Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation (“this documentation”) for protocols, file formats, data portability, computer languages, and standards support. Additionally, overview documents cover inter-protocol relationships and interactions.

- **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 can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications documentation.

- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.

- **Patents.** Microsoft has patents that might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications document might be covered by the Microsoft [Open Specifications Promise](https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-oxdscli/224e5276-1a9e-4a60-9b2d-b310d082eb2d) or the [Microsoft Community Promise](https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-oxdscli/224e5276-1a9e-4a60-9b2d-b310d082eb2d). If you would prefer a written license, or if the technologies described in this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting [iplg@microsoft.com](mailto:iplg@microsoft.com).

- **License Programs.** To see all of the protocols in scope under a specific license program and the associated patents, visit the [Patent Map](https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-oxdscli/224e5276-1a9e-4a60-9b2d-b310d082eb2d).

- **Trademarks.** The names of companies and products contained in this documentation might be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit [www.microsoft.com/trademarks](https://www.microsoft.com/en-us/about/legal/trademarks).

- **Fictitious Names.** The example companies, organizations, products, domain names, email addresses, logos, people, places, and events that are 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 as specifically described above, whether by implication, estoppel, or otherwise.

**Tools.** The Open Specifications documentation does 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 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.

**Support.** For questions and support, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com).
# Revision Summary

<table>
<thead>
<tr>
<th>Date</th>
<th>Revision History</th>
<th>Revision Class</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>4/4/2008</td>
<td>0.1</td>
<td>New</td>
<td>Initial Availability.</td>
</tr>
<tr>
<td>4/25/2008</td>
<td>0.2</td>
<td>Minor</td>
<td>Revised and updated property names and other technical content.</td>
</tr>
<tr>
<td>6/27/2008</td>
<td>1.0</td>
<td>Major</td>
<td>Initial Release.</td>
</tr>
<tr>
<td>8/6/2008</td>
<td>1.01</td>
<td>Minor</td>
<td>Revised and edited technical content.</td>
</tr>
<tr>
<td>9/3/2008</td>
<td>1.02</td>
<td>Minor</td>
<td>Revised and edited technical content.</td>
</tr>
<tr>
<td>12/3/2008</td>
<td>1.03</td>
<td>Minor</td>
<td>Revised and edited technical content.</td>
</tr>
<tr>
<td>4/10/2009</td>
<td>2.0</td>
<td>Major</td>
<td>Updated technical content and applicable product releases.</td>
</tr>
<tr>
<td>7/15/2009</td>
<td>3.0</td>
<td>Major</td>
<td>Revised and edited for technical content.</td>
</tr>
<tr>
<td>11/4/2009</td>
<td>4.0.0</td>
<td>Major</td>
<td>Updated and revised the technical content.</td>
</tr>
<tr>
<td>2/10/2010</td>
<td>5.0.0</td>
<td>Major</td>
<td>Updated and revised the technical content.</td>
</tr>
<tr>
<td>5/5/2010</td>
<td>5.1.0</td>
<td>Minor</td>
<td>Updated the technical content.</td>
</tr>
<tr>
<td>8/4/2010</td>
<td>6.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>11/3/2010</td>
<td>6.1</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>3/18/2011</td>
<td>7.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>8/5/2011</td>
<td>8.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>10/7/2011</td>
<td>8.0</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>1/20/2012</td>
<td>9.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>4/27/2012</td>
<td>9.1</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>7/16/2012</td>
<td>9.1</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>10/8/2012</td>
<td>10.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>2/11/2013</td>
<td>10.0</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>7/26/2013</td>
<td>10.1</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>11/18/2013</td>
<td>10.1</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>2/10/2014</td>
<td>11.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>4/30/2014</td>
<td>11.0.1</td>
<td>Editorial</td>
<td>Changed language and formatting in the technical content.</td>
</tr>
<tr>
<td>7/31/2014</td>
<td>11.0.1</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>Date</td>
<td>Revision History</td>
<td>Revision Class</td>
<td>Comments</td>
</tr>
<tr>
<td>------------</td>
<td>------------------</td>
<td>----------------</td>
<td>----------------------------------------------------</td>
</tr>
<tr>
<td>10/30/2014</td>
<td>11.1</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>3/16/2015</td>
<td>12.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>5/26/2015</td>
<td>13.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>9/14/2015</td>
<td>14.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>6/13/2016</td>
<td>15.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>9/14/2016</td>
<td>15.0</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>6/20/2017</td>
<td>16.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>10/17/2017</td>
<td>16.1</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>12/12/2017</td>
<td>17.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>7/24/2018</td>
<td>18.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>10/1/2018</td>
<td>19.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>2/19/2020</td>
<td>19.1</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>5/19/2020</td>
<td>19.2</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>4/22/2021</td>
<td>20.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>8/17/2021</td>
<td>21.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
</tbody>
</table>
# Table of Contents

1 Introduction ........................................................................................................... 7  
  1.1 Glossary ............................................................................................................. 7  
  1.2 References ......................................................................................................... 10  
    1.2.1 Normative References .................................................................................. 10  
    1.2.2 Informative References ............................................................................... 11  
  1.3 Overview ............................................................................................................ 11  
  1.4 Relationship to Other Protocols ......................................................................... 11  
  1.5 Prerequisites/Preconditions ............................................................................... 12  
  1.6 Applicability Statement ..................................................................................... 12  
  1.7 Versioning and Capability Negotiation .............................................................. 12  
  1.8 Vendor-Extensible Fields ................................................................................... 12  
  1.9 Standards Assignments ...................................................................................... 12  

2 Messages............................................................................................................... 13  
  2.1 Transport .......................................................................................................... 13  
  2.2 Message Syntax ................................................................................................ 13  
    2.2.1 Namespaces .................................................................................................. 13  
    2.2.2 HTTP Headers ............................................................................................ 13  
      2.2.2.1 X-MapiHttpCapability ......................................................................... 13  
      2.2.2.2 X-AnchorMailbox ............................................................................... 13  
      2.2.2.3 X-ClientCanHandle ............................................................................ 13  
    2.2.3 Autodiscover Request .................................................................................. 14  
      2.2.3.1 Autodiscover ....................................................................................... 14  
        2.2.3.1.1 Request .......................................................................................... 14  
        2.2.3.1.1.1 AcceptableResponseSchema .................................................... 14  
        2.2.3.1.1.2 EMailAddress .......................................................................... 14  
        2.2.3.1.1.3 LegacyDN .................................................................................. 14  
    2.2.4 Autodiscover Response ................................................................................ 15  
      2.2.4.1 Autodiscover ....................................................................................... 15  
        2.2.4.1.1 Response ...................................................................................... 15  
        2.2.4.1.1.1 User .......................................................................................... 15  
        2.2.4.1.1.1.1 AutoDiscoverSMTPAddress ................................................. 15  
        2.2.4.1.1.1.2 DefaultABView ..................................................................... 15  
        2.2.4.1.1.1.3 DeploymentId ........................................................................ 15  
        2.2.4.1.1.1.4 DisplayName ......................................................................... 15  
        2.2.4.1.1.1.5 LegacyDN .............................................................................. 16  
        2.2.4.1.1.2 Account ...................................................................................... 16  
        2.2.4.1.1.2.1 AccountType ......................................................................... 16  
        2.2.4.1.1.2.2 Action .................................................................................... 16  
        2.2.4.1.1.2.3 MicrosoftOnline .................................................................... 16  
        2.2.4.1.1.2.4 ConsumerMailbox ................................................................. 16  
        2.2.4.1.1.2.5 AlternativeMailbox ................................................................. 16  
        2.2.4.1.1.2.5.1 DisplayName ...................................................................... 17  
        2.2.4.1.1.2.5.2 LegacyDN ........................................................................... 17  
        2.2.4.1.1.2.5.3 Server .................................................................................. 17  
        2.2.4.1.1.2.5.4 SmtpAddress ....................................................................... 17  
        2.2.4.1.1.2.5.5 Type .................................................................................... 17  
      2.2.4.1.1.2.6 Protocol ...................................................................................... 18  
      2.2.4.1.1.2.6.1 AD ......................................................................................... 18  
      2.2.4.1.1.2.6.2 ASUrl .................................................................................... 18  
      2.2.4.1.1.2.6.3 AddressBook ........................................................................ 18  
      2.2.4.1.1.2.6.3.1 ExternalUrl ........................................................................ 18  
      2.2.4.1.1.2.6.3.2 InternalUrl ........................................................................ 19  
      2.2.4.1.1.2.6.4 AuthPackage ......................................................................... 19  
      2.2.4.1.1.2.6.5 AuthRequired ........................................................................ 19  


[MS-OXDSCLI] - v20210817  
Autodiscover Publishing and Lookup Protocol  
Copyright © 2021 Microsoft Corporation  
Release: August 17, 2021
3 Protocol Details

3.1 Client Details

3.1.1 Abstract Data Model
## 3 Protocol Examples

### 3.1 General

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1.2</td>
<td>Timers</td>
<td>31</td>
</tr>
<tr>
<td>3.1.3</td>
<td>Initialization</td>
<td>31</td>
</tr>
<tr>
<td>3.1.4</td>
<td>Higher-Layer Triggered Events</td>
<td>31</td>
</tr>
<tr>
<td>3.1.5</td>
<td>Message Processing Events and Sequencing Rules</td>
<td>31</td>
</tr>
<tr>
<td>3.1.5.1</td>
<td>Nonfunctional URIs</td>
<td>32</td>
</tr>
<tr>
<td>3.1.5.2</td>
<td>HTTP 302 Redirects</td>
<td>32</td>
</tr>
<tr>
<td>3.1.5.3</td>
<td>Autodiscover Redirect</td>
<td>32</td>
</tr>
<tr>
<td>3.1.5.4</td>
<td>Autodiscover Configuration Information</td>
<td>32</td>
</tr>
<tr>
<td>3.1.5.5</td>
<td>Autodiscover Server Errors</td>
<td>33</td>
</tr>
<tr>
<td>3.1.6</td>
<td>Timer Events</td>
<td>33</td>
</tr>
<tr>
<td>3.1.7</td>
<td>Other Local Events</td>
<td>33</td>
</tr>
</tbody>
</table>

### 3.2 Security

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.2.1</td>
<td>Abstract Data Model</td>
<td>33</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Timers</td>
<td>33</td>
</tr>
<tr>
<td>3.2.3</td>
<td>Initialization</td>
<td>33</td>
</tr>
<tr>
<td>3.2.4</td>
<td>Higher-Layer Triggered Events</td>
<td>33</td>
</tr>
<tr>
<td>3.2.5</td>
<td>Message Processing Events and Sequencing Rules</td>
<td>33</td>
</tr>
<tr>
<td>3.2.5.1</td>
<td>Processing the MapiHttpCapability Header</td>
<td>34</td>
</tr>
<tr>
<td>3.2.6</td>
<td>Timer Events</td>
<td>35</td>
</tr>
<tr>
<td>3.2.7</td>
<td>Other Local Events</td>
<td>35</td>
</tr>
</tbody>
</table>

## 4 Protocol Examples

### 4.1 Autodiscover Request

- Page 37

### 4.2 Autodiscover Redirect

- Page 37

### 4.3 Autodiscover Configuration

- Page 38

### 4.4 MapiHttp Response

- Page 39

### 4.5 Autodiscover Server Errors

- Page 40

## 5 Security

### 5.1 Security Considerations for Implementers

- Page 41

### 5.2 Index of Security Parameters

- Page 41

## 6 Appendix A: XSDs

### 6.1 Autodiscover Request XSD

- Page 42

### 6.2 Autodiscover Response XSD

- Page 42

### 6.3 Autodiscover Error Response XSD

- Page 45

### 6.4 Autodiscover Redirect Response XSD

- Page 46

## 7 Appendix B: Product Behavior

- Page 48

## 8 Change Tracking

- Page 51

## 9 Index

- Page 52
1 Introduction

The Autodiscover Publishing and Lookup Protocol is used by clients to retrieve URLs and settings that are needed to gain access to the web services that are offered by the server.

