[MS-OXLDAP]: Lightweight Directory Access Protocol (LDAP) Version 3 Extensions Specification

Revision Summary			
Author	Date	Version	Comments
Microsoft Corporation	April 4, 2008	0.1	Initial Availability.
Microsoft Corporation	April 25, 2008	0.2	Revised and updated property names and other technical content.
Microsoft Corporation	June 27, 2008	1.0	Initial Release.
Microsoft Corporation	August 6, 2008	1.01	Revised and edited technical content.

Table of Contents

1 Introduction ***4***
1.1 Glossary 4
1.2 References 4
1.2.1 Normative References 4
1.2.2 Informative References 5
1.3 Protocol Overview 6
1.4 Relationship to Other Protocols 6
1.5 Prerequisites/Preconditions 6
1.6 Applicability Statement 6
1.7 Versioning and Capability Negotiation 6
1.8 Vendor-Extensible Fields 6
1.9 Standards Assignments 6
2 Messages ***6***
2.1 Transport 6
2.2 Message Syntax 6
2.2.1 Protocol-Specific Name Attributes 9
2.2.1.1 Display Name 9
2.2.2 Protocol-Specific Organizational Attributes 9
2.2.2.1 Reports 9
2.2.3 Protocol-Specific E-Mail Attributes 9
2.2.3.1 Exchange Distinguished Name 9
2.2.3.2 Proxy Addresses 10
2.2.3.3 Exchange Home Server 10
2.2.4 Other Protocol-Specific Attributes 10
2.2.4.1 Object Class 10
2.2.4.2 S/MIME Certificate 11
3 Protocol Details ***11***
3.1 Client Details 11
3.1.1 Abstract Data Model 11
3.1.2 Timers 11
3.1.3 Initialization 11
3.1.3.1 Querying for Supported Controls 11
3.1.3.2 Querying for Supported Capabilities 12
3.1.4 Higher-Layer Triggered Events 12
3.1.4.1 Issuing a Search Request 12
3.1.4.1.1 Retrieving a Search Base 13
3.1.4.1.2 Basic Search Filter 14
3.1.4.1.3 Advanced Search Filter 14
3.1.4.1.4 Ambiguous Name Resolution (ANR) Search Filter 15
3.1.5 Message Processing Events and Sequencing Rules 15
3.1.6 Timer Events 15
3.1.7 Other Local Events 16

3.2 Server Details 16
3.2.1 Abstract Data Model 16
3.2.2 Timers 16
3.2.3 Initialization 16
3.2.4 Higher-layer Triggered Events 16
3.2.5 Message Processing Events and Sequencing Rules 16
3.2.5.1 Handling a Query for the supportedControl Attribute 16
3.2.5.2 Handling a Query for the supportedCapabilities Attribute 16
3.2.5.3 Handling Search Requests 17
3.2.5.3.1 Handling a Query for the defaultNamingContext Attribute 17
3.2.5.3.2 Responding to Query Attributes 17
3.2.6 Timer Events 17
3.2.7 Other Local Events 17
4 Protocol Examples 17
4.1 Simple Search Scenario 17
5 Security 19
5.1 Security Considerations for Implementers 19
5.2 Index of Security Parameters 19
6 Appendix A: Office/Exchange Behavior 19
Index 22

1 Introduction

This document specifies Office extensions to the **Lightweight Directory Access Protocol (LDAP)**, as specified in [RFC4511] and [RFC4512], as well as extensions to the LDAP user schema [RFC4519]. LDAP is an Internet protocol used to query and modify directory entries, and is commonly leveraged to query and create a user directory containing information about a large number of users or groups of users.

1.1 Glossary

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

distinguished name (DN)
LDAP server
Lightweight Directory Access Protocol (LDAP)

The following terms are specific to this document:

AD-type server: An **LDAP server** that returns an OID value of "1.2.840.113556.1.4.800" when queried for the supportedCapabilities **LDAP attribute**. See section 3.1.3.2.

LDAP attribute: The attribute specified in [RFC4512] section 2.2.

