# [MS-GPOL]:
# Group Policy:
# Core Protocol Specification

---

**Intellectual Property Rights Notice for Open Specifications Documentation**

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

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

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

---

## Revision Summary

| Date | Revision History | Revision Class | Comments |
|---|---|---|---|
| 03/14/2007 | 1.0 | Major | Updated and revised the technical content. |
| 04/10/2007 | 1.1 | Minor | Updated the technical content. |
| 05/18/2007 | 2.0 | Major | New format |
| 06/08/2007 | 2.0.1 | Editorial | Revised and edited the technical content. |
| 07/10/2007 | 2.1 | Minor | Added glossary term; minor rewrites. |
| 08/17/2007 | 3.0 | Major | Updated and revised the technical content. |
| 09/21/2007 | 4.0 | Major | Updated and revised the technical content. |
| 10/26/2007 | 5.0 | Major | Updated and revised the technical content. |
| 01/25/2008 | 5.0.1 | Editorial | Revised and edited the technical content. |
| 03/14/2008 | 5.0.2 | Editorial | Revised and edited the technical content. |
| 06/20/2008 | 6.0 | Major | Updated and revised the technical content. |
| 07/25/2008 | 6.0.1 | Editorial | Revised and edited the technical content. |
| 08/29/2008 | 6.0.2 | Editorial | Revised and edited the technical content. |
| 10/24/2008 | 7.0 | Major | Updated and revised the technical content. |
| 12/05/2008 | 8.0 | Major | Updated and revised the technical content. |
| 01/16/2009 | 8.0.1 | Editorial | Revised and edited the technical content. |
| 02/27/2009 | 8.0.2 | Editorial | Revised and edited the technical content. |
| 04/10/2009 | 8.0.3 | Editorial | Revised and edited the technical content. |
| 05/22/2009 | 9.0 | Major | Updated and revised the technical content. |
| 07/02/2009 | 10.0 | Major | Updated and revised the technical content. |
| 08/14/2009 | 11.0 | Major | Updated and revised the technical content. |
| 09/25/2009 | 11.1 | Minor | Updated the technical content. |
| 11/06/2009 | 12.0 | Major | Updated and revised the technical content. |
| 12/18/2009 | 12.1 | Minor | Updated the technical content. |
| 01/29/2010 | 13.0 | Major | Updated and revised the technical content. |
| 03/12/2010 | 14.0 | Major | Updated and revised the technical content. |
| 04/23/2010 | 15.0 | Major | Updated and revised the technical content. |

| Date | Revision History | Revision Class | Comments |
|---|---|---|---|
| 06/04/2010 | 16.0 | Major | Updated and revised the technical content. |
| 07/16/2010 | 17.0 | Major | Significantly changed the technical content. |
| 08/27/2010 | 18.0 | Major | Significantly changed the technical content. |
| 10/08/2010 | 19.0 | Major | Significantly changed the technical content. |
| 11/19/2010 | 20.0 | Major | Significantly changed the technical content. |
| 01/07/2011 | 21.0 | Major | Significantly changed the technical content. |
| 02/11/2011 | 22.0 | Major | Significantly changed the technical content. |

# Contents

# 1 Introduction

This document specifies the Group Policy: Core Protocol, a protocol that communicates administrator-defined policies between a **domain member** and a **Group Policy server**.

## 1.1 Glossary

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

**access control entry (ACE)**
**Active Directory**
**Active Directory object**
**administrative tool**
**client-side extension GUID (CSE GUID)**
**computer account**
**computer policy mode**
**computer-scoped Group Policy object distinguished name**
**computer-scoped Group Policy object path**
**curly braced GUID string**
**discretionary access control list (DACL)**
**distinguished name (DN)**
**domain**
**domain account**
**domain controller (DC)**
**domain member (member machine)**
**domain name (2)**
**domain naming context (domain NC)**
**domain user**
**forest**
**fully qualified domain name (FQDN) (2)**
**globally unique identifier (GUID)**
**Group Policy object (GPO)**
**Group Policy object (GPO) container version**
**Group Policy object (GPO) distinguished name (DN)**
**Group Policy object (GPO) file system version**
**Group Policy object (GPO) GUID**
**Group Policy object (GPO) path**
**Group Policy object (GPO) version**
**Internet host name**
**Lightweight Directory Access Protocol (LDAP)**
**machine Group Policy object (GPO) version**
**policies path**
**policy application**
**policy setting**
**policy target**
**root directory system agent-specific entry (rootDSE)**
**scope of management (SOM)**
**scoped Group Policy object (GPO) path**
**scoped Group Policy object (GPO) distinguished name (DN)**
**site**
**site distinguished name (DN)**
**system volume (SYSVOL)**

**tool extension GUID or administrative plug-in GUID**
**Unicode**
**Unicode string**
**universally unique identifier (UUID)**
**user GPO version**
**user policy mode**
**user principal name (UPN)**
**user-scoped Group Policy object distinguished name (DN)**
**user-scoped Group Policy object path**
**WMI Query Language (WQL)**

The following terms are specific to this document:

**Client:** In this document, the capitalized use of this term refers to a **domain** member, including the **domain controller (DC)**, that is involved in a **policy application** sequence.

**directory string:** A string encoded in UTF-8 as defined in [RFC2252] section 6.10.

**enforced Group Policy object (GPO):** A **Group Policy object (GPO)** that is specifically associated with a **scope of management (SOM)** so that the associated **GPO** has a higher **GPO precedence** compared to other **GPOs** that are associated with the same **SOM** and compared to **GPOs** that are associated with descendant **SOMs**. An enforced **GPO** cannot be blocked by a descendant **SOM** using the gpOptions attribute.

**Group Policy Extension:** A network protocol that depends on the Group Policy: Core Protocol and defines a **client-side extension GUID** or a **tool extension GUID**.

**Group Policy object (GPO) distinguished name (DN) list:** An ordered set of scoped **GPO DNs**, one for each **GPO** for which a **Group Policy Extension** is to request and retrieve settings. Each element in the list corresponds to one of the elements in the corresponding **GPO path list**. An element in the **GPO DN list** corresponds to an element in the **GPO path list** if both elements have the same ordinality in their respective lists.

**Group Policy object (GPO) path list:** An ordered set of scoped **GPO paths** in which each element in the set corresponds to one of the elements in the **GPO DN list** set. An element in the **GPO path list** corresponds to an element in the **GPO DN list** if both elements have the same ordinality within their respective lists. Each **GPO path** specifies the file system path for the **GPO** with the corresponding element in the **GPO DN list**.