Sections 1.5, 1.8, 1.9, 2, and 3 of this specification are normative. All other sections and examples in this specification are informative.

1.1 Glossary

This document uses the following terms:

**Active Directory**: The Windows implementation of a general-purpose directory service, which uses LDAP as its primary access protocol. Active Directory stores information about a variety of objects in the network such as user accounts, computer accounts, groups, and all related credential information used by Kerberos [MS-KILE]. Active Directory is either deployed as Active Directory Domain Services (AD DS) or Active Directory Lightweight Directory Services (AD LDS), which are both described in [MS-ADOD]: Active Directory Protocols Overview.

**address book**: A collection of Address Book objects, each of which are contained in any number of address lists.

**authentication**: The act of proving an identity to a server while providing key material that binds the identity to subsequent communications.

**Autodiscover client**: A client that queries for a set of server locations where setup and configuration information for an [RFC2821]-compliant email address is stored.

**Autodiscover server**: A server in a managed environment that makes setup and configuration information available to Autodiscover clients. The location of Autodiscover servers is made available via the Autodiscover HTTP Service Protocol, as described in [MS-OXDISCO].

**calendar**: A date range that shows availability, meetings, and appointments for one or more users or resources. See also Calendar object.

**contact**: A person, company, or other entity that is stored in a directory and is associated with one or more unique identifiers and attributes, such as an Internet message address or login name.

**Contacts folder**: A Folder object that contains Contact objects.

**display name**: A text string that is used to identify a principal or other object in the user interface. Also referred to as title.

**distinguished name (DN)**: In the Active Directory directory service, the unique identifier of an object in Active Directory, as described in [MS-ADTS] and [RFC2251].

**domain**: A set of users and computers sharing a common namespace and management infrastructure. At least one computer member of the set must act as a domain controller (DC) and host a member list that identifies all members of the domain, as well as optionally hosting the Active Directory service. The domain controller provides authentication of members, creating a unit of trust for its members. Each domain has an identifier that is shared among its members. For more information, see [MS-AUTHSOD] section 1.1.1.5 and [MS-ADTS].

**Domain Name System (DNS)**: A hierarchical, distributed database that contains mappings of domain names to various types of data, such as IP addresses. DNS enables the location of computers and services by user-friendly names, and it also enables the discovery of other information stored in the database.

**email address**: A string that identifies a user and enables the user to receive Internet messages.
**endpoint**: A communication port that is exposed by an application server for a specific shared service and to which messages can be addressed.

**enterprise/site/server distinguished name (ESSDN)**: An X500 DN that identifies an entry in an abstract naming scheme that is separate from an **address book**. The naming scheme defines enterprises, which contain sites, and sites contain servers and users. There is no concrete data structure that embodies an ESSDN. Instead, an address book entry can contain an ESSDN as a property of the entry.

**Exchange Control Panel (ECP)**: A feature that enables end users to manage server options without the assistance of an administrator.

**fully qualified domain name (FQDN)**: In **Active Directory**, a fully qualified domain name (FQDN) that identifies a **domain**.

**Global Address List (GAL)**: An address list that conceptually represents the default address list for an **address book**.

**globally unique identifier (GUID)**: A term used interchangeably with universally unique identifier (UUID) in Microsoft protocol technical documents (TDs). Interchanging the usage of these terms does not imply or require a specific algorithm or mechanism to generate the value. Specifically, the use of this term does not imply or require that the algorithms described in [RFC4122] or [C706] must be used for generating the GUID. See also universally unique identifier (UUID).

**Hypertext Transfer Protocol (HTTP)**: An application-level protocol for distributed, collaborative, hypermedia information systems (text, graphic images, sound, video, and other multimedia files) on the World Wide Web.

**Hypertext Transfer Protocol Secure (HTTPS)**: An extension of HTTP that securely encrypts and decrypts web page requests. In some older protocols, "Hypertext Transfer Protocol over Secure Sockets Layer" is still used (Secure Sockets Layer has been deprecated). For more information, see [SSL3] and [RFC5246].

**Internet Message Access Protocol - Version 4 (IMAP4)**: A protocol that is used for accessing email and news items from mail servers, as described in [RFC3501].

**Lightweight Directory Access Protocol (LDAP)**: The primary access protocol for **Active Directory**. Lightweight Directory Access Protocol (LDAP) is an industry-standard protocol, established by the Internet Engineering Task Force (IETF), which allows users to query and update information in a directory service (DS), as described in [MS-ADTS]. The Lightweight Directory Access Protocol can be either version 2 [RFC1777] or version 3 [RFC3377].

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

**message store**: A unit of containment for a single hierarchy of Folder objects, such as a mailbox or public folders.

**offline address book (OAB)**: A collection of address lists that are stored in a format that a client can save and use locally.

**Out of Office (OOF)**: One of the possible values for the free/busy status on an appointment. It indicates that the user will not be in the office during the appointment.

**Post Office Protocol - Version 3 (POP3)**: A protocol that is used for accessing email from mail servers, as described in [RFC1939].

**public folder**: A Folder object that is stored in a location that is publicly available.
remote procedure call (RPC): A communication protocol used primarily between client and server. The term has three definitions that are often used interchangeably: a runtime environment providing for communication facilities between computers (the RPC runtime); a set of request-and-response message exchanges between computers (the RPC exchange); and the single message from an RPC exchange (the RPC message). For more information, see [C706].

Secure Sockets Layer (SSL): A security protocol that supports confidentiality and integrity of messages in client and server applications that communicate over open networks. SSL supports server and, optionally, client authentication using X.509 certificates [X509] and [RFC5280]. SSL is superseded by Transport Layer Security (TLS). TLS version 1.0 is based on SSL version 3.0 [SSL3].

Short Message Service (SMS): A communications protocol that is designed for sending text messages between mobile phones.

Simple Mail Transfer Protocol (SMTP): A member of the TCP/IP suite of protocols that is used to transport Internet messages, as described in [RFC5321].

site mailbox: A repository comprised of a mailbox and a web-based collaboration environment that is presented to users as a mailbox in an email client. A site mailbox uses team membership to determine which users have access to the repository.

Transport Layer Security (TLS): A security protocol that supports confidentiality and integrity of messages in client and server applications communicating over open networks. TLS supports server and, optionally, client authentication by using X.509 certificates (as specified in [X509]). TLS is standardized in the IETF TLS working group.

Uniform Resource Identifier (URI): A string that identifies a resource. The URI is an addressing mechanism defined in Internet Engineering Task Force (IETF) Uniform Resource Identifier (URI): Generic Syntax [RFC3986].

Uniform Resource Locator (URL): A string of characters in a standardized format that identifies a document or resource on the World Wide Web. The format is as specified in [RFC1738].

web server: A server computer that hosts websites and responds to requests from applications.

web service: A unit of application logic that provides data and services to other applications and can be called by using standard Internet transport protocols such as HTTP, Simple Mail Transfer Protocol (SMTP), or File Transfer Protocol (FTP). Web services can perform functions that range from simple requests to complicated business processes.

XML: The Extensible Markup Language, as described in [XML1.0].

XML namespace: A collection of names that is used to identify elements, types, and attributes in XML documents identified in a URI reference [RFC3986]. A combination of XML namespace and local name allows XML documents to use elements, types, and attributes that have the same names but come from different sources. For more information, see [XMLNS-2ED].

XML schema definition (XSD): The World Wide Web Consortium (W3C) standard language that is used in defining XML schemas. Schemas are useful for enforcing structure and constraining the types of data that can be used validly within other XML documents. XML schema definition refers to the fully specified and currently recommended standard for use in authoring XML schemas.

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.

[MS-N2HT] Microsoft Corporation, "Negotiate and Nego2 HTTP Authentication Protocol".


[MS-OCAUTHWS] Microsoft Corporation, "OC Authentication Web Service Protocol".

[MS-OFBA] Microsoft Corporation, "Office Forms Based Authentication Protocol".

[MS-OXABREF] Microsoft Corporation, "Address Book Name Service Provider Interface (NSPI) Referral Protocol".

[MS-OXCMAPIHTTP] Microsoft Corporation, "Messaging Application Programming Interface (MAPI) Extensions for HTTP".


[MS-OXDISCO] Microsoft Corporation, "Autodiscover HTTP Service Protocol".


[MS-OXWOAB] Microsoft Corporation, "Offline Address Book (OAB) Retrieval File Format".


[MS-OXWUMS] Microsoft Corporation, "Voice Mail Settings Web Service Protocol".

[MS-RPCH] Microsoft Corporation, "Remote Procedure Call over HTTP Protocol".


1.2.2 Informative References


1.3 Overview

This protocol is a set of methods, headers, and content types that extend HTTP version 1.1, as described in [RFC2616]. A list of possible Autodiscover server URIs is first discovered utilizing the Autodiscover HTTP Service Protocol, as described in [MS-OXDISCO]. This protocol enables Autodiscover clients to acquire email configuration settings for specific email addresses from the list of Autodiscover servers obtained from the Autodiscover HTTP Service Protocol.

This document specifies the following Autodiscover operations:

- A mechanism for Autodiscover clients to issue queries against Autodiscover servers.
- A mechanism for Autodiscover servers to send client configuration data to Autodiscover clients.
- A mechanism for Autodiscover servers to send referrals to Autodiscover clients.

1.4 Relationship to Other Protocols

This protocol and the Autodiscover HTTP Service Protocol described in [MS-OXDISCO] work together to use the standard HTTP mechanisms described in [RFC2068] to provide client management over the Internet. This protocol requires the Autodiscover HTTP Service Protocol to find Autodiscover servers that support this protocol. A higher-level protocol, given a server name or URL, uses this protocol to locate the corresponding fully qualified domain name (FQDN).

This protocol relies on HTTP 1.1, as described in [RFC2616]. It relies on HTTPS, as described in [RFC2818], for data protection services.

For conceptual background information and overviews of the relationships and interactions between this and other protocols, see [MS-OXPROTO].
1.5 Prerequisites/Preconditions
This protocol requires a web server that supports the HTTP POST command, as specified in [RFC2518] and [RFC2068].

This protocol also requires that Autodiscover clients have URIs that point to Autodiscover servers. Autodiscover clients can obtain these URIs by using the Autodiscover HTTP Service Protocol specified in [MS-OXDISCO].

This protocol assumes that the client has found the Autodiscover server via the Autodiscover HTTP Service Protocol, as specified in [MS-OXDISCO].

1.6 Applicability Statement
This protocol is used by a client to discover email configuration settings for a given email address.

1.7 Versioning and Capability Negotiation
Different versions of this protocol can be negotiated by using the AcceptableResponseSchema element, specified in section 2.2.3.1.1.1.

1.8 Vendor-Extensible Fields
Vendors MAY pass additional XML elements to Autodiscover clients from the Autodiscover server. To do so, the vendor SHOULD use a separate XML namespace and pass this in the AcceptableResponseSchema element, as specified in section 2.2.3.1.1.1.

1.9 Standards Assignments
None.
2 Messages

2.1 Transport

Messages are transported by using an HTTP POST command, as specified in [RFC2518] and [RFC2068].

This protocol SHOULD be used with Secure Sockets Layer (SSL)/Transport Layer Security (TLS), as specified in [RFC2246].

2.2 Message Syntax

All messages sent between the Autodiscover client and the Autodiscover server are XML messages.

2.2.1 Namespaces


2.2.2 HTTP Headers

This protocol defines the following HTTP headers, as specified in [RFC2616] section 4.2.

- X-MapiHttpCapability (section 2.2.2.1)
- X-AnchorMailbox (section 2.2.2.2)
- X-ClientCanHandle (section 2.2.2.3)

2.2.2.1 X-MapiHttpCapability

The X-MapiHttpCapability header is an optional header used in Autodiscover requests to indicate support for the Messaging Application Programming Interface (MAPI) Extensions for HTTP, as specified in [MS-OXCMAP|HTTP]. If present in a request, the value of this header MUST be an integer value greater than zero (0) that corresponds to the highest version of the response format for the Protocol element (section 2.2.4.1.1.2.6) supported by the client. See section 3.2.5.1 for more details on the available versions of the response format.