LDAP Distinguished Name: A string representing an object on a directory server, as specified in [RFC4514].

multi-valued LDAP attribute: An **LDAP attribute** that can have one or more values, as specified in [RFC4512].

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

1.2 References

1.2.1 Normative References

[MS-OXGLOS] Microsoft Corporation, "Office Exchange Protocols Master Glossary", June 2008.

[RFC1274] Barker, P. and Kille, S., "The COSINE and Internet X.500 Schema", RFC 1274, November 1991, http://www.ietf.org/rfc/rfc1274.txt.

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

[RFC2696] Weider, C., Herron, A., Anantha, A., and Howes, T., "LDAP Control Extension for Simple Paged Results Manipulation", RFC 2696, September 1999, http://www.ietf.org/rfc/rfc2696.txt.

[RFC2798] Smith, M., "Definition of the inetOrgPerson LDAP Object Class", RFC 2798, April 2000, http://www.ietf.org/rfc/rfc2798.txt.

[RFC2891] Howes, T., Wahl, M., and Anantha, A., "LDAP Control Extension for Server Side Sorting of Search Results", RFC 2891, August 2000, http://www.ietf.org/rfc/rfc2891.txt.

[RFC4234] Crocker, D., Ed. and Overell, P., "Augmented BNF for Syntax Specifications: ABNF", RFC 4234, October 2005, http://www.ietf.org/rfc/rfc4234.txt.

[RFC4511] Sermersheim, J., "Lightweight Directory Access Protocol (LDAP): The Protocol", RFC 4511, June 2006, http://www.ietf.org/rfc/rfc4511.txt.

[RFC4512] Zeilenga, K., "Lightweight Directory Access Protocol (LDAP): Directory Information Models", RFC 4512, June 2006, http://www.ietf.org/rfc/rfc4512.txt.

[RFC4514] Zeilenga, K., "Lightweight Directory Access Protocol (LDAP): String Representation of Distinguished Names", RFC 4514, June 2006, http://www.ietf.org/rfc/rfc4514.txt.

[RFC4519] Sciberras, A., "Lightweight Directory Access Protocol (LDAP): Schema for User Applications", RFC 4519, June 2006, http://www.ietf.org/rfc/rfc4519.txt.

[RFC4523] Zeilanga, K., "Lightweight Directory Access Protocol (LDAP) Schema Definitions for X.509 Certificates", RFC 4523, June 2006, http://www.ietf.org/rfc/rfc4523.txt.

[RFC4524] Zeilenga, K., "COSINE LDAP/X.500 Schema", RFC 4524, June 2006, http://www.ietf.org/rfc/rfc4524.txt.

1.2.2 Informative References

[LDAPEX-SVB] Boreham, D., Sermersheim, J., Kashi, A., "LDAP Extensions for Scrolling View Browsing of Search Results", November 2002, http://www.ietf.org/proceedings/02nov/I-D/draft-ietf-ldapext-ldapv3-vlv-09.txt.

1.3 Protocol Overview

LDAP is an Internet protocol specified in [RFC4511] that is used for querying and modifying entries in a directory server.

This document specifies an extension to the Lightweight Directory Access Protocol as specified in [RFC4511], [RFC4512], and [RFC4519]. It specifies which portions of these RFCs are implemented by this protocol extension, and it defines specific attributes used in addition to those specified in these RFCs.

1.4 Relationship to Other Protocols

This protocol extends [RFC4511], [RFC4512], and [RFC4519].

1.5 Prerequisites/Preconditions

None.

1.6 Applicability Statement

This protocol extension can be used to retrieve specific information from an **LDAP server**.

1.7 Versioning and Capability Negotiation

This protocol extension does not introduce any versioning constraints beyond those specified in [RFC4511].

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

This protocol extends the **LDAP** protocol as specified in [RFC4511].

2.2 Message Syntax

Message syntax follows the **LDAP** standard, as specified in [RFC4511]. According to the LDAP standard, an attribute list can contain implementation-specific attributes. The attributes specific to this protocol extension are defined in this section.