**Group Policy object (GPO) precedence:** An ordering between the **GPOs** that are associated with a **policy target**. A **policy setting** defined in a **GPO** that has a lower precedence can be overridden by a **policy setting** defined in a **GPO** that has a higher precedence.

**Group Policy (GP) server:** A server that holds a database of Group Policy objects (GPOs) that other machines can retrieve. The **GP server** must be a **domain controller (DC)**.

**link order:** An integer that describes the precedence of a **GPO** that is associated with a **scope of management (SOM)** when compared to other **GPOs** that are associated with that **SOM**. A **GPO** that has a smaller **link order** associated with an **SOM** has higher **GPO precedence** than a **GPO** that has a higher **link order** associated with the same **SOM**.

**policy source:** The **LDAP distinguished name** of an Active Directory account object that is used to compute a **GPO** list.

**server:** In this document, the capitalized use of this term refers to the **GP server** that is involved in a **policy application** sequence.

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

## 1.2 References

### 1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information. Please check the archive site, http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624, as an additional source.

[C706] The Open Group, "DCE 1.1: Remote Procedure Call", C706, August 1997, http://www.opengroup.org/public/pubs/catalog/c706.htm

[MS-ADA1] Microsoft Corporation, "Active Directory Schema Attributes A-L", June 2007.

[MS-ADA2] Microsoft Corporation, "Active Directory Schema Attributes M", July 2006.

[MS-ADA3] Microsoft Corporation, "Active Directory Schema Attributes N-Z", July 2006.

[MS-ADLS] Microsoft Corporation, "Active Directory Lightweight Directory Services Schema", June 2007.

[MS-ADSC] Microsoft Corporation, "Active Directory Schema Classes", July 2006.

[MS-ADSO] Microsoft Corporation, "Active Directory System Overview", July 2009.

[MS-ADTS] Microsoft Corporation, "Active Directory Technical Specification", July 2006.

[MS-DFSC] Microsoft Corporation, "Distributed File System (DFS): Referral Protocol Specification", July 2006.

[MS-DISO] Microsoft Corporation, "Domain Interactions System Overview", November 2009.

[MS-DRSR] Microsoft Corporation, "Directory Replication Service (DRS) Remote Protocol Specification", July 2006.

[MS-DTYP] Microsoft Corporation, "Windows Data Types", January 2007.

[MS-GPFR] Microsoft Corporation, "Group Policy: Folder Redirection Protocol Extension", July 2006.

[MS-GPSI] Microsoft Corporation, "Group Policy: Software Installation Protocol Extension", July 2006.

[MS-GPIPSEC] Microsoft Corporation, "Group Policy: IP Security (IPsec) Protocol Extension", July 2006.

[MS-GPREG] Microsoft Corporation, "Group Policy: Registry Extension Encoding", July 2006.

[MS-GPSCR] Microsoft Corporation, "Group Policy: Scripts Extension Encoding", July 2006.

[MS-KILE] Microsoft Corporation, "Kerberos Protocol Extensions", July 2006.

[MS-NLMP] Microsoft Corporation, "NT LAN Manager (NTLM) Authentication Protocol Specification", July 2006.

[MS-NRPC] Microsoft Corporation, "Netlogon Remote Protocol Specification", March 2007.

[MS-SPNG] Microsoft Corporation, "Simple and Protected GSS-API Negotiation Mechanism (SPNEGO) Extension", July 2006.

[MS-WMI] Microsoft Corporation, "Windows Management Instrumentation Remote Protocol Specification", September 2007.

[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

[RFC2251] Wahl, M., Howes, T., and Kille, S., "Lightweight Directory Access Protocol (v3)", RFC 2251, December 1997, http://www.ietf.org/rfc/rfc2251.txt

[RFC2252] Wahl, M., Coulbeck, A., Howes, T., and Kille, S., "Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions", RFC 2252, December 1997, http://www.ietf.org/rfc/rfc2252.txt

[RFC2254] Howes, T., "The String Representation of LDAP Search Filters", RFC 2254, December 1997, http://www.ietf.org/rfc/rfc2254.txt

[RFC2256] Wahl, M., "A Summary of the X.500(96) User Schema for use with LDAPv3", RFC 2256, December 1997, http://www.ietf.org/rfc/rfc2256.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

[RFC792] Postel, J., "Internet Control Message Protocol", RFC 792, September 1981, http://www.ietf.org/rfc/rfc792.txt

### 1.2.2 Informative References

[MS-GLOS] Microsoft Corporation, "Windows Protocols Master Glossary", March 2007.

## 1.3 Overview

The Group Policy: Core Protocol is a client/server protocol that allows **Clients** to discover and retrieve **policy settings** that administrators of a **domain** create. Policy settings are administrative directives that administrators make regarding the behavior of the Clients. For example, an administrator might want to configure every computer in a certain group of computers to open a specific port in their firewall. That administrator can use Group Policy to state that directive, and it will eventually be communicated to the Clients through the Group Policy: Core Protocol.

### 1.3.1 User and Computer Policy Settings

The behavior of the Clients fall into two categories: user policy settings, and computer policy settings.

User policy settings specify behavior for interactively logged-on users and can potentially affect different users who are logged on to the same computer. There are also settings that should affect that user no matter what computer the user logs on to. Such settings include the desktop background image for a user or the user's default location for saving documents.

Computer policy settings are either behaviors that can affect the computer (even when no users are logged on to the computer) or settings that globally affect every user who is logged on to the computer. Examples of such settings include a setting that enables a computer to host a Web

server, that schedules automated disk backups of the computer, or that specifies a standard Web home page for all users of the computer.

The Group Policy: Core Protocol does not define any policy settings itself. A vendor defines settings by implementing **Group Policy Extensions** or by using data-driven Group Policy Extensions (such as the Group Policy: Registry Extension Encoding, as specified in [MS-GPREG]) that allow for the definition of new settings.

### 1.3.2 Protocol Operational Modes

The Group Policy: Core Protocol has two primary modes of operation: policy application and policy administration. In **policy application** driven by the Client, the Client retrieves administrator-specified behaviors from the GP server. The other operational mode is policy administration driven by the administrator, in which administrator-operated tools on a remote computer can be used to modify the behavioral specifications (or policies) that the GP server returns when the Client performs policy application.

### 1.3.3 Policy Application

The Client's interaction with the GP server in policy application follows a pull model in which the Client polls a GP server to check for new behavioral specifications from administrators. The settings that are retrieved through policy application are intended to affect either the Client computer itself or a **domain user** that is interactively logged on to the Client. Because of this, policy application operates in two modes: a **computer policy mode** that retrieves computer policy settings that are based on the Client computer's account, and a **user policy mode** that retrieves user policy settings that are based on the account of a domain user who is interactively logged on to the computer. The account for which the settings are being retrieved is called the **policy target**. For computer policy mode, the policy target is always the Client computer's domain account; for user policy mode, the policy target is the account of a domain user who is interactively logged on to the Client.