2.2.2.2 X-AnchorMailbox

The X-AnchorMailbox header identifies the email address of the account for which the configuration information will be retrieved. This header SHOULD be passed if the client sends the X-MapiHttpCapability header.

2.2.2.3 X-ClientCanHandle

The X-ClientCanHandle header contains a comma-delimited list of capabilities that the client supports. The following table specifies valid values for this header.
<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Negotiate&quot;</td>
<td>If this value is present, the server will return a value of &quot;negotiate&quot; in the AuthPackage element (section 2.2.4.1.1.2.6.4) if the server is configured to accept Negotiate authentication. If this value is not present, the server will not return a value of &quot;negotiate&quot; in the AuthPackage element.</td>
</tr>
<tr>
<td>&quot;ExHttpInfo&quot;</td>
<td>If this value is present, the server will return a Protocol element (section 2.2.4.1.1.2.6) with a Type element (section 2.2.4.1.1.2.6.46) set to &quot;EXHTTP&quot; if the server is configured to accept RPC/HTTP connections. If this value is not present, the server will not return a Protocol element with a Type element set to &quot;EXHTTP&quot;.</td>
</tr>
</tbody>
</table>

**2.2.3 Autodiscover Request**

Autodiscover requests consist of a single Autodiscover element (section 2.2.3.1), which contains information about the user within its child elements.

**2.2.3.1 Autodiscover**

The Autodiscover element is the root element of an Autodiscover request. The elements in the following sections can be child elements of the Autodiscover element.

**2.2.3.1.1 Request**

The Request element contains the request to the Autodiscover service. It is a required element of the Autodiscover element (section 2.2.3.1).

The AcceptableResponseSchema element and the EMailAddress or LegacyDN element MUST be child elements of the Request element.

**2.2.3.1.1.1 AcceptableResponseSchema**

The AcceptableResponseSchema element identifies the schema for an Autodiscover response.

Clients MUST include this element. The value MUST be the following: http://schemas.microsoft.com/exchange/autodiscover/outlook/responseschema/2006a.

**2.2.3.1.1.2 EMailAddress**

The EMailAddress element identifies the email address of the account for which the configuration information will be retrieved.

This element is an optional element for an Autodiscover request. If it is omitted, the LegacyDN MUST be present.

**2.2.3.1.1.3 LegacyDN**

The LegacyDN element identifies a user's mailbox by a legacy distinguished name (DN). The LegacyDN element is also known as the enterprise/site/server distinguished name (ESSDN), which is the naming scheme that defines the user.
The **LegacyDN** element is an optional element in the request. If it is omitted, the **EMailAddress** element MUST be present.

### 2.2.4 Autodiscover Response

Autodiscover responses consist of a single **Autodiscover** element (section 2.2.4.1), which contains configuration information for the user’s mailbox within its child elements.

### 2.2.4.1 Autodiscover

The **Autodiscover** element is the root element of an Autodiscover response. The elements in the following sections can be child elements of the **Autodiscover** element.

#### 2.2.4.1.1 Response

The **Response** element contains the response from the **Autodiscover server** that includes a list of URLs that are used to establish a connection with web services.

The elements specified in the following sections can be child elements of the **Response** element. For an example that shows the XML schema of the **Response** element and its child elements, see section 4.

#### 2.2.4.1.1.1 User

The **User** element and its child elements provide user-specific information. Servers MUST include this element if the server does not need to redirect the request and encounters no errors.

The elements specified in the following sections can be child elements of the **User** element.

#### 2.2.4.1.1.1.1 AutoDiscoverSMTPAddress

The **AutoDiscoverSMTPAddress** element represents the user’s primary Simple Mail Transfer Protocol (SMTP) address. It is a required child element of the **User** element (section 2.2.4.1.1.1). This is the email address that is used for the Autodiscover process. The **AutoDiscoverSMTPAddress** element returns the proxy address in lieu of the email address if a proxy address exists.

#### 2.2.4.1.1.1.2 DefaultABView

The **DefaultABView** element indicates the default view for the user's address book. If the **Global Address List (GAL)** is the default view, this element SHOULD NOT be present. If the Contacts folder in the user’s mailbox is the default view, this element SHOULD<3> be present with a value of "contacts".

#### 2.2.4.1.1.1.3 DeploymentId

The **DeploymentId** element uniquely identifies the server forest in a GUID format. It is a required child element of the **User** element (section 2.2.4.1.1.1).

The **DeploymentId** element is returned when the user is within a server forest. The returned value is the GUID identifier of the Active Directory forest in which the mailbox user account is contained.

#### 2.2.4.1.1.1.4 DisplayName

The **DisplayName** element represents the user's display name. It is a required child element of the **User** element (section 2.2.4.1.1.1).
2.2.4.1.1.1.5 LegacyDN

The LegacyDN element identifies a user's mailbox by DN. The LegacyDN is also known as the ESSDN, which is the naming scheme that defines the user. The LegacyDN element is a required child element of the User element (section 2.2.4.1.1).

2.2.4.1.1.2 Account

The Account element specifies account settings for the user.

The elements specified in the following sections can be child elements of the Account element.

2.2.4.1.1.2.1 AccountType

The AccountType element represents the account type. It is a required element of the Account element (section 2.2.4.1.1.2) if the server does not need to redirect the request. The value MUST be "email".

2.2.4.1.1.2.2 Action

The Action element provides information that is used to determine whether another Autodiscover request is required to return the user configuration information. It is a required child element of the Account element (section 2.2.4.1.1.2). The following table specifies valid values for this element.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;settings&quot;</td>
<td>The Autodiscover server has returned configuration settings in the Protocol element (section 2.2.4.1.1.2.6).</td>
</tr>
<tr>
<td>&quot;redirectAddr&quot;</td>
<td>The Autodiscover server has returned a RedirectAddr element (section 2.2.4.1.1.2.8).</td>
</tr>
<tr>
<td>&quot;redirectUrl&quot;</td>
<td>The Autodiscover server has returned a RedirectUrl element (section 2.2.4.1.1.2.9).</td>
</tr>
</tbody>
</table>

2.2.4.1.1.2.3 MicrosoftOnline

The MicrosoftOnline element specifies whether the user account is an online account. It is a required element of the Account element (section 2.2.4.1.1.2) when the value of the Action element (section 2.2.4.1.1.2.2) is "settings".<4> The value SHOULD be "False".

2.2.4.1.1.2.4 ConsumerMailbox

The ConsumerMailbox element specifies whether the user account is a consumer mailbox. It is a required element of the Account element (section 2.2.4.1.1.2) when the value of the Action element (section 2.2.4.1.1.2.2) is "settings".<5> The value SHOULD be "False".

2.2.4.1.1.2.5 AlternativeMailbox

The AlternativeMailbox element contains information that enables clients to open an additional mailbox. It is an optional child element of the Account element (section 2.2.4.1.1.2). <6>

The AlternativeMailbox element is returned only when an alternative mailbox is associated with the user.
The elements specified in the following sections can be child elements of the `AlternativeMailbox` element.

### 2.2.4.1.1.2.5.1 DisplayName

The `DisplayName` element represents the additional mailbox user's display name. It is a required child element of the `AlternativeMailbox` element (section 2.2.4.1.1.2.5). This string is used to override how a client will display the user's name in the alternative mailbox.<7>

### 2.2.4.1.1.2.5.2 LegacyDN

The `LegacyDN` element identifies the additional mailbox by DN. It is an optional child element of the `AlternativeMailbox` element (section 2.2.4.1.1.2.5). The `LegacyDN` is also known as the ESSDN, which is the naming scheme that defines the alternative user.<8>

The `LegacyDN` element MUST be present if the `SmtpAddress` element (section 2.2.4.1.1.2.5.4) is not present. The `LegacyDN` element MUST NOT be present if the `SmtpAddress` element is present.

### 2.2.4.1.1.2.5.3 Server

The `Server` element contains the FQDN of the mail server that contains the additional mailbox. It is an optional child element of the `AlternativeMailbox` element (section 2.2.4.1.1.2.5).<9>

The `Server` element MUST be present if the `SmtpAddress` element (section 2.2.4.1.1.2.5.4) is not present. The `Server` element MUST NOT be present if the `SmtpAddress` element is present.

### 2.2.4.1.1.2.5.4 SmtpAddress

The `SmtpAddress` element contains an SMTP address assigned to the alternative mailbox. This SMTP address can be used in the `EMailAddress` element (section 2.2.3.1.1.2) of an Autodiscover request to discover configuration settings for the alternative mailbox. It is an optional child element of the `AlternativeMailbox` element (section 2.2.4.1.1.2.5).<10>

The `SmtpAddress` element MUST be present if the `LegacyDN` element (section 2.2.4.1.1.2.5.2) and the `Server` element (section 2.2.4.1.1.2.5.3) are not present. The `SmtpAddress` element MUST NOT be present if the `LegacyDN` element and `Server` element are present.

### 2.2.4.1.1.2.5.5 Type

The `Type` element identifies the type of the additional mail account.<11> It is a required child element of the `AlternativeMailbox` element (section 2.2.4.1.1.2.5).

The following table specifies the possible values of the `Type` element.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Archive&quot;</td>
<td>The alternative mailbox represented by the parent <code>AlternativeMailbox</code> element is an archive mailbox for the user. An archive mailbox is a second mailbox provisioned for a user that is used to store historical messaging data.</td>
</tr>
<tr>
<td>&quot;Delegate&quot;</td>
<td>The alternative mailbox represented by the parent <code>AlternativeMailbox</code> element is owned by another user. The current user has permissions to open this mailbox.</td>
</tr>
<tr>
<td>&quot;TeamMailbox&quot;&lt;12&gt;</td>
<td>The alternative mailbox represented by the parent <code>AlternativeMailbox</code> element is a site mailbox that is</td>
</tr>
</tbody>
</table>
2.2.4.1.1.2.6 Protocol

The Protocol element contains the configuration information for connecting a client to the server.

The Protocol element is a required child element of the Account (section 2.2.4.1.1.2) element when the value of the Action element (section 2.2.4.1.1.2.2) is "settings". In this case, if the Protocol element contains information that the client can use to communicate with the mailbox via the Messaging Application Programming Interface (MAPI) Extensions for HTTP, as specified in [MS-OXCMAPIHTTP], it MUST contain the XML attributes listed in the following table.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>&quot;mapiHttp&quot;</td>
</tr>
<tr>
<td>Version</td>
<td>An integer value that MUST be greater than zero (0) and less than or equal to the value of the X-MapiHttpCapability header (section 2.2.2.1) included in the Autodiscover request.</td>
</tr>
</tbody>
</table>

The Protocol element is an optional child element of the External element (section 2.2.4.1.1.2.6.25) and the Internal element (section 2.2.4.1.1.2.6.27). The following sections describe elements that can be child elements of the Protocol element.

2.2.4.1.1.2.6.1 AD

The AD element specifies the Active Directory server used in conjunction with the mailbox. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6). The element contains the FQDN of a Lightweight Directory Access Protocol (LDAP) server that the client can connect to for directory information.

2.2.4.1.1.2.6.2 ASUrl

The ASUrl element specifies the URL of the best endpoint instance of Availability web services for an email-enabled user, as specified in [MS-OXWAVLS]. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6).

2.2.4.1.1.2.6.3 AddressBook

The AddressBook element contains information that the client can use to connect to an NSPI server via Messaging Application Programming Interface (MAPI) Extensions for HTTP, as specified in [MS-OXCMAPIHTTP], to retrieve address book information.

2.2.4.1.1.2.6.3.1 ExternalUrl

The ExternalUrl element contains a URL that the client can use to connect to an NSPI server via Messaging Application Programming Interface (MAPI) Extensions for HTTP when the client is located outside of the firewall.

2.2.4.1.1.2.6.3.2 InternalUrl
The **InternalUrl** element contains a URL that the client can use to connect to an NSPI server via Messaging Application Programming Interface (MAPI) Extensions for HTTP when the client is located inside of the firewall.