The following table lists every **LDAP attribute** for which the client SHOULD query. In many cases, more than one LDAP attribute corresponds to a single field in the table below, because different server implementations of the LDAP protocol use different attribute names to represent similar concepts (fields). In those cases, attributes listed first

in the table take precedence over attributes listed later. For example, for the Last Name field, the sn attribute takes precedence over the surname attribute.

The client SHOULD implement [RFC4519], [RFC4524], [RFC2798], and [RFC4523], and it SHOULD support the attributes that are listed in the following table. Attributes specific to this protocol are marked by comments in the "Additional Notes" column, and are further described in this section.

Field	LDAP attribute	Additional notes
Name attributes		
Display Name	display-name displayname cn commonName	The display-name and displayname attributes are specific to this protocol (section 2.2.1.1). The cn and commonName attributes are specified in [RFC4519].
Last Name	sn surname	Specified in [RFC4519].
First Name	givenName	Specified in [RFC4519].
Initials	initials	Specified in [RFC4519].
Organizational attributes		
Company Name	organizationName o <1>	Specified in [RFC4519].
Title	title	Specified in [RFC4519].
Organizational Unit	ou organizationalUnitName department	The department attribute is specific to this protocol. The ou and organizationalUnitName attributes are specified in [RFC4519].
Office Location	physicalDeliveryOfficeName	Specified in [RFC4519].
Assistant Name	secretary	Specified in [RFC4524].
Manager	manager	Specified in [RFC4524].
Reports	directReports reports	Multi-valued attributes, specific to this protocol (section 2.2.2.1).
E-Mail attributes		
Email Address	mail	Specified in [RFC4524].

Exchange Distinguished Name	legacyExchangeDN	This attribute is specific to this protocol (section 2.2.3.1).
Account	mailNickname uid	The mailNickname attribute is specific to this protocol, and is used in the same way as the uid attribute. The uid attribute is specified in [RFC4519].
X.400 Address	TextEncodedORaddress	This attribute is specific to this protocol. Text representation of an X.400 O/R address. For more details, see [RFC1274].
Exchange Home Server	msExchHomeServerName	This attribute is specific to this protocol (section 2.2.3.3).
Proxy Addresses	proxyAddresses otherMailbox	Multi-valued attributes specific to this protocol (section 2.2.3.2).
Physical Address attributes		
Address	postalAddress streetAddress	Specified in [RFC4519].
Locality / City	l	Specified in [RFC4519].
State	st	Specified in [RFC4519].
Postal Code	postalCode	Specified in [RFC4519].
Country	co	Specified in [RFC4519].
Telephone attributes		
Telephone Number	telephoneNumber	Specified in [RFC4519].
Secondary Phone Number	Telephone-Office2	This attribute is specific to this protocol, and is used to query for a secondary telephone number associated with the directory entry.
Fax Number	facsimileTelephoneNumber	Specified in [RFC4519].

Assistant Phone Number	Telephone-Assistant	This attribute is specific to this protocol, and is used to query for the assistant's telephone number associated with the directory entry.
Home Phone	homephone	Specified in [RFC4524].
Cell Phone	mobile	Specified in [RFC4524].
Pager Number	pager	Specified in [RFC4524].
Notes	info	Specified in [RFC4524].
Other attributes		
User Certificate	userCertificate;binary	Specified in [RFC4523].
S/MIME Certificate	userSMIMECertificate;binary	For more details, see section 2.2.4.2.
Unused	user-cert;binary <2>	
Object Class	objectClass	For more details about supported values, see section 2.2.4.1.
Role Occupant	roleOccupant	Specified in [RFC4519].

2.2.1 Protocol-Specific Name Attributes

2.2.1.1 Display Name

The Display Name attribute SHOULD be used as the primary name to be shown to the user when displaying an **LDAP** entry. If the Display Name entry is empty or not user-readable, the client SHOULD construct a display name from other attributes. Applications use implementation-specific logic to construct a display name when needed.<3>

2.2.2 Protocol-Specific Organizational Attributes

2.2.2.1 Reports

The Reports attribute is a multi-valued string attribute containing the **LDAP Distinguished Names** of any direct reports.