#### 1.3.3.1 Server Discovery and Group Policy Object Association

Policy application starts with a discovery step that is based on locating a domain controller as specified in section 3.2.5.1.1 in order to identify a **DC**. The Client initiates this step. After a domain controller is located, the Client performs two sets of queries on the directory of the GP server by using the **Lightweight Directory Access Protocol (LDAP)**.

The purpose of the first set of queries is to determine what sets of behavior specifications, called **Group Policy objects (GPOs)**, have been assigned to the policy target account (that is, the GPOs that an administrator has configured as being applicable to the policy target account). Because domain accounts are stored in Active Directory, information about the GPOs that are associated with those accounts is also stored there.

Domain accounts are stored as objects in Active Directory in a hierarchy of organizational unit containers that is rooted in a container for the domain itself. Each of these containers can also specify a set of GPOs, and this association means that the set of GPOs applies to all accounts in the same container. Thus, the first set of queries performs a search on the hierarchy of the policy target account in order to identify the associated set of GPOs.

#### 1.3.3.2 GPO Retrieval

The second set of queries assembles the logical GPO from its component parts that include its **Active Directory** portion and its file system-based portion. This second set of queries is also performed through the LDAP, and it uses the names of GPOs that are returned in the first search to

perform a query that returns detailed attributes for each of the GPOs that are associated with the policy target. These attributes describe details such as the following:

- Precedence between GPOs to allow for resolution of conflicts between different GPOs (for example, if one GPO requests to set the background to green and another requests to set it to blue).
- Information used for filtering to allow exclusion of some accounts in a container from being associated with a GPO.
- Identification of classes of settings that are contained within a GPO.
- Version information on the Active Directory portion of the GPO.
- Location of information for that GPO stored outside Active Directory on the GP server's **SYSVOL** domain-based Distributed File System (DFS) share, as specified in [MS-DFSC] section 3.1.5.4.4.

The Client also uses file access to query the SYSVOL share for a file that contains version information for the file system storage portion of the GPO. The Client uses all of this information to decide which of the GPOs have certain classes of settings that require protocol activity in the next and final step of policy application.

#### 1.3.3.3 Group Policy Extension Settings Retrieval

The last step of policy application is the actual retrieval of settings. In this step, the Client uses its computed list of GPOs that contain different classes of settings to invoke a protocol sequence that is specific to each class of settings called a Group Policy Extension (for example, the Group Policy: Registry Extension Encoding, as specified in [MS-GPREG]). Such an invocation is done by using a unique **client-side extension GUID (CSE GUID)** in the GPO to identify the class. The Group Policy Client then executes the plug-in code (which is associated with that CSE GUID on the Client) that obtains the Group Policy Extension's settings from the GPO through a protocol exchange with the GP server and that interprets those settings in a specific manner. The GP Client itself has no knowledge of the internal details of specific Group Policy Extensions.

These Group Policy Extensions retrieve the settings of their specific classes that are stored in each GPO, typically by using LDAP to access the Active Directory storage portion of the GPO on the GP server or by reading or writing the file system portion of the GPO on the GP server, or both. After the settings are retrieved, the Group Policy Extension plug-in on the Client can interpret the settings and enforce the behaviors they specify.

### 1.3.4 Policy Administration

In policy administration mode, an **administrative tool** locates the GP server, as specified in section 1.3.3.1, and operates on the same **Active Directory objects** as policy application. Instead of applying policy settings locally, policy administration allows an administrator to create, update, and delete policy settings, and then updates the GP server by using the LDAP.

Just as policy application supports Group Policy Extension plug-ins on the Client, to consume settings of a given mode, policy administration supports Group Policy Extension plug-ins to the administrative tool for authoring Group Policy Extension-specific settings. GPOs with settings for a particular Group Policy Extension are identified with a **tool extension GUID** to enable administrative tools to identify a plug-in that is capable of administering the settings. Such Group Policy Extensions (for example, as specified in [MS-GPREG]) typically use LDAP to store settings in Active Directory, or they store settings in files.

## 1.4 Relationship to Other Protocols

Note that the Group Policy: Core Protocol by itself is not capable of communicating policy settings directly. The Group Policy: Core Protocol only does so by being extended by one or more protocol Group Policy Extensions (for example, as specified in [MS-GPREG], [MS-GPSCR], and [MS-GPIPSEC]) that are capable of communicating policy settings of a given class. These Group Policy Extensions depend on the Group Policy: Core Protocol to execute first on the Client to identify GPOs that the Group Policy Extension should query or update. The Group Policy: Core Protocol has no dependency on any such Group Policy Extensions. Any number of such Group Policy Extensions can be added without requiring any changes to the Group Policy: Core Protocol.

The Group Policy depends on the following protocols to allow it to exchange information between a Client and a GP server:

- Simple and Protected Generic Security Service Application Program Interface Negotiation Mechanism (SPNEGO) Protocol Extensions, as specified in [MS-SPNG], for authentication.
- Kerberos Protocol Extensions, as specified in [MS-KILE], for authentication.
- NT LAN Manager (NTLM) Authentication Protocol, as specified in [MS-NLMP], for authentication.
- DFS: Referral Protocol, as specified in [MS-DFSC], to provide location-independent access to the GP server for Clients during policy application and policy administration.
- LDAP v3, as specified in [RFC2251], for transmitting Group Policy settings and instructions between the Client and the GP server.
- DRS Remote Protocol, as specified in [MS-DRSR] is used for the DN Discovery.
- Netlogon Remote Protocol, as specified in [MS-NRPC], is used for DC Discovery.