### 2.2.4.1.1.2.6.4 AuthPackage

The **AuthPackage** element specifies the **authentication** method that is used when authenticating to the server that contains the user’s **mailbox**. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6). The **AuthPackage** element is used only when the **Type** element (section 2.2.4.1.1.2.6.46) has a text value of "EXCH", "EXPR", or "EXHTTP".

The possible values are specified in the following table.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;basic&quot;</td>
<td>Indicates that the client SHOULD use basic authentication, as specified in [RFC2617].</td>
</tr>
<tr>
<td>&quot;kerb&quot;</td>
<td>Indicates that the client SHOULD use Kerberos authentication, as specified in [RFC4120].</td>
</tr>
<tr>
<td>&quot;kerbntlm&quot;</td>
<td>Indicates that the client SHOULD use Kerberos authentication or NTLM authentication, as specified in [MS-NLMP].</td>
</tr>
<tr>
<td>&quot;ntlm&quot;</td>
<td>Indicates that the client SHOULD use NTLM authentication.</td>
</tr>
<tr>
<td>&quot;certificate&quot;</td>
<td>Indicates that the client SHOULD use certificate authentication, as specified in [MS-OCAUTHWS].</td>
</tr>
<tr>
<td>&quot;negotiate&quot;&lt;13&gt;</td>
<td>Indicates that the client SHOULD use the <strong>Negotiate</strong> method for authentication, as specified in [MS-N2HT].</td>
</tr>
<tr>
<td>&quot;anonymous&quot;&lt;14&gt;</td>
<td>Indicates that the client SHOULD authenticate anonymously by using an <strong>SSL</strong> connection.</td>
</tr>
</tbody>
</table>

The **AuthPackage** element is returned only when there is an external mailbox server authentication method. If the **AuthPackage** element is omitted, the client SHOULD use Kerberos or NTLM authentication.

### 2.2.4.1.1.2.6.5 AuthRequired

The **AuthRequired** element specifies whether **authentication** is required. It MAY be an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6). The possible values are specified in the following table.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;on&quot;</td>
<td>Authentication is required by the server.</td>
</tr>
<tr>
<td>&quot;off&quot;</td>
<td>Authentication is not required by the server.</td>
</tr>
</tbody>
</table>

If a value is not specified, the default value is "on".

The **AuthRequired** element is returned only when the **Type** element (section 2.2.4.1.1.2.6.46) has a text value of "POP3".

### 2.2.4.1.1.2.6.6 CertPrincipalName
The **CertPrincipalName** element specifies the **SSL** certificate principal name that is required to connect to the server by using SSL. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6).

If the **CertPrincipalName** element is not specified, the default value is "msstd:SERVER", where "SERVER" is the value that is specified in the **Server** element (section 2.2.4.1.1.2.6.36). For example, if "SERVER" is specified as "server.Contoso.com" and **CertPrincipalName** is left blank with SSL turned on, the default value of **CertPrincipalName** would be "msstd:server.Contoso.com".

The **CertPrincipalName** element is returned only when the connection to the server is authenticated with SSL.

### 2.2.4.1.1.2.6.7 DomainName

The **DomainName** element specifies the user's **domain**. It MAY be an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6). If no value is specified, the default value is the **email address** in **user principal name (UPN)** format. For example: `<username>@<domain>`.

### 2.2.4.1.1.2.6.8 DomainRequired

The **DomainRequired** element contains a text value that indicates whether the domain is required for **authentication**. It MAY be an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6).

The possible values are specified in the following table.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;on&quot;</td>
<td>The domain name is required for authentication.</td>
</tr>
<tr>
<td>&quot;off&quot;</td>
<td>The domain name is not required for authentication.</td>
</tr>
</tbody>
</table>

### 2.2.4.1.1.2.6.9 EcpUrl

The **EcpUrl** element is the base **Exchange Control Panel (ECP) URL**. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6).<15> The URL contains the following information:

- **Protocol**: requires "https"
- **Host**: Host name
- **Path**: ECP path within the host server

The value of the **EcpUrl** element is similar to the following: "https://machine.domain.Contoso.com/ecp".

### 2.2.4.1.1.2.6.10 EcpUrl-aggr

The **EcpUrl-aggr** element contains a value that, when appended to the value of the **EcpUrl** element (section 2.2.4.1.1.2.6.9), results in a **URL** that can be used to access email aggregation settings. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6).<16> The value of the **EcpUrl-aggr** element is similar to the following: "?p=personalsettings/EmailSubscriptions.slab&exsvurl=1".

### 2.2.4.1.1.2.6.11 EcpUrl-extinstall

The **EcpUrl-extinstall** element contains a value that, when appended to the value of the **EcpUrl** element (section 2.2.4.1.1.2.6.9), results in a **URL** that can be used to view or change the mail add-
ins currently installed in the user’s mailbox. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6).<17>

The value of the EcpUrl-extinstall element is similar to the following:
"Extension/InstalledExtensions.slab?exsvurl=1&amp;realm=contoso.com".

### 2.2.4.1.1.2.6.12 EcpUrl-mt

The EcpUrl-mt element contains a value that, when appended to the value of the EcpUrl element (section 2.2.4.1.1.2.6.9), results in a URL that can be used to access email message tracking settings. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6).<18>

The value of the EcpUrl-mt element contains parameters contained within '<' and '>' characters that are substituted by the client as shown in the following table.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Substitute with</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsOWA</td>
<td>A string that specifies whether the call was invoked from Outlook Web App (OWA). 'y' is substituted if it was invoked from OWA; 'n' otherwise.</td>
</tr>
<tr>
<td>MsgID</td>
<td>Internet message identifier of the message to be tracked as specified by the Message-ID header. See [RFC2822].</td>
</tr>
<tr>
<td>Mbx</td>
<td>The SMTP address of the user’s mailbox.</td>
</tr>
<tr>
<td>Sender</td>
<td>The SMTP address of the message's sender.</td>
</tr>
</tbody>
</table>

The value of the EcpUrl-mt element is similar to the following:

### 2.2.4.1.1.2.6.13 EcpUrl-photo

The EcpUrl-photo element contains a value that, when appended to the value of the EcpUrl element (section 2.2.4.1.1.2.6.9), results in a URL that can be used to view or change the user's current photo. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6).<19>

The value of the EcpUrl-photo element is similar to the following:
"PersonalSettings/EditAccount.aspx?chgPhoto=1&amp;realm=contoso.com".

### 2.2.4.1.1.2.6.14 EcpUrl-publish

The EcpUrl-publish element contains a value that, when appended to the value of the EcpUrl element (section 2.2.4.1.1.2.6.9), results in a URL that can be used to access calendar publishing settings. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6).<20>

The value of the EcpUrl-publish element contains a parameter contained within '<' and '>' characters that are substituted by the client, as shown in the following table.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Substitute with</th>
</tr>
</thead>
<tbody>
<tr>
<td>FldID</td>
<td>The folder identifier to the calendar folder to be published.</td>
</tr>
</tbody>
</table>

The value of the EcpUrl-publish element is similar to the following:
"customize/calendarpublishing.slab?exsvurl=1&amp;FldID=<FldID>".
The **EcpUrl-ret** element contains a value that, when appended to the value of the **EcpUrl** element (section 2.2.4.1.1.2.6.9), results in a **URL** that can be used to access retention tag settings. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6).<21> The value of the **EcpUrl-ret** element is similar to the following: "?p=organize/retentionpolicytags.slab&exsvurl=1".

## 2.2.4.1.1.2.6.16 EcpUrl-sms

The **EcpUrl-sms** element contains a value that, when appended to the value of the **EcpUrl** element (section 2.2.4.1.1.2.6.9), results in a **URL** that can be used to access Short Message Service (SMS) settings. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6).<22> The value of the **EcpUrl-sms** element is similar to the following: "?p=sms/textmessaging.slab&exsvurl=1".

## 2.2.4.1.1.2.6.17 EcpUrl-tm

The **EcpUrl-tm** element contains a value that, when appended to the value of the **EcpUrl** element (section 2.2.4.1.1.2.6.9), results in a **URL** that can be used to access a list of all **site mailboxes** of which the user is currently a member. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6).<23>

The value of the **EcpUrl-tm** element is similar to the following: "?ftr=TeamMailbox&realm=contoso.com".

## 2.2.4.1.1.2.6.18 EcpUrl-tmCreating

The **EcpUrl-tmCreating** element contains a value that, when appended to the value of the **EcpUrl** element (section 2.2.4.1.1.2.6.9), results in a **URL** that can be used to create a new **site mailbox**. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6).<24>

The value of the **EcpUrl-tmCreating** element contains parameters contained within '<' and '>' characters that are substituted by the client, as shown in the following table.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Substitute with</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPUrl</td>
<td>The URL to create a new site mailbox.</td>
</tr>
<tr>
<td>Title</td>
<td>The title used to create a new site mailbox.</td>
</tr>
</tbody>
</table>

The value of the **EcpUrl-tmCreating** element is similar to the following: "?ftr=TeamMailboxCreating&SPUrl=<SPUrl>&Title=<Title>&realm=contoso.com".

## 2.2.4.1.1.2.6.19 EcpUrl-tmEditing

The **EcpUrl-tmEditing** element contains a value that, when appended to the value of the **EcpUrl** element (section 2.2.4.1.1.2.6.9), results in a **URL** that can be used to edit an existing **site mailbox**. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6).<25>

The value of the **EcpUrl-tmEditing** element contains a parameter contained within '<' and '>' characters that is substituted by the client, as shown in the following table.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Substitute with</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id</td>
<td>The <strong>SMTP email address</strong> or the <strong>ESSDN</strong> assigned to the site mailbox.</td>
</tr>
</tbody>
</table>

The value of the **EcpUrl-tmEditing** element is similar to the following: "?ftr=TeamMailboxEditing&Id=<Id>&realm=contoso.com".
2.2.4.1.1.2.6.20  EcpUrl-tmHiding

The EcpUrl-tmHiding element contains a value that, when appended to the value of the EcpUrl element (section 2.2.4.1.1.2.6.9), results in a URL that can be used to unsubscribe the user from a site mailbox. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6). The value of the EcpUrl-tmHiding element contains a parameter contained within '<' and '>' characters that is substituted by the client, as shown in the following table.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Substitute with</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id</td>
<td>The SMTP email address or the ESSDN assigned to the site mailbox.</td>
</tr>
</tbody>
</table>

The value of the EcpUrl-tmHiding element is similar to the following: "?ftr=TeamMailboxHiding&Id=<Id>&realm=contoso.com".

2.2.4.1.1.2.6.21  EcpUrl-um

The EcpUrl-um element contains a value that, when appended to the value of the EcpUrl element (section 2.2.4.1.1.2.6.9), results in a URL that can be used to access voice mail settings. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6). The value of the EcpUrl-um element is similar to the following: "?p=customize/voicemail.aspx&exsvurl=1".

2.2.4.1.1.2.6.22  Encryption

The Encryption element specifies the required encryption for the connection to the server. It MAY be an optional child element of the Protocol element (section 2.2.4.1.1.2.6). This element is valid only if the value of the Type element (section 2.2.4.1.1.2.6.46) is "IMAP", "POP3", or "SMTP". If the Encryption element is present, it overrides the SSL element (section 2.2.4.1.1.2.6.44). The following table specifies the possible values of the Encryption element.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;None&quot;</td>
<td>No encryption is used.</td>
</tr>
<tr>
<td>&quot;SSL&quot;</td>
<td>SSL encryption is used.</td>
</tr>
<tr>
<td>&quot;TLS&quot;</td>
<td>TLS encryption is used.</td>
</tr>
<tr>
<td>&quot;Auto&quot;</td>
<td>The most secure encryption that the client and server support is used.</td>
</tr>
</tbody>
</table>

2.2.4.1.1.2.6.23  EmwsUrl

The EmwsUrl element specifies the URL for the management web services virtual directory. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6).

2.2.4.1.1.2.6.24  EwsUrl

The EwsUrl element specifies the URL for the web services virtual directory. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6).