2.2.3 Protocol-Specific E-Mail Attributes

2.2.3.1 Exchange Distinguished Name

Exchange Distinguished Name (*legacyExchangeDN*) is an attribute that represents a **distinguished name** of the entry. This distinguished name MUST be formatted as specified in [MS-OXOABK]. This value MAY be used as a proxy address for an entry.<4>

2.2.3.2 Proxy Addresses

If multiple e-mail addresses are associated with an entry, they MUST be included in the Proxy Addresses attribute. These addresses can be used as alternate e-mail addresses to reach the user. Specific e-mail addresses can be retrieved from this value depending on the intended use. The semantics of proxy addresses are not constrained by this protocol, and are specific to the protocol that creates the proxy addresses. This protocol does not constrain how a client uses proxy addresses. For the client, these proxy addresses have the same semantics as the values of the **PidTagAddressBookProxyAddresses** property specified in [MS-OXOABK].

```
The format of each e-mail address MUST be:

emailString ::= <emailType> ": " <emailAddress>

emailType ::= ; A string indicating what type of e-mail it is. i.e.
SMTP, x500, etc

emailAddress := ;A string representing the e-mail address
```

Examples:

```
SMTP:user1@example.com
x500:/o=example/cn=user1
```

2.2.3.3 Exchange Home Server

The Exchange Home Server attribute MUST contain the **distinguished name** of the mailbox server where mail is delivered for that user. For the client, this attribute has the same semantics as the **PidTagAddressBookHomeMessageDatabase** property specified in [MS-OXOABK].

2.2.4 Other Protocol-Specific Attributes

2.2.4.1 Object Class

The client SHOULD support the following values for the objectClass attribute.

Value	User type
organizationalPerson	Specified in [RFC4519].
groupOfNames group	groupOfNames is specified in [RFC4519]. group is a value specific to this protocol and is used in the same way as groupOfNames.
Remote-Address	A value specific to this protocol; represents a recipient that is known to be from a foreign or remote messaging system.
Public-Folder	A value specific to this protocol; represents a place where public discussions take place such as a bulletin board, public folder, or shared folder.

The protocol client SHOULD use this value to help distinguish between different types of directory entries when displaying entries to the user. For example, the protocol client could display a different icon or bold the item to make it easy for a user viewing the object to distinguish its type. If no objectClass is returned for an entry, then the client MUST treat it as an organizationalPerson.

This value is used to determine **PidTagDisplayType** as specified in [MS-OXOABK]. The following objectClass values correspond to the following PidTagDisplayType values.

objectClass value	PidTagDisplayType value
organizationalPerson	DT_MAILUSER
groupOfNames group	DT_DISTLIST
Remote-Address	DT_REMOTE_MAILUSER
Public-Folder	DT_FORUM

2.2.4.2 S/MIME Certificate

This binary attribute contains certificates in the format specified in [RFC2798] or certificates in the format defined for the **PidTagUserX509Certificate** property, as specified in [MS-OXOABK]. If available, this attribute SHOULD be preferred over the userCertificate attribute for S/MIME applications.

3 Protocol Details

3.1 Client Details

3.1.1 Abstract Data Model

This extension does not introduce any states or conceptual objects beyond the ones specified in [RFC4511].

3.1.2 Timers

None.

3.1.3 Initialization

Besides the initialization specified in [RFC4511], this protocol extension specifies two operations that SHOULD be performed upon connecting to an **LDAP server**.

3.1.3.1 Querying for Supported Controls

Upon connecting to the **LDAP server**, the client SHOULD query the server for the supportedControl attribute as specified in [RFC4512]. The object identifier (OID) values returned by the server indicate what capabilities the server supports and makes available

to the client. The client SHOULD<5> recognize the following three OID values that a server can return.