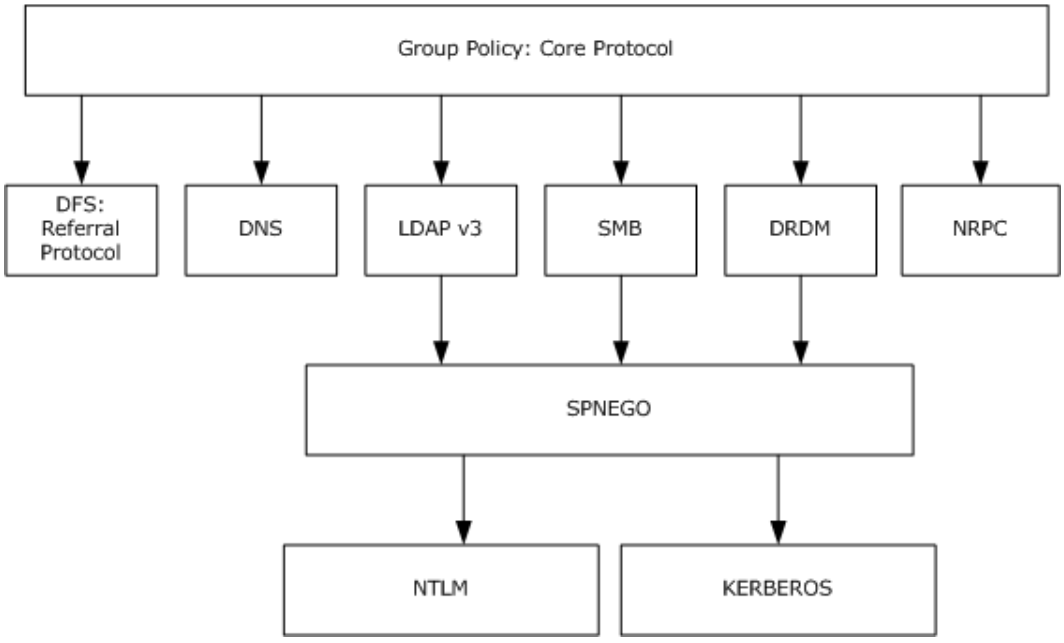


**Figure 1: Group Policy: Core Protocol relationship diagram**

The Internet Control Message Protocol (ICMP), as specified in [RFC792], MAY be used for Link Speed Determination.

## 1.5 Prerequisites/Preconditions

Preconditions for Group Policy: Core Protocol communications between a Client and **Server** are the following:

- The Server is assumed to be a DC.
- The Client must be joined to the Server domain.
- For user policy mode, the Client must be joined to a domain for which the user domain has a bidirectional domain trust.
- All DCs in the domain must be configured to require signing of LDAP traffic, as specified in [RFC2251] section 4.2.2.

## 1.6 Applicability Statement

The Group Policy: Core Protocol is only applicable for communicating administrative directives to Clients. Because the Group Policy: Core Protocol itself is not encrypted, it SHOULD NOT be used to directly transmit directives that are sensitive and must be sent securely (such as password information that Clients use to access resources).

The Group Policy: Core Protocol is not applicable if an administrator requires explicit acknowledgment that the policy settings have been retrieved or enforced by the Client computers.

The Group Policy: Core Protocol is not applicable if different settings need to be applied to each Client. This protocol is intended for applying the same settings to large groups of Clients.<1><2><3>

## 1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

- Protocol versions: The Group Policy: Core Protocol provides a versioning capability in the **gPCFunctionalityVersion** attribute of the Active Directory object class for a Group Policy Object (GPO) specified in section 2.2. There is no capability negotiation that is associated with this version. The version itself is a simple integer. There is only one version currently, and if the Client receives anything other than that version for a GPO, the GPO does not participate in this protocol, as specified in section 3.2.5.1.5.
- Security and authentication methods: The Group Policy: Core Protocol supports the following authentication methods: NT LAN Manager protocol (NTLM) and Kerberos. The authentication method in use is negotiated using the mechanisms specified in [MS-SPNG].

## 1.8 Vendor-Extensible Fields

The Group Policy: Core Protocol allows vendors to define Group Policy Extensions to the protocol. These Group Policy Extensions enable vendors to store vendor-specific data in a GPO on the GP server. For the Client to access that data, it must be able to identify a system component that can retrieve and interpret that data.

To facilitate this, the GPO Active Directory object schema has two attributes, **gPCMachineExtensionNames** and **gPCUserExtensionNames**, in which a vendor can append

both a CSE GUID that identifies that GPO as having that vendor's particular extra Group Policy Extension data stored inside it, and a tool extension GUID that allows the vendor to associate an administrative tool that can update the data. The vendor obtains the **UUIDs** of the CSE GUID and the tool extension GUID by generating them according to the standard **GUID** algorithm, as specified in [C706]. After they are generated, the vendor SHOULD include the GUID in these attributes, as specified in section 2.2. Vendors do not need to collaborate or obtain GUIDs from a central authority; the GUID generation algorithm ensures that no two vendors make use of the same GUID.

Each CSE GUID and tool extension GUID defined by a vendor MUST be treated as a standards assignment to the **gPCMachineExtensionNames** and **gPCUserExtensionNames** attributes that MUST be declared in the Group Policy Extension documentation that is associated with the CSE GUID and tool extension GUID.

## 1.9 Standards Assignments

There are no standards assignments for the Group Policy: Core Protocol.

# 2 Messages

## 2.1 Transport

The Group Policy: Core Protocol is a sequence of protocol conversations using different transports. The initial protocol conversation locates the GP Server specified in section 3.2.5.1.1.

Subsequent messages are exchanged by using a combination of file access and LDAP. The Group Policy: Core Protocol allows Clients and administrative tools to access policy instructions stored on the GP server. The Client and administrative tools use file access and LDAP as transports to access that storage, which itself is split between network file system storage and Active Directory. Group Policy defines specific file formats and directory structure layouts that define the structure of the file system storage.

Similarly, Group Policy also defines objects with specific schemas that are stored in Active Directory of the GP server, and Clients and administrative tools use LDAP to access Active Directory to obtain these structured objects. Almost all of the data that is exchanged in a Group Policy protocol conversation consists of file access and LDAP as the transports for conveying the Group Policy: Core Protocol.

For the structure of the files and Active Directory objects, see section 2.2.

## 2.2 Message Syntax

The Group Policy: Core Protocol is an amalgam of protocol conversations. For the purposes of this document, different phases of this conversation are described as messages. These messages are themselves bidirectional; that is, they can contain multiple pairs of both requests and responses.

There are two classes of protocol conversations. Each message can be categorized into one of the following two classes:

- Policy application messages
- Administrative messages

Policy application messages are exchanged during policy application after which a Group Policy Extension typically takes action to apply administrative policy. Collectively, the following sequence of eight messages is referred to in this documentation as a policy application message:

1. Distinguished Name (DN) Discovery
2. Domain Scope of Management (SOM) Search
3. Site Search
4. Group Policy Object (GPO) Search
5. WMI Filter Search
6. Link Speed Determination

Administrative messages allow an administrator to view and update policies in a domain. They are only used by an administrative plug-in, never by a Client plug-in. Administrative messages consist of the following:

- GPO Read Administration

- GPO Write Administration

**Note** All usage of file access and LDAP in the following message syntaxes include SPNEGO messages in the appropriate part of the protocol sequences. For computer policy mode, they MUST include Kerberos authentication.