2.2.4.1.1.2.6.25  External
The **External** element contains the collection of **URLs** that a client can connect to outside the firewall. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6). If the server is configured for external access, the **External** element will contain a **Protocol** element (section 2.2.4.1.1.2.6) and an **OWAUrl** element (section 2.2.4.1.1.2.6.27.1). The **Protocol** element SHOULD contain an **ASUrl** element (section 2.2.4.1.1.2.6.2) and a **Type** element (section 2.2.4.1.1.2.6.46). The **Protocol** element SHOULD NOT contain any other child elements.

### 2.2.4.1.1.2.6.26 GroupingInformation

The **GroupingInformation** element specifies the grouping hint for certain clients. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6). <28>

### 2.2.4.1.1.2.6.27 Internal

The **Internal** element contains a collection of **URLs** that a client can connect to when it is inside the firewall. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6).

If the server is configured for internal access, the **Internal** element contains a **Protocol** element, (as specified in section 2.2.4.1.1.2.6) and an **OWAUrl** element (as specified in section 2.2.4.1.1.2.6.27.1). The **Protocol** child element SHOULD contain an **ASUrl** element (as specified in section 2.2.4.1.1.2.6.2) and a **Type** element (as specified in section 2.2.4.1.1.2.6.46). The **Protocol** child element SHOULD NOT contain any other child elements.

### 2.2.4.1.1.2.6.27.1 OWAUrl

The **OWAUrl** element describes the **URL**, as specified in [RFC3986], and the **authentication** method that is used to access the server. It is a required child element of the **Internal** element (section 2.2.4.1.1.2.6.27) and the **External** element (section 2.2.4.1.1.2.6.25).

The **OWAUrl** has a required **AuthenticationMethod** attribute. This attribute specifies the allowed authentication methods that are supported by the server. This attribute can be one or more of the values in the following table. Multiple values are separated by commas.

<table>
<thead>
<tr>
<th>Value</th>
<th>Authentication method</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;WindowsIntegrated&quot;</td>
<td>Integrated Windows Authentication, as specified in [MS-OCAUTHWS].</td>
</tr>
<tr>
<td>&quot;Fba&quot;</td>
<td>Forms Based Authentication, as specified in [MS-OFBA].</td>
</tr>
<tr>
<td>&quot;Ntlm&quot;</td>
<td>NTLM Authentication, as specified in [MS-NLMP].</td>
</tr>
<tr>
<td>&quot;Digest&quot;</td>
<td>Digest Authentication, as specified in [RFC2617].</td>
</tr>
<tr>
<td>&quot;Basic&quot;</td>
<td>Basic Authentication, as specified in [RFC2617].</td>
</tr>
<tr>
<td>&quot;LiveIdFba&quot;&lt;29&gt;</td>
<td>Live Id Authentication, as specified in [MS-OCAUTHWS].</td>
</tr>
</tbody>
</table>

### 2.2.4.1.1.2.6.28 LoginName

The **LoginName** element specifies the user's mail server logon name. It MAY be an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6).

### 2.2.4.1.1.2.6.29 MailStore
The **MailStore** element contains information that the client can use to connect to a **mailbox** via Messaging Application Programming Interface (MAPI) Extensions for HTTP, as specified in [MS-OXCMAP HTTP], to retrieve mailbox information.

### 2.2.4.1.1.2.6.29.1 ExternalUrl

The **ExternalUrl** element contains a URL that the client can use to connect to a **mailbox** via Messaging Application Programming Interface (MAPI) Extensions for HTTP when the client is located outside of the firewall.

### 2.2.4.1.1.2.6.29.2 InternalUrl

The **InternalUrl** element contains a URL that the client can use to connect to a **mailbox** via Messaging Application Programming Interface (MAPI) Extensions for HTTP when the client is located inside of the firewall.

### 2.2.4.1.1.2.6.30 MdbDN

The **MdbDN** element contains the DN of the **mailbox** database. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6).

### 2.2.4.1.1.2.6.31 OABUrl

The **OABUrl** element specifies the offline address book (OAB) configuration server URL for a server. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6). For more details about the services that are available at this URL, see [MS-OXWOAB].

The **OABUrl** element is returned if there is an internal or external OAB configured for the user.

### 2.2.4.1.1.2.6.32 OOFUrl

The **OOFUrl** element specifies the URL of the best instance of the Out of Office (OOF) Web Service for a mail-enabled user. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6). For more details about the services that are available at this URL, see [MS-OXWOOF].

The **OOFUrl** element is returned when the server implements a URL for internal or external access to the Out of Office (OOF) Web Service. If the **OOFUrl** element is omitted, the **Out of Office (OOF)** services are not available to the client.

### 2.2.4.1.1.2.6.33 Port

The **Port** element specifies the port that is used to connect to the message store. It MAY be an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6). For more details, see [MS-OXCRPC].

The **Port** element is not returned when the **Server** element contains a URL.

### 2.2.4.1.1.2.6.34 PublicFolderServer

The **PublicFolderServer** element specifies the FQDN for the public folder server. It is an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6).

### 2.2.4.1.1.2.6.35 ReferralPort

The **ReferralPort** element specifies the port that is used to get a referral to a directory. It MAY be an optional child element of the **Protocol** element (section 2.2.4.1.1.2.6). For more details, see [MS-OXABREF].
2.2.4.1.1.2.6.36 Server

The Server element specifies the name of the mail server. It is a required child element of the Protocol element (section 2.2.4.1.1.2.6) that has a Type element (section 2.2.4.1.1.2.6.46) value of "EXCH", "EXPR", "EXHTTP", "POP3", "SMTP", or "IMAP". The value will be either a host name or an IP address.

2.2.4.1.1.2.6.37 ServerDN

The ServerDN element specifies the DN of the mail server. It is a required child element of the Protocol element (section 2.2.4.1.1.2.6) when the Type element (section 2.2.4.1.1.2.6.46) has a value of "EXCH".

2.2.4.1.1.2.6.38 ServerExclusiveConnect

The ServerExclusiveConnect element specifies whether the client uses the connection information contained in the parent Protocol element (section 2.2.4.1.1.2.6) first when the client attempts to connect to the server. It is an optional child element of the Protocol element. The possible values are specified in the following table.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;on&quot;</td>
<td>Clients SHOULD use the connection information in the parent Protocol element first when attempting to connect to the server.</td>
</tr>
<tr>
<td>&quot;off&quot;</td>
<td>Clients SHOULD NOT use the connection information in the parent Protocol element first when attempting to connect to the server unless there are no other Protocol elements that contain a ServerExclusiveConnect element with a value of &quot;on&quot;.</td>
</tr>
</tbody>
</table>

If the element is not present, the default value is "off". If no Protocol elements in the response have a ServerExclusiveConnect element set to "on", the client can use the Protocol elements in any order.

The ServerExclusiveConnect element is used only when the Type element (section 2.2.4.1.1.2.6.46) is equal to "EXPR", "EXCH", or "EXHTTP".

2.2.4.1.1.2.6.39 ServerVersion

The ServerVersion element represents the version number of the server software. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6).

The ServerVersion value is a 32-bit hexadecimal number that contains the major version number, minor version number, and major build number of the server. The ServerVersion element is used only when the Type element (section 2.2.4.1.1.2.6.46) has a value of "EXCH".

2.2.4.1.1.2.6.40 SharingUrl

The SharingUrl element specifies the endpoint for a sharing server, which is a server used for sharing calendars and contacts. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6).

The SharingUrl element is returned when the server implements a URL for cross-organization sharing.
2.2.4.1.1.2.6.41 SiteMailboxCreationURL

The SiteMailboxCreationURL element contains a URL to a self-service web site that can be used to create a new site mailbox. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6).<31>

2.2.4.1.1.2.6.42 SMTPLast

The SMTPLast element specifies whether the Simple Mail Transfer Protocol (SMTP) server requires that email be downloaded before it sends email by using the SMTP server. It MAY be an optional child element of the Protocol element (section 2.2.4.1.2.6).

The possible values are specified in the following table.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;on&quot;</td>
<td>The server requires that email be downloaded before the client sends mail via SMTP.</td>
</tr>
<tr>
<td>&quot;off&quot;</td>
<td>The server does not require that email be downloaded before the client sends mail via SMTP.</td>
</tr>
</tbody>
</table>

If this element is not present, the default value is "off".

The SMTPLast element is used only when the Type element (section 2.2.4.1.1.2.6.46) is equal to "SMTP".

2.2.4.1.1.2.6.43 SPA

The SPA element indicates whether secure password authentication is required. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6). This element is only valid when the value of the Type element (section 2.2.4.1.1.2.6.46) is "SMTP", "POP3", or "IMAP". The possible values are specified in the following table.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;on&quot;</td>
<td>SPA is required.</td>
</tr>
<tr>
<td>&quot;off&quot;</td>
<td>SPA is not required.</td>
</tr>
</tbody>
</table>

If this element is not present, the default value is "on".

2.2.4.1.1.2.6.44 SSL

The SSL element specifies whether the server requires SSL for logon. It is an optional child element of the Protocol element (section 2.2.4.1.1.2.6).

The possible values are specified in the following table.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;on&quot;</td>
<td>SSL is required.</td>
</tr>
<tr>
<td>&quot;off&quot;</td>
<td>SSL is not required.</td>
</tr>
</tbody>
</table>

If a value is not specified, the default value is "on".

2.2.4.1.1.2.6.45 TTL
The **TTL** element specifies the time, in hours, during which the settings remain valid. It is an optional child element of the **Protocol** element (section 2.2.4.1.2.6).

A value of "0" (zero) indicates that rediscovery is not required. If the **TTL** element is omitted, the default value is "1".

### 2.2.4.1.2.6.46 Type

The **Type** element identifies the type of the configured mail account. It is an optional child element of the **Protocol** element (section 2.2.4.1.2.6). If the **Protocol** element has a Type attribute, then the **Type** element MUST NOT be present. If the **Protocol** element does not have a Type attribute, then the **Type** element MUST be present. The possible values are specified in the following table.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;EXCH&quot;</td>
<td>The <strong>Protocol</strong> element contains information that the Autodiscover client can use to communicate with the mailbox via a remote procedure call (RPC). For details, see [MS-OXCRPC].</td>
</tr>
<tr>
<td>&quot;EXPR&quot;</td>
<td>The <strong>Protocol</strong> element contains information that the Autodiscover client can use to communicate when outside the firewall, including RPC/HTTP connections. For details, see [MS-RPCH].</td>
</tr>
<tr>
<td>&quot;EXHTTP&quot;&lt;32&gt;</td>
<td>The <strong>Protocol</strong> element contains information that the Autodiscover client can use to communicate via RPC/HTTP connections.</td>
</tr>
<tr>
<td>&quot;POP3&quot;&lt;33&gt;</td>
<td>The <strong>Protocol</strong> element contains settings that the client can use to communicate with the mail server via <strong>POP3</strong>. For details, see [RFC1939].</td>
</tr>
<tr>
<td>&quot;SMTP&quot;&lt;34&gt;</td>
<td>The <strong>Protocol</strong> element contains settings the client can use to send mail via <strong>SMTP</strong>. For details, see [RFC2821].</td>
</tr>
<tr>
<td>&quot;IMAP&quot;&lt;35&gt;</td>
<td>The <strong>Protocol</strong> element contains settings the client can use to communicate with the mail server via <strong>IMAP4</strong>. For details, see [RFC3501].</td>
</tr>
<tr>
<td>&quot;DAV&quot;&lt;36&gt;</td>
<td>The <strong>Protocol</strong> element contains settings the client can use to communicate with the mail server via the DAV protocol. For details, see [RFC2518].</td>
</tr>
<tr>
<td>&quot;WEB&quot;</td>
<td>The <strong>Protocol</strong> element contains settings the client can use to connect via a web browser.</td>
</tr>
</tbody>
</table>

### 2.2.4.1.2.6.47 UMUrl

The **UMUrl** element specifies the **URL** of the best instance of the Voice Mail Settings Web Service protocol ([MS-OWWUMS]) for a mail-enabled user. It is an optional child element of the **Protocol** element (section 2.2.4.1.2.6).