Object Identifier (OID) value	Server support
2.16.840.1.113730.3.4.9	Virtual List Support, as specified in [LDAPEX-SVB]. <6>
1.2.840.113556.1.4.319	Paged Results Support, as specified in [RFC2696].
1.2.840.113556.1.4.473	Server Sort Support, as specified in [RFC2891].

Any other OID value returned by the server MAY<7> be ignored by the client.

3.1.3.2 Querying for Supported Capabilities

Upon connecting to the **LDAP server**, the client SHOULD query the server for the supportedCapabilities attribute as specified in [RFC4511], and MUST recognize the OID value for an AD-type server: 1.2.840.113556.1.4.800.

Any other OID values returned by the server MAY<8> be ignored by the client. If the client does not query for this capability, or the server does not return the value in the table above, the client MUST treat the server as a non-AD-type LDAP server.

When sorting, the protocol client SHOULD use the displayName attribute instead of the cn attribute on **AD-type servers**.

3.1.4 Higher-Layer Triggered Events

3.1.4.1 Issuing a Search Request

All search requests issued by the client MUST follow the **Search Request** definition specified in section 4.5.1 of [RFC4511], with the following options specified.

Search Request parameters	Value
baseObject	See section 3.1.4.1.1.
Scope	wholeSubtree
derefAliases	derefAlways
typesOnly	FALSE
sizeLimit	Specified by user.
timeLimit	Specified by user.

Search Request parameters	Value
AttributeSelection	cn, commonName, mail, roleOccupant, display-name, displayname, sn, surname, co, organizationName, o, givenName, legacyExchangeDN, objectClass, uid, mailNickname, title, company, physicalDeliveryOfficeName, telephoneNumber
Filter	Depends on the type of search (sections 3.1.4.1.2, 3.1.4.1.3, and 3.1.4.1.4).

3.1.4.1.1 Retrieving a Search Base

A **Search Base** is a string representing the **LDAP Distinguished Name** of the base object entry relative to which a search is to be performed. This value SHOULD be used as the *baseObject* of a **Search Request** as specified in [RFC4511].

The client can use a user-provided string as the search base. If a search base is not specified, the client SHOULD <9> send a Search Request to the server (as specified in [RFC4511] section 4.5.1) with the following options specified.

Search request parameters	Value
baseObject	Empty string (i.e. a zero length string).
Scope	baseObject
derefAliases	neverDerefAliases
typesOnly	FALSE
sizeLimit	0
TimeLimit	0
Filter	(objectClass=*)
Attributes	objectClass, defaultNamingContext

If the server returns a defaultNamingContext attribute, it MAY <10> be used as the search base for the **LDAP** search. The client SHOULD query for the defaultNamingContext attribute before any search, and SHOULD then utilize the return value as the *baseObject* of any subsequent searches. Since the *baseObject* SHOULD be

specified during a search, the client SHOULD issue an LDAP search request for a defaultNamingContext before any other search requests, if no Search Base has been specified.

3.1.4.1.2 Basic Search Filter

When performing a basic search, the client SHOULD use the following filter as the search filter. <11>

This search filter is specified in Augmented Backus-Naur Form (ABNF)specified in [RFC4234].

```
basicSearchFilter ::= " (&(|(mail= " <search-string> "*)(cn= "
           <search-string> "*)(sn= " <search-string> "*)(givenName=
           " <search-string> "*)(displayName= " <search-string>
           "*))) "

search-string ::= ; a user specified search string
```

3.1.4.1.3 Advanced Search Filter

The client SHOULD <12> provide a way to search on one or more **LDAP** attributes. The client SHOULD use strings provided by the user to construct the appropriate LDAP filter.

This search filter is specified in Augmented Backus-Naur Form (ABNF) specified in[RFC4234].

```
advancedFilter ::= "(&(| " *<indvidualAttribute> ")) "

individualAttribute ::= "( " <attributeName> "= " <attributeValue>
           ") "

attributeName ::= displayName / display-name / cn /
           physicalDeliveryOfficeName / roomNumber / uid /
           mailNickname / givenName / sn / telephoneNumber / l /
           title / department / mail

attributeValue ::= [<containsORbegins>] <userSpecifiedValue> "* "

containsORbegins ::= "*"    ; include if searching for a substring,
           exclude if looking for a string beginning with a
           substring

userSpecifiedValue ::=      ; a user specified value for that field
```

For each **Search Field** requested by the user, the client MUST add all <attributeValue> entries specified in the following table.