The authentication requirements mean that for user policy mode, if the Client needs the settings for a policy target, the Client MUST be able to authenticate all LDAP and file operations against the Server as the policy target account. Thus all LDAP and file operations that can be authenticated include authentication traffic that authenticates the policy target against the Server.

**Note** All references in this document to object **distinguished names (DN)** and attribute names through LDAP correspond exactly to objects and attributes that are stored on the DC LDAP Server, according to the Active Directory schema, as specified in [MS-ADSC], [MS-ADA1], [MS-ADA2], and [MS-ADA3].

The Group Policy: Core Protocol provides a Group Policy Extension mechanism that allows other protocols to insert themselves into this protocol's sequences; each such Group Policy Extension has its own document (for example, [MS-GPREG] and [MS-GPSCR]). Note that the Group Policy: Core Protocol does not require any of these Group Policy Extensions; for example, vendors can use this protocol with only their own Group Policy Extensions.

### 2.2.1 DN Discovery

DN Discovery uses the DRS Remote Protocol, as specified in [MS-DRSR]. It is authenticated using SPNEGO, as specified in [MS-SPNG].

The message syntax of the traffic the query generates is specified in [MS-DRSR] section 4.1.3 for the remote procedure call (RPC) method, DRSCrackNames. The Client makes the call to the GP server with the dwInVersion set to 1 with a DRS_MSG_CRACKREQ *pMsgIn* structure parameter that passes in the specified account name in the format DS_NT4_ACCOUNT_NAME, as specified in [MS-DRSR] section 4.1.4.1.3. As specified in [MS-DRSR], the method returns a code of 0 if it is successful with a DRS_MSG_CRACKREPLY structure that contains a DS_NAME_RESULTW structure, which in turn contains an array of DS_NAME_RESULT_ITEMW structures, each of which corresponds to a requested name. Inside each DS_NAME_RESULT_ITEMW structure is a **pName** field that contains the fully qualified distinguished name for the corresponding requested account.

The detailed specification of the *pMsgIn* parameter is as follows.

| Field | Value |
|---|---|
| **CodePage** | MUST be set to 0. |
| **LocaleId** | MUST be set to 0. |
| **dwFlags** | MUST be set to 0. |
| **formatOffered** | 2<br>**Note** In this DRS_MSG_CRACKREQ structure sent by the Client to the GP server, one of the elements in the **rpNames** parameter MUST be of the form DS_NT4_ACCOUNT_NAME, as specified in [MS-DRSR] section 4.1.4.1.3. Any or all other formats specified in [MS-DRSR] section 4.1.4.1.2 MAY also be included. These other optional formats MAY be ignored by Group Policy: Core Protocol implementations. |
| **formatDesired** | 1<br>**Note** According to the syntax specified in [MS-DRSR] section 4.1.4.1.3, if one of the |

| Field | Value |
|---|---|
| | elements in the rpNames parameter is a valid account name of the form DS_NT4_ACCOUNT_NAME, then the implementation of DRSCrackNames MUST return a fully qualified distinguished name in the corresponding DS_NAME_RESULT_ITEMW structure inside the DS_NAME_RESULTW structure that this method returns when it completes successfully. If, however, formatDesired is set to a value other than 1, the implementation MUST return DS_NAME_ERROR_NO_MAPPING in every DS_NAME_RESULT_ITEMW structure in rItems.<4> |
| **cNames** | MUST be greater than or equal to 1. |
| **rpNames** | At least one of the names in the rpNames array MUST contain the account name to be cracked, in the DS_NT4_ACCOUNT_NAME format. |

Protocol details of this RPC method are specified in [MS-DRSR] section 4.1.4.

**Note** The DSR Remote Protocol, as specified in [MS-DRSR], itself supports caching the results of this message, so this message might not always appear in the protocol sequence for policy application.

### 2.2.2 Domain SOM Search

The Domain SOM Search message uses LDAP as a transport. The purpose of this message is to allow the Client to query the GP server for SOMs that are associated with the policy target account.

An LDAP SearchRequest MUST be sent to the GP server with the following parameters.

| Parameter | Value |
|---|---|
| *baseObject* | LDAP DN for the root of the domain. |
| *scope* | MUST be the whole subtree (2). |
| *derefAliases* | MUST be set to 0 (neverDerefAliases). |
| *sizeLimit* | No limit is set (this is set to 0 by default). |
| *timeLimit* | MAY be 0 (infinite), but SHOULD be 240 (seconds).<5> |
| *typesOnly* | MUST be set to 0. |
| *filter* | The following LDAP filter (using the representation specified in [RFC2254]) MUST be used:<br>(\|(distinguishedName=<OUPath1>)(distinguishedName=<OUPath2>)...<br>(distinguishedName=<DomainRoot DN>))<br>Where <OUPath1> and <OUPath2> are LDAP DNs for an object of type organizationalUnit, <DomainRoot DN> is the DN of the root of the domain, and all other characters are to be taken literally. |
| *attributes* | The following literal attribute names MUST be passed as inputs to the LDAP search request, and the following attributes are of the domain and organizational unit Active Directory containers (that is, **SOMs**): **gpLink** and **gpOptions**. |

A successful reply from the LDAP search request MUST contain one or more LDAP searchResponse messages. Those messages MUST contain one or more searchResultEntries. Those searchResultEntries MUST contain an objectName DN attribute, which is the SOM named by that DN. The searchResultEntry MUST also contain an attributes field with the values in Active Directory for

the gpLink and gpOptions attributes of the SOM objects that were searched for. The attributes MUST have the following formats:

gpLink: MUST be a Directory String encoded in UTF-8 as defined in [RFC2252] section 6.10 with the following format:

[<GPO DN_1>;<GPLinkOptions_1>][<GPO DN_2>;<GPLinkOptions_2>]...
[<GPODN_n>;<GPLinkOptions_n>]

where "[", "]" and ";" are to be taken literally, <GPO DN*> are **GPO DNs**, and <GPLinkOptions> is a bit field with the following flags (any bitwise combination of the flag values is valid) defining the state of the association of the GPO referenced by the GPO DN with this and only this SOM:

| Value | Meaning |
|---|---|
| 0x00000000 | The GPO Link preceding the <GPLinkOptions> field is not ignored and is not an **enforced GPO**. This is the default <GPLinkOptions> value. |
| 0x00000001 | The GPO Link preceding the <GPLinkOptions> field MUST be ignored. |
| 0x00000002 | The GPO Link preceding this <GPLinkOptions> is an enforced GPO. |
| 0x00000003 | The GPO Link preceding the <GPLinkOptions> field MUST be ignored; in other words, when the 0x00000001 bit is set, the 0x00000002 bit is ignored, and the behavior is the same as if the flag value were 0x00000001. |