The **UMUrl** element is returned when the server implements a URL for internal or external access to the Voice Mail Settings Web Service.

### 2.2.4.1.2.6.48 UsePOPAuth

The **UsePOPAuth** element indicates whether the authentication information that is provided for a **POP3** type of account is also used for **SMTP**. It MAY be an optional child element of the **Protocol** element (section 2.2.4.1.2.6).

The possible values are specified in the following table.
<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;on&quot;</td>
<td>Use the POP3 authentication information for SMTP.</td>
</tr>
<tr>
<td>&quot;off&quot;</td>
<td>Do not use the POP3 authentication information for SMTP.</td>
</tr>
</tbody>
</table>

The **UsePOPAuth** element is used only when the value of the **Type** element (section 2.2.4.1.1.2.6.46) is equal to "SMTP".

### 2.2.4.1.1.2.7 PublicFolderInformation

The **PublicFolderInformation** element contains information that enables clients to send an Autodiscover request to discover public folder settings. It is an optional child element of the **Account** element (section 2.2.4.1.1.2). There MUST NOT be more than one **PublicFolderInformation** element in a response.

The elements specified in the following sections can be child elements of the **PublicFolderInformation** element.

### 2.2.4.1.1.2.7.1 SmtpAddress

The **SmtpAddress** element contains an SMTP address assigned to the public folder message store configured for the user. This SMTP address can be used in the **EmailAddress** element (section 2.2.3.1.1.2) of an Autodiscover request to discover public folder settings. It is a required child element of the **PublicFolderInformation** element (section 2.2.4.1.1.2.7).

### 2.2.4.1.1.2.8 RedirectAddr

The **RedirectAddr** element specifies the email address to use for a subsequent Autodiscover request. It is a required child element of the **Account** element (section 2.2.4.1.1.2) when the value of the **Action** element (section 2.2.4.1.1.2.2) is "redirectAddr".

The **RedirectAddr** element is returned when the server requires another email address to perform another Autodiscover request.

### 2.2.4.1.1.2.9 RedirectUrl

The **RedirectUrl** element specifies the URL of the server to use for a subsequent Autodiscover request. It is a required child element of the **Account** element (section 2.2.4.1.1.2) when the value of the **Action** element (section 2.2.4.1.1.2.2) is "redirectUrl".

The **RedirectUrl** element is returned when the server requires another URL to perform another Autodiscover request.

### 2.2.4.1.1.3 Error

The **Error** element contains an Autodiscover error response. It is an optional child element of the **Response** element (section 2.2.4.1.1). The **Error** element has two attributes, as listed in the following table.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>Represents the time when the error response was returned.</td>
</tr>
<tr>
<td>Id</td>
<td>Represents a hash value of the name of the mail server.</td>
</tr>
</tbody>
</table>

The elements specified in the following sections can be child elements of the **Error** element.
2.2.4.1.1.3.1 DebugData

The **DebugData** element contains the debug data for an Autodiscover error response. It is a required child element of the **Error** element (section 2.2.4.1.1.3). The contents of this element will depend on the implementation of the Autodiscover server.

2.2.4.1.1.3.2 ErrorCode

The **ErrorCode** element contains the error code for an error Autodiscover response. It is a required child element of the **Error** element (section 2.2.4.1.1.3).

The following table lists the current error codes.

<table>
<thead>
<tr>
<th>Error code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>The email address cannot be found. The Autodiscover server cannot determine how to provide configuration information for the requested email address.</td>
</tr>
<tr>
<td>501</td>
<td>Bad Address. The Autodiscover server recognizes the given email address but is unable to provide configuration information because the given email address has no configuration options.</td>
</tr>
<tr>
<td>600</td>
<td>Invalid Request. The XML request was improperly formatted.</td>
</tr>
<tr>
<td>601</td>
<td>The Autodiscover server was unable to provide configuration information of the requested type.</td>
</tr>
<tr>
<td>602</td>
<td>Bad Address. The Autodiscover server recognizes the specified email address but is unable to provide configuration information because of configuration errors.</td>
</tr>
<tr>
<td>603</td>
<td>The Autodiscover server threw an internal error.</td>
</tr>
</tbody>
</table>

2.2.4.1.1.3.3 Message

The **Message** element contains the error message for an error Autodiscover response. It is a required child element of the **Error** element (section 2.2.4.1.1.3). The **Message** element SHOULD be in the form of a human-readable error message.
3 Protocol Details

3.1 Client Details

3.1.1 Abstract Data Model
None.

3.1.2 Timers
Clients SHOULD implement a Time-To-Live timer, initialized to the number of hours specified by the value of the TTL element (section 2.2.4.1.1.2.6.45) in the Autodiscover response. If the value of the TTL element is "0", this timer is not used. If the TTL element is absent, this timer SHOULD be initialized to one hour.

3.1.3 Initialization
It is assumed that the Autodiscover client has an email address for which discovery information is needed.

It is also assumed that the Autodiscover client has a list of potential Autodiscover server URIs. This list could be generated by using the [MS-OXDISCO] protocol. The list could also be preconfigured.

3.1.4 Higher-Layer Triggered Events
When an Autodiscover client is configuring itself to access a user's mailbox, it sends an HTTP POST request that contains an Autodiscover request, as specified in section 2.2.3. The client waits for a response and processes the response as specified in section 3.1.5.

If the client supports the Messaging Application Programming Interface (MAPI) Extensions for HTTP, as specified in [MS-OXCMAPIHTTP], it SHOULD include an X-MapiHttpCapability header (section 2.2.2.1) in the Autodiscover request. If the client does not support the Messaging Application Programming Interface (MAPI) Extensions for HTTP, it MUST NOT include an X-MapiHttpCapability header in the Autodiscover request.

3.1.5 Message Processing Events and Sequencing Rules
The results of an Autodiscover request fall into the following categories.

- The URI is not functional. The client SHOULD process this response as specified in section 3.1.5.1.
- The HTTP POST command returns an HTTP 302 Redirection response. The client SHOULD process this response as specified in section 3.1.5.2.
- The Autodiscover server returns an Action element (section 2.2.4.1.1.2.2) with a value of "redirectAddr" or "redirectUrl". The client SHOULD process this response as specified in section 3.1.5.3.
- The Autodiscover server returns configuration information. The client SHOULD process this response as specified in section 3.1.5.4.
- The Autodiscover server returns error information. The client SHOULD process this response as specified in section 3.1.5.5.
3.1.5.1 Nonfunctional URIs

If the Autodiscover client attempts to send an HTTP POST request to a nonfunctional URI, it SHOULD retry the HTTP POST request using the next URI in its list of potential Autodiscover server URIs. It SHOULD NOT abort the Autodiscover request unless it has attempted all of the URIs in its list of potential Autodiscover server URIs.

3.1.5.2 HTTP 302 Redirects

If the server returns a redirection URL via an HTTP 302 Redirect response, the client SHOULD repost the request to the redirection URL contained in the Location header (as specified in [RFC2068] section 14.30) of the response.

3.1.5.3 Autodiscover Redirect

If the server returns an Autodiscover response (as specified in section 2.2.4) which contains an Action element (section 2.2.4.1.1.2.2) with a value of "redirectAddr", the client SHOULD send a new Autodiscover request. The value of the EMailAddress element (section 2.2.3.1.1.2) in the new request SHOULD be set to the value of the RedirectAddr element (section 2.2.4.1.1.2.8) in the Autodiscover response.

If the server returns an Autodiscover response which contains an Action element with a value of "redirectUrl", the client SHOULD send a new Autodiscover request to the URL contained in the value of the RedirectUrl element (section 2.2.4.1.1.2.9) in the Autodiscover response.

See section 4.2 for an example of an Autodiscover Redirect response.

3.1.5.4 Autodiscover Configuration Information

If the server returns an Autodiscover response (as specified in section 2.2.4) that contains a User element (section 2.2.4.1.1.1) and an Account element (section 2.2.4.1.1.2), the client SHOULD use the information contained within the response to configure itself. It SHOULD NOT send further Autodiscover requests to the next URI in its list of potential Autodiscover server URIs. For an example of an Autodiscover response that contains configuration information, see section 4.3.

If the server response contains multiple Protocol elements (section 2.2.4.1.1.2.6), the client uses the following rules to choose which Protocol element to use to connect.

1. If the server response contains a Protocol element that contains a ServerExclusiveConnect element (section 2.2.4.1.1.2.6.38) with a value of "on", the configuration information in that Protocol element SHOULD <38> be used first.

2. If the server response contains one or more Protocol elements that contain a Type element (section 2.2.4.1.1.2.6.46) with a value of "EXHTTP", the client SHOULD <39> ignore any Protocol elements that contain a Type element with a value of "EXPR".

3. If there are multiple Protocol elements that contain a Type element with a value of "EXHTTP", the client SHOULD <40> store each set of configuration information represented by these elements, and attempt to connect using the configuration information sets in the order in which they appeared in the response, stopping when a successful connection is made. For example, if a response contains two Protocol elements that contain a Type element with the value "EXHTTP", the client attempts to connect using the information in the first such Protocol element. If the connection attempt fails, the client attempts to connect using the information in the second such Protocol element.
3.1.5.5 Autodiscover Server Errors

If the server returns an Autodiscover response (as specified in section 2.2.4) that contains an Error element (section 2.2.4.1.1.3), the client SHOULD retry the HTTP POST request using the next URI in its list of potential Autodiscover server URIs. For an example of an Autodiscover response that contains an Error element, see section 4.5.

3.1.6 Timer Events

When the Time-To-Live timer specified in section 3.1.2 expires, clients SHOULD issue a new Autodiscover request and apply any changes indicated by the response to its configuration.

3.1.7 Other Local Events

None.

3.2 Server Details

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

An Autodiscover server MUST respond to HTTP POST requests to the URL "https://<Server>/autodiscover/autodiscover.xml", where "<Server>" is a valid host name for the server.

The server SHOULD validate the body of the HTTP POST request, ensuring that it is a valid Autodiscover request as specified in section 2.2.3.

If the server receives a request that contains both the EMailAddress element (section 2.2.3.1.1.2) and the LegacyDN element (section 2.2.3.1.1.3), the value of the LegacyDN element MUST be used.

If the server needs to redirect the Autodiscover client to another URL, it SHOULD send a 302 Redirect response with the Location header set to the new URL. Alternatively, it MAY send an Autodiscover response (as specified in section 2.2.4) with a RedirectUrl element (section 2.2.4.1.1.2.9) value set to the new URL.

If the server needs to redirect the Autodiscover client to another email address, it SHOULD send an Autodiscover response with a RedirectAddr element (section 2.2.4.1.1.2.8) value set to the new email address.
If the server encounters an error, it SHOULD send an Autodiscover response with an Error element (section 2.2.4.1.1.3). It SHOULD set the value of the ErrorCode element to one of the values in the table in section 2.2.4.1.3.2, but MAY use a value not in the table.

If the server does not need to redirect the request and encounters no errors, it MUST return an Autodiscover response with a User element (section 2.2.4.1.1.1) containing information about the user represented by the email address in the EmailAddress element (section 2.2.3.1.1.2) of the request and an Account element (section 2.2.4.1.1.2) containing configuration information for the user's mailbox.

If the Autodiscover request includes an X-MapiHttpCapability header (section 2.2.2.1), the server SHOULD perform the additional processing specified in section 3.2.5.1.

3.2.5.1 Processing the X-MapiHttpCapability Header

If the Autodiscover request contains an X-MapiHttpCapability header (section 2.2.2.1), the server validates the value of the header. The value is considered valid if it is an integer value greater than zero (0) and if the server supports a version of the response format for the Protocol element (section 2.2.4.1.1.2.6) less than or equal to the value.

If the value of the X-MapiHttpCapability header is invalid, the server responds as if the X-MapiHttpCapability header was not present.

If the value of the X-MapiHttpCapability header is valid, or if the server deduces the client's MapiHttp capability based on the user agent header, the server SHOULD modify the Autodiscover response according to the following requirements.