Search Field	attributeValue	Notes

Search Field	attributeValue	Notes
Display Name	displayName display-name cn	displayName and display-name are used in AD-type search filters. cn is used in non-AD-type search filters. For more details about which **LDAP servers** are **AD-type servers,** see section 3.1.3.2.
Office Location	physicalDeliveryOfficeName roomNumber	
Account	uid mailNickname	
First Name	givenName	
Last Name	sn	
Telephone Number	telephoneNumber	
Locality / City	l	
Title	title	
Department	department	
Email	mail	

3.1.4.1.4 Ambiguous Name Resolution (ANR) Search Filter

An ambiguous name resolution (ANR) search is a search algorithm implemented by the client that allows a client to find directory objects by matching user-provided strings with common attributes. ANR is useful when locating objects for which the user does not have complete information. For example, a user might know the name "John Smith", but not the e-mail address. When the client performs an ANR search, it SHOULD use the following query.

This search query is specified in ABNF specified in[RFC4234].

```
ANRFilter ::= " (&(mail=*) (|(mail= " <search-string> "*)(cn= "
            <search-string> "*)(sn= " <search-string> "*)(givenName=
            " <search-string> "*)(displayName= " <search-string>
            "*))) "

search-string ::= ; a user specified search string
```

3.1.5 Message Processing Events and Sequencing Rules

None.

3.1.6 Timer Events

None.

3.1.7 Other Local Events

None.

3.2 Server Details

3.2.1 Abstract Data Model

This extension does not introduce any states or conceptual objects beyond those specified by [RFC4511].

3.2.2 Timers

None.

3.2.3 Initialization

This protocol extension requires no initialization beyond that specified in [RFC4511].

3.2.4 Higher-layer Triggered Events

None.

3.2.5 Message Processing Events and Sequencing Rules

3.2.5.1 Handling a Query for the supportedControl Attribute

The server MUST respond to a query for the supportedControl attribute as specified in [RFC4512]. For each of the following controls it supports, the server MUST return the corresponding OID value.

Object Identifier (OID) value	Server support
2.16.840.1.113730.3.4.9	Virtual List Support Server MUST implement [LDAPEX-SVB].
1.2.840.113556.1.4.319	Paged Results Support Server MUST implement [RFC2696].
1.2.840.113556.1.4.473	Server Sort Support Server MUST implement [RFC2891].

The server MAY return other OID values if it provides support for more capabilities than the ones specified in this protocol.

3.2.5.2 Handling a Query for the supportedCapabilities Attribute

The server MUST respond to a query for the supportedCapabilities attribute as specified in [RFC4511]. If the server supports **AD-type server** capabilities<13> as specified in this protocol, it MUST return the following OID value.

Object Identifier (OID) value	Server type
1.2.840.113556.1.4.800	AD-type server.

The server MAY return other OID values if it provides support for more capabilities than the ones specified in this protocol.

3.2.5.3 Handling Search Requests

3.2.5.3.1 Handling a Query for the defaultNamingContext Attribute

The server MAY<14> respond to a query for the attribute defaultNamingContext as specified in section 3.1.4.1.1. If the server returns a value for the defaultNamingContext, the server MUST return the **LDAP Distinguished Name** of the base object. The client MUST use the value returned by this query as the *baseObject* in future search requests.

3.2.5.3.2 Responding to Query Attributes

A server SHOULD<15> support the attributes specified in section 2.2.

3.2.6 Timer Events

None.

3.2.7 Other Local Events

None.