**Note** The presence of the GPO DNs in the gpLink attribute of the SOM from which it came defines an association of the GPO DNs with the SOM. The order in which **GPO paths** appear in this attribute specifies the **link order** for the associated GPOs. A GPO can be linked one or more times to a SOM object, and the <GPLinkOptions> field can be configured independently on each of the links.

gpOptions: This is an LDAP INTEGER (as defined in [RFC2252] section 6.16). It is used to block Group Policy inheritance. A value of "1" for this attribute in a given SOM container means that non-enforced GPO links to SOM objects higher in the Active Directory hierarchy of this SOM container MUST be ignored. GPO links to the SOM object in which this attribute is set to "1" are not affected. A value of "0" means that GPOs in this SOM's container hierarchy in the Active Directory MUST be honored. The default value is "0".

### 2.2.3 Site Search

The purpose of this message is to allow the Client to query the GP server for SOMs that are associated with the **site** that is associated with the Client computer's account, because a site is also considered a SOM with relevance to the Group Policy: Core Protocol.

An LDAP **SearchRequest** MUST be sent to the GP server with the following parameters:

| Parameter | Value |
|---|---|
| *baseObject* | Zero-length string (meaning **rootDSE** DN as defined in [MS-ADTS] section 1.1). |
| *Scope* | MUST be set to 0. Search the base entry only. Exclude entries below the base. |
| *derefAliases* | MUST be set to 0 (neverDerefAliases). |
| *sizeLimit* | MUST be set to 1 (the *Scope* parameter limits search to the base entry only and therefore, at most one entry can be returned.) |

| Parameter | Value |
|---|---|
| *timeLimit* | MAY be 0 (infinite), but SHOULD be 240 (seconds).<6> |
| *typesOnly* | MUST be set to 0 (FALSE). |
| *Filter* | The following LDAP filter (using the representation as specified in [RFC2254]) MUST be used:<br>(objectClass=*) |
| *attributes* | configurationNamingContext, nTSecurityDescriptor |

As specified in [RFC2251], a reply from the LDAP **SearchRequest** is received by the Client from the GP server with one LDAP searchResponse message. That message contains **searchResultEntries** which contain an **attributes** field with the values **nTSecurityDescriptor**, as specified in [MS-DTYP] section 2.4.6, and **configurationNamingContext**, from the rootDSE DN as defined in [MS-ADTS] section 1.1. The type of this value is a distinguishedName. From this value and the SiteName value, the **site distinguished name (DN)** can be computed. This computation is specified in section 3.2.5.1.4.

Another **SearchRequest** is made with the following parameters:

| Parameter | Value |
|---|---|
| *baseObject* | Base Search Scope is the site DN. This computation is specified in section 3.2.5.1.4. |
| *Scope* | Search only the root of the computer's domain (this MUST be set to 0). |
| *derefAliases* | MUST be set to 0 (neverDerefAliases). |
| *sizeLimit* | No limit is set (this MUST be set to 0). |
| *timeLimit* | MAY be 0 (infinite), but SHOULD be 240 (seconds).<7> |
| *typesOnly* | MUST be set to 0 (FALSE). |
| *Filter* | The following LDAP filter (using the representation as specified in [RFC2254]) MUST be used:<br>(objectClass=*) |
| *Attributes* | **gpLink** and **gpOptions** attributes. |

The searchResponse received MUST meet the same requirements as those specified in the Domain Scope of Management Search (section 2.2.2).

### 2.2.4 GPO Search

The GPO Search message uses file access and LDAP as transports. The purpose of this message is to allow the Client to query the GPOs that are associated with SOMs.

An LDAP SearchRequest MUST be sent to the GP server with the following parameters.

| Parameter | Value |
|---|---|
| *baseObject* | cn=policies,cn=system,<LDAP DN for the root of the domain.> |

| Parameter | Value |
|---|---|
| *Scope* | Search entire subtree (this MUST be set to 2). |
| *derefAliases* | MUST be set to 0 (neverDerefAliases). |
| *sizeLimit* | SHOULD be set to 65536. |
| *timeLimit* | MAY be 0 (infinite), but SHOULD be 240 (seconds).<8> |
| *typesOnly* | MUST be set to 0 (FALSE). |
| *Filter* | The following LDAP filter (using the representation as specified in [RFC2254]) MUST be used:<br>(\|(**distinguishedName**=<GPOPath1>)(**distinguishedName**=<GPOPath2>)...(**distinguishedName**=<GPOPathN>))<br>where <GPOPath1> and <GPOPathN> are the GPO DNs (as specified in sections 2.2.2 and 2.2.3) without the prefix "LDAP://"; all other characters are to be interpreted literally. |
| *attributes* | nTSecurityDescriptor, cn, displayName, gPCFileSysPath, versionNumber, gPCMachineExtensionNames, gPCUserExtensionNames, gPCFunctionalityVersion, flags, gPCWQLFilter, and objectClass. |

The Client receives a reply from the search request from the GP server with one or more LDAP searchResponse messages. Those messages contain one or more **searchResultEntries**. Those **searchResultEntries** MUST contain an **objectName** DN attribute that is the GPO named by that DN. The **searchResultEntry** also MUST contain an **attributes** field with the values in Active Directory for the attributes of the GPOs that were searched for. The attributes MUST have the following format.