1. The server determines the highest version of the response format for the Protocol element that it supports that is less than or equal to the value of the X-MapiHttpCapability header in the Autodiscover response.

2. The response MUST include a Protocol element that contains a Type attribute set to "mapiHttp" and a Version attribute, as specified in section 2.2.4.1.1.2.6, that corresponds to the highest version determined in step 1. The child elements of the Protocol element MUST conform to the version, as indicated in the table below.

3. The response MUST NOT include a Protocol element that contains a Type element (section 2.2.4.1.1.2.6.46) set to "EXCH" or "EXPR".

The possible versions for the response format are specified in the following table.

<table>
<thead>
<tr>
<th>Version</th>
<th>Response Format</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>&lt;MailStore&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;InternalUrl&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>...&lt;/InternalUrl&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;ExternalUrl&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>...&lt;/ExternalUrl&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;/MailStore&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;AddressBook&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;InternalUrl&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>...&lt;/InternalUrl&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;ExternalUrl&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>...&lt;/ExternalUrl&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;/AddressBook&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>For both the MailStore element (section 2.2.4.1.1.2.6.29) and the AddressBook element (section 2.2.4.1.1.2.6.3), there MUST be at least one child element.</code></td>
<td></td>
</tr>
</tbody>
</table>

If the Autodiscover request contains an X-MapiHttpCapability header, the X-AnchorMailbox header (section 2.2.2.2) SHOULD also be sent. If this header is not sent, the server does not fail but the response can reflect an incorrect state of accessing the mailbox.
3.2.6 Timer Events

None.

3.2.7 Other Local Events

None.
4 Protocol Examples

The following topology is used in this example and is illustrated in the following diagram:

- The **Domain Name System (DNS)** name of the mail server is mail.contoso.com.
- The DNS name of the Web service computer is webservice.contoso.com. It has a valid SSL certificate.
- Autodiscover **web services** are available at https://webservice.contoso.com/autodiscover/autodiscover.xml.

![Figure 1: Client and server topology](image)

From executing the Autodiscover HTTP Service Protocol [MS-OXDISCO], the client has the following list of possible **Autodiscover servers**:

- https://contoso.com/autodiscover/autodiscover.xml
- https://webservice.contoso.com/autodiscover/autodiscover.xml

The Autodiscover service is only available on https://webservice.contoso.com/autodiscover/autodiscover.xml, but https://contoso.com/autodiscover/autodiscover.xml is configured to respond with an **HTTP 302 Redirect** response with the **Location** header set to "https://webservice.contoso.com/autodiscover/autodiscover.xml".
Figure 2: Client and server autodiscovery

Step 1

The Autodiscover client is configured to use the email address user@contoso.com.

The client sends the Autodiscover request XML shown in section 4.1 via HTTP POST to the following URL: https://contoso.com/autodiscover/autodiscover.xml.

Step 2

The client is returned an HTTP 302 redirection to the following URL: https://webservice.contoso.com/autodiscover/autodiscover.xml.

Step 3

The client then reposts the request to this URL.

Step 4

The user’s mailbox is on mail.contoso.com. The server returns the response XML shown in section 4.3.

4.1 Autodiscover Request

The following example shows an Autodiscover request.

```xml
<Autodiscover
 <Request>
  <EMailAddress>user@contoso.com</EMailAddress>
  <AcceptableResponseSchema>
  </AcceptableResponseSchema>
 </Request>
</Autodiscover>
```

4.2 Autodiscover Redirect

The following example shows an Autodiscover redirect to a new email address.
4.3 Autodiscover Configuration

The following example shows an Autodiscover response that contains configuration information.

```xml
<?xml version="1.0" encoding="utf-8"?>
    <User>
      <DisplayName>User Display Name</DisplayName>
      <LegacyDN>/o=microsoft/ou=Contoso/cn=Recipients/cn=486021</LegacyDN>
      <AutoDiscoverSMTPAddress>user@Contoso.com</AutoDiscoverSMTPAddress>
      <DeploymentId>30c3a927-42aa-5de8-91e3-8e5b4655ed00</DeploymentId>
    </User>
    <Account>
      <AccountType>email</AccountType>
      <Action>settings</Action>
      <Protocol>
        <Type>EXCH</Type>
        <Server>ExchangeServer.Contoso.com</Server>
        <ServerDN>/o=Contoso/ou=Exchange Administrative Group (GZZHBOHF23SPELT)/cn=Configuration/cn=cn=ExchangeServer</ServerDN>
        <ServerVersion>720180F0</ServerVersion>
        <MdbDN>/o=Contoso/ou=Exchange Administrative Group (GZZHBOHF23SPELT)/cn=cn=cn=Microsoft Private MDB</MdbDN>
        <PublicFolderServer>PublicFolderServer.Contoso.com</PublicFolderServer>
        <AD>ADServer.Contoso.com</AD>
        <ASUrl>https://mail.Contoso.com/ews/exchange.asmx</ASUrl>
        <EwsUrl>https://mail.Contoso.com/ews/exchange.asmx</EwsUrl>
        <OOFUrl>https://mail.Contoso.com/ews/exchange.asmx</OOFUrl>
        <UMUrl>https://mail.Contoso.com/unifiedmessaging/service.asmx</UMUrl>
        <OABUrl>https://mail.Contoso.com/oab/68b5509d-87f6-4e78-a9ff-74d7d9572787</OABUrl>
      </Protocol>
      <Protocol>
        <Type>EXPR</Type>
        <Server>RPHHTTPServer.Contoso.com</Server>
        <SSL>On</SSL>
        <AuthPackage>Ntlm</AuthPackage>
        <ASUrl>https://mail.Contoso.com/ews/exchange.asmx</ASUrl>
        <EcpUrl>https://mail.Contoso.com/ecp/EcpUrl</EcpUrl>
        <EcpUrl-um>?p=customize/voicemail.aspx&amp;exsvurl=1</EcpUrl-um>
        <EcpUrl-aggr>?p=personalsettings/EmailSubscriptions.slab&amp;exsvurl=1</EcpUrl-aggr>
        <EcpUrl-sms>?p=textmessaging.slab&amp;exsvurl=1</EcpUrl-sms>
        <EcpUrl-publish>customize/calendarpublishing.slab?exsvurl=1&amp;FldID=&lt;FldID&gt;&amp;MsgID=&lt;MsgID&gt;&amp;Msgx=&lt;Msgx&gt;&amp;Sender=&lt;Sender&gt;</EcpUrl-publish>
        <EcpUrl-nt>PersonalSettings/DeliveryReport.aspx?exsvurl=1&amp;IsOWA=1&amp;IsOWA&amp;MSGID=1&amp;Msgx=1&amp;Sender=&lt;Sender&gt;</EcpUrl-nt>
        <EcpUrl-ret>?organize/retentionpolicytags.slab&amp;exsvurl=1</EcpUrl-ret>
        <EwsUrl>https://mail.Contoso.com/ews/exchange.asmx</EwsUrl>
        <OOFUrl>https://mail.Contoso.com/ews/exchange.asmx</OOFUrl>
        <UMUrl>https://mail.Contoso.com/unifiedmessaging/service.asmx</UMUrl>
        <OABUrl>https://mail.Contoso.com/oab/58b5509d-87f6-4e78-a9ff-74d7d9572787</OABUrl>
      </Protocol>
    </Account>
  </Response>
</Autodiscover>
```
4.4 MapiHttp Response

The following example shows a MapiHttp response (section 2.2.4.1.1.2.6) that contains configuration information.

```xml
<?xml version="1.0" encoding="utf-8"?>
    <User>
      <DisplayName>User Display Name</DisplayName>
      <LegacyDN>/o=microsoft/ou=Contoso/cn=Recipients/cn=486021</LegacyDN>
      <AutoDiscoverSMTPAddress>user@Contoso.com</AutoDiscoverSMTPAddress>
      <DeploymentId>30c3a927-42aa-5de8-91e3-8e5b4655ed00</DeploymentId>
    </User>
    <Account>
      <AccountType>email</AccountType>
      <Action>settings</Action>
      <Protocol Type="mapiHttp" Version="1">
        <MailStore>
          <InternalUrl>https://mail.Contoso.com/mapi/emsmdb/?MailboxId=416c6368-656d-794a-6f45-57615272456e@Contoso.com</InternalUrl>
        </MailStore>
        <AddressBook>
          <InternalUrl>https://mail.Contoso.com/mapi/nspi/?MailboxId=416c6368-656d-794a-6f45-57615272456e@Contoso.com</InternalUrl>
        </AddressBook>
      </Protocol>
    </Account>
  </Response>
</Autodiscover>
```
4.5 Autodiscover Server Errors

The following example shows an Autodiscover response that contains an error.

```xml
  <Response>
    <Error Time="17:40:40.6157343" Id="3191339394">
      <ErrorCode>500</ErrorCode>
      <Message>The email address cannot be found.</Message>
      <DebugData />
    </Error>
  </Response>
</Autodiscover>
```
5 Security

5.1 Security Considerations for Implementers

There are no special security considerations specific to this specification. It is recommended that clients perform an Autodiscover request by using this protocol over HTTPS (HTTP with SSL).

It is also recommended that a server not answer Autodiscover queries unless the Autodiscover client has been authenticated with the Autodiscover server.

5.2 Index of Security Parameters

None.
6 Appendix A: XSDs

For ease of implementation, the following sections provide the four XML schema definitions (XSDs) for this protocol.

<table>
<thead>
<tr>
<th>XSD name</th>
<th>Prefix</th>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autodiscover request XSD</td>
<td>xs:</td>
<td>6.1</td>
</tr>
<tr>
<td>Autodiscover response XSD</td>
<td>xs:</td>
<td>6.2</td>
</tr>
<tr>
<td>Autodiscover error response XSD</td>
<td>xs:</td>
<td>6.3</td>
</tr>
<tr>
<td>Autodiscover redirect response XSD</td>
<td>xs:</td>
<td>6.4</td>
</tr>
</tbody>
</table>

6.1 Autodiscover Request XSD

The following is the Autodiscover request XSD.

```xml
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Autodiscover">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="EMailAddress" type="xs:string" minOccurs="0"/>
              <xs:element name="LegacyDN" type="xs:string" minOccurs="0"/>
              <xs:element name="AcceptableResponseSchema" type="xs:string" minOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

6.2 Autodiscover Response XSD

The following is the Autodiscover response XSD.