4 Protocol Examples

4.1 Simple Search Scenario

If the client is directed to search for a user named "John" in an AD-type **LDAP** server, the following sequence of events will occur:

- The client sends an **LDAP Bind Request** to the server.
    - **BindRequest (0x00):**
      Version: 3
      Name: Null
      authentication: Authentication type = sasl
- The **LDAP server** receives the request and returns a **Bind Response** to the client.
    - **BindResponse (0x01):**
      Status: Success
      MatchedDN: Null
      ErrorMessage: Null

- The client sends a **Search Request** to the server for the defaultNamingContext (as described in section 3.1.4.1.1).

  - **SearchRequest (0x03):**
    BaseObject: Null
    Scope: baseObject
    Alias: neverDerefAliases
    SizeLimit: 0 (no limit)
    TimeLimit: 0 (no limit)
    TypesOnly: False
    Filter: (objectClass=*)
    Attributes: (objectClass)(defaultNamingContext)

- The LDAP server returns the **Search Base** to the client in the defaultNamingContext attribute.

  - **SearchResultEntry (0x04):**
    ObjectNames: Null
    Attributes Returned:
    defaultNamingContext: (DC=company,DC=corp, DC=contoso,DC=com)

  - **SearchResultDone(0x05):**
    Status: Success
    MatchedDN: NULL
    ErrorMessage: NULL

- The client uses the **Search Base** and the simple query as specified in section 3.1.4.1.2 to send another **Search Request** to the server.

  - **Search Request (0x03):**
    BaseObject: (DC=company,DC=corp, DC=contoso,DC=com)
    Scope: WholeSubtree
    Alias: derefAlways
    SizeLimit: 100 entries
    TimeLimit: 60 seconds
    TypesOnly: False
    Filter:
    (&(|(mail=john*)(cn=john*)(sn=john*)(givenName=john*)(displayName=john*)))
    Attributes:
    (cn)(commonName)(mail)(roleOccupant)(display-name)(displayname)(sn)(surname)(co)(organizationName)(o)(givenName)(legacyExchangeDN)(objectClass)(uid)(mailNickname)(title)(company)(physicalDeliveryOfficeName)(telephoneNumber)

- The LDAP server returns results that match the query. (The trace below represents one result that matched the query.)

  - **SearchResultsEntry (0x04):**
    ObjectName: CN=John, OU=UsersOU, DC=company, DC=corp,

```
DC=contoso, DC=com
Attributes:
objectClass:              ( top ) ( person )
(organizationalPerson ) ( user )
cn:                       John Smith
sn:                       Smith
title:                    Dr.
physicalDeliveryOfficeName: 36/2495
telephoneNumber:          1 (425) 555-0534
givenName:                John
displayName:              John Smith
company:                  contoso
mailNickname:             jsmith
legacyExchangeDN:         /o=contoso/ou=First Admin
Group/cn=Recipents/cn=jsmith
mail:                     jsmith@contoso.com
```

- **SearchResultDone(0x05):**

```
Status:            Success
MatchedDN:         NULL
ErrorMessage:      NULL
```

- The client sends an **LDAP Unbind Request** to the server.
  - **UnbindRequest(0x02)**
- The client uses the attributes returned by the server to display search results to the user.

5 Security

5.1 Security Considerations for Implementers

There are no security considerations specific to this protocol extension beyond those specified in [RFC4511].

5.2 Index of Security Parameters

None.

6 Appendix A: Office/Exchange Behavior

The information in this specification is applicable to the following versions of Office/Exchange:

- Office 2003 with Service Pack 3 applied
- Exchange 2003 with Service Pack 2 applied
- Office 2007 with Service Pack 1 applied
- Exchange 2007 with Service Pack 1 applied

Exceptions, if any, are noted below. Unless otherwise specified, any statement of optional behavior in this specification prescribed using the terms SHOULD or SHOULD NOT implies Office/Exchange behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies Office/Exchange does not follow the prescription.

<1> Section 2.2: Outlook 2003 SP3 and Outlook 2007 SP1 query for the ‘o’ attribute, but do not use the value received from the server.

<2> Section 2.2: Outlook 2003 SP3 and Outlook 2007 SP1 query for this attribute, but do not use the value received from the server.