| Attribute | Format |
|---|---|
| **nTSecurityDescriptor** | A security descriptor whose format is specified in [MS-DTYP] section 2.4.6. |
| **cn** | The common name of the GPO; all GPO common names are **curly braced GUID strings** of the form {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}. |
| **displayName** | A human-readable **directory string** description of the GPO. |
| **gPCFileSysPath** | A GPO path. |
| **versionNumber** | A **GPO container version**. It is a 32-bit integer which consists of 16 bits of **user GPO version** and 16 bits of **machine GPO version**. |
| **gPCMachineExtensionNames** | A directory string with the format:<br>[<CSE GUID1><TOOL GUID1>][<CSE GUID 2><TOOL GUID2>]<br>where <CSE GUIDn> is a CSE GUID and <TOOL GUIDn> is a tool extension GUID, and the "[" and "]" characters are to be interpreted literally. The CSE GUID and tool extension GUID are each a 38-character "curly braced GUID string" as defined in [MS-GLOS]. |
| **gPCUserExtensionNames** | This attribute has the same format as **gPCMachineExtensionNames**. |
| **gPCFunctionalityVersion** | A 32-bit integer, as specified in section 1.7. This MUST be set to 2 for the GPO to be included in the protocol sequence; any other value |

| Attribute | Format |
|---|---|
| | means the GPO MUST be considered denied. |
| **flags** | A 32-bit integer that is interpreted as a flags bit field. Any bitwise combination of the following two flag values is valid. The Client MUST ignore any other flags:<br>If no bits are set (0x00000000): This GPO is enabled for both user and computer policy mode.<br>If bit 0 is set (0x00000001): Ignore this GPO for user policy mode.<br>If bit 1 is set (0x00000002): Ignore this GPO for computer policy mode.<br>If both bits are set (0x00000003): This GPO is disabled for both user and computer policy mode. |
| **gPCWQLFilter** | A directory string of the format:<br>"["<DOMAIN NAME>";"<WMI FILTER ID>";"<FLAGS>"]"<br>where "[", ";", and "]" are to be included literally, and where <WMI FILTER ID> is the identifier of the WMI filter, <DOMAIN NAME> is the **fully qualified domain name (FQDN)** of the domain in which the WMI filter is defined, and <FLAGS> MUST be ignored by the Client. |
| **objectClass** | Name of the Active Directory object class type. |

For each GPO successfully retrieved in each search, <gpo path>\gpt.ini is a file on the Server. The directory <gpo path> corresponds to the file system path retrieved for the GPO in the **gPCFileSysPath** attribute of the search.

The format of the file is as follows:

The gpt.ini file MUST be a **Unicode** file that can be described with the following Augmented Backus-Naur Form (ABNF), as specified in [RFC4234].

```
IniFile = WhiteSpace Sections WhiteSpace
Sections = Section / Sections Section
WhiteSpaceClass = %x0009 / %x0020
WhiteSpace = *WhiteSpaceClass
LineBreak = CR / LF / CRLF
IniId = 1*ALPHAKey
Id = IniIdIniValue = 1*(ALPHA / "_" / DIGIT )
Section = SectionId Keys
Keys = Key / Keys Key
SectionId = "[" SectionName "]" WhiteSpace LineBreak
SectionName = 1*SectionChar
SectionChar = ALPHA / "_" / WSP
Key = KeyId WhiteSpace "=" WhiteSpace IniValue WhiteSpace
      LineBreak
```

Abstractly, the file is described as having unique sections that correspond to the section tags in the previous code example. Each section MUST have a unique SectionId. The Key tags that are part of the definition of section define abstract "Keys" that MUST be unique within that abstract section only, defined by their associated KeyId tags. When testing uniqueness of the KeyId and SectionId tags, case MUST be ignored.

Using the sections, keys, and values terminology of this documentation, the specific format of gpt.ini can be specified as follows:

- Sections: The file MUST have the required section, "General". If this section is not present, the file is considered corrupt, and the protocol exchange MUST be terminated.
- Keys: The required key, "Version", MUST exist under the "General" section.
- Value: The value of the key, "Version", MUST be a 32-bit integer that corresponds to a GPO version. This is where the GPO file system version is defined. It is a 32-bit integer which consists of 16 bits of user GPO version and 16 bits of machine GPO version.

### 2.2.5 WMI Filter Search

The WMI Filter Search message uses LDAP as a transport. The purpose of this message is to allow the Client to query the GP server for filters using a WQL Query (as specified in [MS-WMI] section 2.2.1) that additionally constrain the set of GPOs that Group Policy Extensions are to use.<9>

An LDAP SearchRequest MUST be sent to the GP server with the following parameters.

| Parameter | Value |
|---|---|
| *baseObject* | CN=<WMI FILTER ID>,CN=SOM,CN=WMIPolicy,CN=System, <DomainRoot DN><br>where <WMI FILTER ID> is the identifier of the WMI filter, and <DomainRoot DN> is the DN of the root of the domain where the filter is defined. |
| *scope* | MUST be the base object (0). |
| *derefAliases* | MUST be set to 0 (neverDerefAliases). |
| *sizeLimit* | No limit is set (this MUST be set to 0). |
| *timeLimit* | MUST be set to 0 (infinite). |
| *typesOnly* | MUST be set to 0 (FALSE). |
| *filter* | The following LDAP filter (using the representation as specified in [RFC2254]) MUST be used:<br>(objectclass=*) |
| *attributes* | The following attribute names are passed as inputs to the LDAP search request: **msWMI-ID**, **msWMI-Name**, **msWMI-Parm1**, **msWMI-Author**, **msWMI-ChangeDate**, **msWMI-CreationDate**, and **msWMI-Parm2**. |

The Client receives a reply from the search request from the GP server with one or more LDAP searchResponse messages. Those messages MUST contain one or more **searchResultEntries**. Those **searchResultEntries** MUST contain an **objectName** DN attribute that is the WMI filter named by that DN. The **searchResultEntry** also MUST contain an **attributes** field with the values in Active Directory for the attributes of the WMI filter object that were searched for. The attributes MUST have the following formats.

| Attribute | Format |
|---|---|
| **msWMI-ID** | GUID. |
| **msWMI-Name** | Directory string that gives a human-friendly name that an administrator defines. |

| Attribute | Format |
|---|---|
| **msWMI-Parm1** | Directory string that gives a human-friendly description of the filter's purpose that an administrator defines. |
| **msWMI-Author** | Directory string that gives the name of the author of the WMI filter. |
| **msWMI-ChangeDate** | Date-Time field indicating when the filter was last updated. |
| **msWMI-CreationDate** | Date-Time field indicating when the filter was created. |
| **msWMI-Parm2** | Directory string that contains the **WMI Query Language (WQL)** query for a WQL query to be executed on the Client computer. |

### 2.2.6 Link Speed Determination

The Client MUST estimate the link speed of the network between the Client and the Domain Controller.<10>The Link Speed Determination message MAY use Internet Control Message Protocol (ICMP) (as specified in [RFC792]) as a transport, supporting at least 2048-byte packets, as an implementation-specific means.

### 2.2.7 GPO Read Administration

This operation is similar to the sequences for policy application, but it is targeted only at a single GPO. This part of the protocol allows users to view the settings and state of an individual GPO.

