RECENT response, which occurs as a result of a SELECT or EXAMINE command.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook ignores the RECENT response.

2.2.48 [RFC3501] Section 7.4.2, FETCH Response

V0070:

The specification describes the response to a FETCH request that specifies the BODY[<section>]<<partial>> data item.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook never sends a FETCH request for a partial body of a message. If a FETCH response includes a partial body, then it is possible that Outlook will overwrite the full body with the partial body.

The FETCH requests that Outlook is capable of sending are specified in V0054 and V0055 in section 2.2.34 of this document.

V0071:

The specification describes the response to a FETCH request that specifies the BODYSTRUCTURE data item.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook never sends a FETCH request that specifies the BODYSTRUCTURE data item.

The FETCH requests that Outlook is capable of sending are specified in V0054 and V0055 in section 2.2.34 of this document.

V0072:

The specification describes the response to a FETCH request that specifies the ENVELOPE data item.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook never sends a FETCH request that specifies the ENVELOPE data item.

The FETCH requests that Outlook is capable of sending are specified in V0054 and V0055 in section 2.2.34 of this document.

2.2.49 [RFC3501] Section 11.1, STARTTLS Security Considerations

V0073:

The specification states that the client SHOULD implement the TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA cipher suite, as specified in [RFC2246].

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not implement the TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA cipher suite and, instead, relies on the operating system to provide the implementation.

V0074:

The specification states that all cipher suites other than TLS_RSA_WITH_RC4_128_MD5 and TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA are OPTIONAL.

Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016

Outlook does not implement any optional cipher suites and, instead, relies on the operating system to provide the implementation of any optional cipher suites.

2.2.50 [RFC2822] Section 3.6, Field Definitions

V0075:

The specification states that header fields SHOULD NOT be reordered when a message is transported or transformed.

Office Outlook 2007, Outlook 2010, Outlook 2013

Outlook can put the header fields in an arbitrary order when a message is transported or transformed.

V0076:

The specification states that trace header fields and resent header fields MUST NOT be reordered.

Office Outlook 2007, Outlook 2010, Outlook 2013

Outlook can put trace header fields and resent header fields in an arbitrary order.

2.2.51 [RFC2822] Section 3.6.8, Optional Fields

V0077:

The specification states "Fields may appear in messages that are otherwise unspecified in this standard" and further states that such fields are uninterpreted.

Office Outlook 2007, Outlook 2010, Outlook 2013

Outlook deletes fields that are unspecified.

2.3 Error Handling

None.

2.4 Security

None.

3 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

4 Index