<3> Section 2.2.1.1: Outlook 2003 SP3 and Outlook 2007 SP1 consider a display name to be not user-readable if it is exactly the same as one of the E-Mail Address attributes. Outlook 2003 SP3 and Outlook 2007 SP1 will construct the display name in the following manner:

```
displayName ::= <common name> / <givenname> " "<surname> / <surname> /
<company name> / <email address> ; NOTE: Priority is given to non-empty
combinations listed first.

common name         ::=   ; Common Name LDAP Attribute
givenname           ::=   ; First Name LDAP Attribute
surname             ::=   ; Last name LDAP Attribute
company name        ::=   ; Organization Name LDAP Attribute
email address       ::=   ; E-Mail Address LDAP Attribute
```

<4> Section 2.2.3.1: Outlook 2003 SP3 and Outlook 2007 SP1 add the Exchange **distinguished name** to the list of proxy addresses (see section 2.2.3.2) if it is not there already. It adds the entry as:

```
proxyAddressAddition ::= "EX:" <Exchange DN>
<Exchange DN> ::= ; The value of the LDAP attribute legacyExchangeDN
```

The Exchange distinguished name is not used elsewhere in Outlook 2003 SP3 and Outlook 2007 SP1.

<5> Section 3.1.3.1: These controls are only used if the client supports browsing the server; if a client does not support this feature, it can choose to ignore them.

<6> Section 3.1.3.1: Outlook 2003 SP3 and Outlook 2007 SP1 use the following string as the VLV search filter “(&(mail=*)(cn=*))”.

<7> Section 3.1.3.1: Outlook 2003 SP3 and Outlook 2007 SP1 ignore all other values.

<8> Section 3.1.3.2: Outlook 2003 SP3 and Outlook 2007 SP1 ignore all other values.

<9> Section 3.1.4.1.1: In Outlook 2003 SP3, empty string **Search Bases** will trigger a defaultNamingContext query to the server. In Outlook 2007 SP1, empty string **Search Bases** will be used as empty strings.

<10> Section 3.1.4.1.1: Outlook 2003 SP3 attempts to verify the **Search Base** returned by the defaultNamingContext attribute. If a **Search Base** is deemed invalid, the subsequent search query will not take place. Outlook 2007 SP1 does not verify the **Search Base** and the search query will always take place.

<11> Section 3.1.4.1.2: Outlook 2003 SP3 does not implement Basic Search.

<12> Section 3.1.4.1.3: Outlook 2003 SP3 does not support E-Mail (**LDAP attribute** mail) in advanced searches.

<13> Section 3.2.5.2: An **AD-type server** will receive different attributes and filters from the client than a non-AD-type server. In general, the displayname and display-name attributes will be requested instead of the cn and commonname attributes. For more details about the client behavior, see sections 3.1.5.4 and 3.1.5.5.

<14> Section 3.2.5.3.1: Active Directory servers do respond to queries for the defaultNamingContext attribute.

<15> Section 3.2.5.3.2: The client can ask for more than one attribute representing the same conceptual data. For more details about which attributes the client can request, and the order of precedence used when handling return values, see section 3.1. A server is only required to return the value for one of the attributes corresponding to a piece of data requested by the client.

Index

Applicability statement, 6
Client details, 11
Glossary, 4
Index of security parameters, 19
Informative references, 5
Introduction, 4
Message syntax, 6
Messages, 6
  Message syntax, 6
  Transport, 6
Normative references, 4
Office/Exchange behavior, 19
Prerequisites/preconditions, 6
Protocol details, 11
  Client details, 11
  Server details, 16
Protocol examples, 17
  Simple search scenario, 17
Protocol overview, 6
References, 4
  Informative references, 5
  Normative references, 4
Relationship to other protocols, 6
Security, 19
  Index of security parameters, 19
  Security considerations for implementers, 19
Security considerations for implementers, 19
Server details, 16
Simple search scenario, 17
Standards assignments, 6
Transport, 6
Vendor-extensible fields, 6
Versioning and capability negotiation, 6