```xml
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Autodiscover">
    <xs:complexType>
      <xs:sequence>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmllns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Response">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="User">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="DisplayName" type="xs:string" />
              <xs:element name="LegacyDN" type="xs:string" />
              <xs:element name="AutoDiscoverSMTPAddress" type="xs:string" />
              <xs:element name="DeploymentId" type="xs:string" />
              <xs:element minOccurs="0" name="DefaultABView" type="xs:string" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="Account">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="AccountType" type="xs:string" />
              <xs:element name="Action" type="xs:string" />
              <xs:element name="MicrosoftOnline">
                <xs:simpleType>
                  <xs:restriction base="xs:string">
                    <xs:enumeration value="True"/>
                    <xs:enumeration value="False"/>
                  </xs:restriction>
                </xs:simpleType>
              </xs:element>
              <xs:element name="ConsumerMailbox" minOccurs="0">
                <xs:simpleType>
                  <xs:restriction base="xs:string">
                    <xs:enumeration value="True"/>
                    <xs:enumeration value="False"/>
                  </xs:restriction>
                </xs:simpleType>
              </xs:element>
              <xs:element maxOccurs="unbounded" name="Protocol">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element minOccurs="0" name="Type" type="xs:string"/>
                    <xs:element minOccurs="0" name="MailStore">
                      <xs:complexType>
                        <xs:all minOccurs="1">
                          <xs:element minOccurs="0" name="InternalUrl" type="xs:string" />
                          <xs:element minOccurs="0" name="ExternalUrl" type="xs:string" />
                        </xs:all>
                      </xs:complexType>
                    </xs:element>
                    <xs:element minOccurs="0" name="AddressBook" type="xs:string" />
                    <xs:element minOccurs="0" name="MailStore">
                      <xs:complexType>
                        <xs:all minOccurs="1">
                          <xs:element minOccurs="0" name="InternalUrl" type="xs:string" />
                          <xs:element minOccurs="0" name="ExternalUrl" type="xs:string" />
                        </xs:all>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
<xs:complexType>
  <xs:element minOccurs="0" name="Internal">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="OWAUrl" maxOccurs="unbounded">
          <xs:complexType>
            <xs:simpleContent>
              <xs:extension base="xs:string">
                <xs:attribute name="AuthenticationMethod" type="xs:string" use="required" />
              </xs:extension>
            </xs:simpleContent>
          </xs:complexType>
        </xs:element>
        <xs:element name="Protocol">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Type" type="xs:string" />
              <xs:element name="ASUrl" type="xs:string" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:complexType>

<xs:element minOccurs="0" name="External">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="OWAUrl" maxOccurs="unbounded">
        <xs:complexType>
          <xs:simpleContent>
            <xs:extension base="xs:string">
              <xs:attribute name="AuthenticationMethod" type="xs:string" use="required" />
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
      </xs:element>
      <xs:element name="Protocol">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="Type" type="xs:string" />
            <xs:element name="ASUrl" type="xs:string" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element minOccurs="0" name="Server" type="xs:string" />
<xs:element minOccurs="0" name="SSL" type="xs:string" />
<xs:element minOccurs="0" name="AuthPackage" type="xs:string" />
<xs:element minOccurs="0" name="ServerDN" type="xs:string" />
<xs:element minOccurs="0" name="ServerVersion" type="xs:string" />
<xs:element minOccurs="0" name="MdbDN" type="xs:string" />
<xs:element minOccurs="0" name="PublicFolderServer" type="xs:string" />
<xs:element minOccurs="0" name="AD" type="xs:string" />
<xs:element minOccurs="0" name="TTL" type="xs:string" />
<xs:element minOccurs="0" name="ASUrl" type="xs:string" />
<xs:element minOccurs="0" name="EwsUrl" type="xs:string" />
<xs:element minOccurs="0" name="EmwsUrl" type="xs:string" />
<xs:element minOccurs="0" name="SharingUrl" type="xs:string" />
<xs:element minOccurs="0" name="EcpUrl" type="xs:string" />
<xs:element minOccurs="0" name="EcpUrl-aggr" type="xs:string" />
<xs:element minOccurs="0" name="EcpUrl-mt" type="xs:string" />
6.3 Autodiscover Error Response XSD

The following is the Autodiscover error response XSD.

```xml
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" targetNamespace="http://schemas.microsoft.com/exchange/autodiscover/responseschema/2006"/>
```
6.4 Autodiscover Redirect Response XSD

The following is the Autodiscover redirect response XSD.

```xml
  <xs:element name="Autodiscover">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Response" ref="q1:Response" />
      </xs:sequence>
    </xs:complexType>
    <xs:element name="Error">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="ErrorCode" type="xs:unsignedShort" minOccurs="1" />  
          <xs:element name="Message" type="xs:string" minOccurs="1" />
          <xs:attribute name="Time" type="xs:time" use="required" />
          <xs:attribute name="Id" type="xs:unsignedInt" use="required" />
        </xs:sequence>
        <xs:attribute name="Time" type="xs:time" use="required" />
      </xs:complexType>
    </xs:element>
  </xs:complexType>
</xs:schema>
```
7 Appendix B: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include updates to those products.

- Microsoft Exchange Server 2007
- Microsoft Exchange Server 2010
- Microsoft Exchange Server 2013
- Microsoft Exchange Server 2016
- Microsoft Exchange Server 2019
- Microsoft Office Outlook 2007
- Microsoft Outlook 2010
- Microsoft Outlook 2013
- Microsoft Outlook 2016
- Microsoft Outlook 2019
- Microsoft Outlook 2021

Exceptions, if any, are noted in this section. If an update version, service pack or Knowledge Base (KB) number appears with a product name, the behavior changed in that update. The new behavior also applies to subsequent updates unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

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

<1> Section 2.2.2.3: The X-ClientCanHandle header is used to indicate the capabilities that the client supports when the client is not Outlook. Office Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016, and Outlook 2019 do not use this header. Exchange 2007, Exchange 2010, and the initial release of Exchange 2013 do not support processing of the X-ClientCanHandle header. Exchange 2013 cumulative update 6, Exchange 2016, and Exchange 2019 support processing of the X-ClientCanHandle header.

<2> Section 2.2.4.1.1.1.1: Exchange 2007 and the initial release version of Exchange 2010 do not include the AutoDiscoverSMTPAddress element. The AutoDiscoverSMTPAddress element was introduced in Microsoft Exchange Server 2010 Service Pack 1 (SP1).

<3> Section 2.2.4.1.1.1.2: Exchange 2007, and the initial release version of Exchange 2010 do not include the DefaultABView element. The DefaultABView element was introduced in Exchange 2010 SP1.

<4> Section 2.2.4.1.1.2.3: The MicrosoftOnline element is not supported by Exchange 2007, Exchange 2010 and Exchange 2013.

<5> Section 2.2.4.1.1.2.4: The ConsumerMailbox element is not supported by Exchange 2007, Exchange 2010 and Exchange 2013.

<6> Section 2.2.4.1.1.2.5: The AlternativeMailbox element is not supported by Exchange 2007.

<7> Section 2.2.4.1.1.2.5.1: The DisplayName element is not supported by Exchange 2007.
Section 2.2.4.1.1.2.5.2: The LegacyDN element is not supported by Exchange 2007.

Section 2.2.4.1.1.2.5.3: The Server element is not supported by Exchange 2007.

Section 2.2.4.1.1.2.5.4: Exchange 2007 and Exchange 2010 do not support the SmtpAddress element. Office Outlook 2007 and Outlook 2010 ignore the SmtpAddress element.

Section 2.2.4.1.1.2.5.5: The Type element is not supported by Exchange 2007.

Section 2.2.4.1.1.2.5.5: Exchange 2007 and Exchange 2010 do not support the "TeamMailbox" value for the Type element. Office Outlook 2007 and Outlook 2010 ignore AlternativeMailbox elements with a value of "TeamMailbox" in the child Type element.

Section 2.2.4.1.1.2.6.4: The Negotiate authentication method is not implemented in Exchange 2007, Exchange 2010, Office Outlook 2007, or Outlook 2010.

Section 2.2.4.1.1.2.6.4: The "anonymous" value for the AuthPackage element is not implemented in Exchange 2007, Exchange 2010, Office Outlook 2007, or Outlook 2010.

Section 2.2.4.1.1.2.6.9: Exchange 2007 does not support the EcpUrl element.

Section 2.2.4.1.1.2.6.9: Exchange 2007 does not support the EcpUrl-aggr element.

Section 2.2.4.1.1.2.6.10: Exchange 2007 and Exchange 2010 do not support the EcpUrl-extinstall element. Office Outlook 2007 and Outlook 2010 ignore this element.

Section 2.2.4.1.1.2.6.11: Exchange 2007 and Exchange 2010 do not support the EcpUrl-publish element.

Section 2.2.4.1.1.2.6.12: Exchange 2007 does not support the EcpUrl-ret element.

Section 2.2.4.1.1.2.6.13: Exchange 2007 and Exchange 2010 do not support the EcpUrl-photo element. Office Outlook 2007 and Outlook 2010 ignore this element.

Section 2.2.4.1.1.2.6.14: Exchange 2007 and Exchange 2010 do not support the EcpUrl-publish element.

Section 2.2.4.1.1.2.6.15: Exchange 2007 does not support the EcpUrl-sms element.

Section 2.2.4.1.1.2.6.16: Exchange 2007 and Exchange 2010 do not support the EcpUrl-tm element. Office Outlook 2007 and Outlook 2010 ignore this element.

Section 2.2.4.1.1.2.6.17: Exchange 2007 and Exchange 2010 do not support the EcpUrl-tmCreating element. Office Outlook 2007 and Outlook 2010 ignore this element.

Section 2.2.4.1.1.2.6.18: Exchange 2007 and Exchange 2010 do not support the EcpUrl-tmEditing element. Office Outlook 2007 and Outlook 2010 ignore this element.

Section 2.2.4.1.1.2.6.19: Exchange 2007 and Exchange 2010 do not support the EcpUrl-tmHiding element. Office Outlook 2007 and Outlook 2010 ignore this element.

Section 2.2.4.1.1.2.6.20: Exchange 2007 and Exchange 2010 do not support the EcpUrl-tmHiding element. Office Outlook 2007 and Outlook 2010 ignore this element.

Section 2.2.4.1.1.2.6.21: Exchange 2007 does not support the EcpUrl-um element.

Section 2.2.4.1.1.2.6.26: Exchange 2007, Exchange 2010, and the initial release of Exchange 2013 don't support the GroupingInformation element. The GroupingInformation element was introduced in Microsoft Exchange Server 2013 Service Pack 1 (SP1).


Section 2.2.4.1.1.2.6.38: Exchange 2007, Exchange 2010, Office Outlook 2007, and Outlook 2010 do not support the ServerExclusiveConnect element.
Section 2.2.4.1.1.2.6.41: Exchange 2007 and Exchange 2010 do not support the EcpUrl-extinstall element. Office Outlook 2007 and Outlook 2010 ignore this element.

Section 2.2.4.1.1.2.6.46: Exchange 2007 and Exchange 2010 do not support the "EXHTTP" value for the Type element. Office Outlook 2007 and Outlook 2010 ignore Protocol elements with a Type child element that has a value of "EXHTTP".

Section 2.2.4.1.1.2.6.46: Exchange 2007, Exchange 2010, Exchange 2013, Exchange 2016, and Exchange 2019 do not support the "POP3" value for the Type element.

Section 2.2.4.1.1.2.6.46: Exchange 2007, Exchange 2010, Exchange 2013, Exchange 2016, and Exchange 2019 do not support the "SMTP" value for the Type element.

Section 2.2.4.1.1.2.6.46: Exchange 2007, Exchange 2010, Exchange 2013, Exchange 2016, and Exchange 2019 do not support the "IMAP" value for the Type element.

Section 2.2.4.1.1.2.6.46: Exchange 2007, Exchange 2010, Exchange 2013, Exchange 2016, and Exchange 2019 do not support the "DAV" value for the Type element.

Section 2.2.4.1.1.2.7: Exchange 2007 and Exchange 2010 do not support the PublicFolderInformation element. Office Outlook 2007 and Outlook 2010 ignore the PublicFolderInformation element.

Section 3.1.5.4: Office Outlook 2007 and Outlook 2010 ignore the ServerExclusiveConnect element.

Section 3.1.5.4: Office Outlook 2007 and Outlook 2010 ignore Protocol elements that contain a Type element with the "EXHTTP" value, and do not ignore Protocol elements that contain a Type element with the "EXPR" value in this case.

Section 3.1.5.4: Office Outlook 2007 and Outlook 2010 ignore Protocol elements that contain a Type element with the "EXHTTP" value.

Section 3.2.5: Exchange 2007, Exchange 2010, and the initial release of Exchange 2013 do not support processing of the X-MapiHttpCapability header. Exchange 2013 SP1, Exchange 2016, and Exchange 2019 support processing of the X-MapiHttpCapability header only when it is specifically enabled.
8 Change Tracking

This section identifies changes that were made to this document since the last release. Changes are classified as Major, Minor, or None.

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.
- A document revision that captures changes to protocol functionality.

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 None means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the relevant technical content is identical to the last released version.

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

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Revision class</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 Appendix B: Product Behavior</td>
<td>Updated list of supported products.</td>
<td>major</td>
</tr>
</tbody>
</table>
Standards assignments 12

T

Timer events
  client 33
  server 35
Timers
  client 31
  server 33

Tracking changes 51

Transport 13

Triggered events - higher-layer
  client 31
  server 33

V

Vendor-extensible fields 12

Versioning 12

X

XML schema definitions 42

XSDs
  Autodiscover error response 45
  Autodiscover redirect response 46
  Autodiscover request 42
  Autodiscover response 42
  overview 42