Attributes and files MUST be interpreted in the same way as interpreted in section 2.2.4 with the only difference being the search protocol sequence in the LDAP search request. This difference is specified in the following table.

| Parameter | Value |
|---|---|
| *baseObject* | Base Search Scope MUST be the GPO DN for some GPO. |
| *Scope* | Search only the root of the computer's domain (this MUST be set to 0). |
| *derefAliases* | MUST be set to 0 (neverDerefAliases). |
| *sizeLimit* | No limit is set (this MUST be set to 0). |
| *timeLimit* | MUST be set to 0 (infinite). |
| *typesOnly* | MUST be set to 0 (FALSE). |
| *Filter* | The following LDAP filter (as specified in [RFC2254]) MUST be used:<br>(objectClass=*) |
| *Attributes* | MAY be NULL, but SHOULD be as specified in section 2.2.4, plus systemFlags, whenCreated, and whenChanged.<11> |

The reply from the search request from the Group Policy (GP) server MUST include the attributes in section 2.2.4 as well as the following additional attributes. Any attributes other than those specified here and in section 2.2.4 MUST be ignored.

| Attribute | Format |
|---|---|
| systemFlags | An integer value that contains flags that define additional properties of this GPO. This value is maintained by the Active Directory server. For more information, see [MS-ADA3] and [MS-ADTS]. |
| whenCreated | The date when this GPO was created. This value is set by the Active Directory server. For more information, see [MS-ADA3]. |
| whenChanged | The date when this GPO was last changed. This value is managed by the Active Directory server. For more information, see [MS-ADA3]. |

### 2.2.8 GPO Write Administration

Administrative tools use the following messages to create or update a GPO.

#### 2.2.8.1 GPO Creation Message

An administrative tool MUST generate the GUID portion of the new GPO DN by using the GUID-generation algorithm, as specified in [C706] Appendix A "Universal Unique Identifier", to ensure that the DN is unique in the domain.

Containers and GPO existence MUST be checked by sending the following LDAP SearchRequest messages to the GP server prior to the applicable LDAP addRequest:

1. LDAP SearchRequest to search for GPO container MUST be sent to the GP server with the following parameters:

| Parameter | Value |
|---|---|
| baseObject | CN=<GPO DN> |
| scope | MUST be the base object (0). |
| filter | The following LDAP filter (using the representation as specified in [RFC2254]) MUST be used:<br>(objectclass=*) |
| attributes | **objectClass** |

2. LDAP SearchRequest to search for user container MUST be sent to the GP server with the following parameters:

| Parameter | Value |
|---|---|
| baseObject | CN=User,CN=<GPO DN> |
| scope | MUST be the base object (0). |
| filter | The following LDAP filter (using the representation as specified in [RFC2254]) MUST be used:<br>(objectclass=*) |
| attributes | **objectClass** |

3. LDAP SearchRequest to search for machine container MUST be sent to the GP server with the following parameters:

| Parameter | Value |
|---|---|
| baseObject | CN=Machine,CN=<GPO DN> |
| scope | MUST be the base object (0). |
| filter | The following LDAP filter (using the representation as specified in [RFC2254]) MUST be used:<br>(objectclass=*) |
| attributes | **objectClass** |

The creation of Policies container MUST be accomplished through an LDAP addRequest message with the following parameters:

| Parameter | Value |
|---|---|
| *entry* | CN=policies,CN=system,<DN of domain naming context> |
| *attributes* | MUST contain two attributes: **objectClass** and **cn**. |

The LDAP addRequest message attribute parameter has the following format:

| Attribute name | Value | Meaning |
|---|---|---|
| objectClass | MUST be the directory string value "container". | Name of the Active Directory object class type to create through this message. |
| cn | MUST be the directory string value "Policies". | Name of the Active Directory container. |

The creation of the Active Directory portion of the new GPO MUST be accomplished through an LDAP addRequest message with the following parameters:

| Parameter | Value |
|---|---|
| *entry* | A GPO DN that is unique for the GPO in the domain. An administrative tool MUST generate the GUID portion of the GPO DN by using the GUID-generation algorithm, as specified in [C706], to ensure that the DN is unique in the domain. |
| *attributes* | MUST contain two attributes: **objectClass** and **cn**. |

The LDAP addRequest message attribute parameter has the following format:

| Attribute name | Value | Meaning |
|---|---|---|
| **objectClass** | MUST be the directory string value "groupPolicyContainer". | Name of the Active Directory object class type to create through this message. |
| **cn** | MUST be the directory string value | Name of the Active Directory GPO container. |

| Attribute name | Value | Meaning |
|---|---|---|
| | "{GPO GUID}". | |

Similar addRequest messages MUST be made to create subcontainers of the groupPolicyContainer object. The addRequest messages MUST have the following parameters and attributes.

A user subcontainer has the following parameters:

| Parameter | Value |
|---|---|
| *entry* | MUST be the directory string value "cn=user,<GPO DN>". |
| *attributes* | MUST contain two attributes: **objectClass** and **cn**. |

A user subcontainer attribute parameter has the following format:

| Attribute name | Value | Meaning |
|---|---|---|
| **objectClass** | MUST be the directory string value "container". | Name of the Active Directory object class type to create through this message. |
| **cn** | MUST be the directory string value "user". | Name of the Active Directory GPO subcontainer. |

A machine subcontainer has the following parameters:

| Parameter | Value |
|---|---|
| *entry* | MUST be the directory string value "cn=machine,<GPO DN>". |
| *attributes* | MUST contain two attributes: **objectClass** and **cn**. |

A machine subcontainer attribute parameter has the following format:

| Attribute name | Value | Meaning |
|---|---|---|
| **objectClass** | MUST be the directory string value "container". | Name of the Active Directory object class type to create through this message. |
| **cn** | MUST be the directory string value "machine". | Name of the Active Directory GPO subcontainer. |

An LDAP SearchRequest MUST be sent to the GP server with the following parameters:

| Parameter | Value |
|---|---|
| baseObject | CN=<GPO DN> |
| scope | MUST be the base object (0). |
| derefAliases | MUST be set to 0 (neverDerefAliases). |

| Parameter | Value |
|---|---|
| sizeLimit | No limit is set (this MUST be set to 0). |
| timeLimit | MUST be set to 0 (infinite). |
| typesOnly | MUST be set to 0 (FALSE). |
| filter | The following LDAP filter (using the representation as specified in [RFC2254]) MUST be used:<br>(objectclass=*) |
| attributes | **nTSecurityDescriptor**: A security descriptor whose format is specified in [MS-DTYP] section 2.4.6. |

### 2.2.8.2 GPO Extension Update Message

The GPO Extension Update message MUST be an LDAP modifyRequest with the following parameters. The result of modifyRequest is a modifyResponse message in reply, as defined in [RFC2251] section 4.6. The **resultCode** field value determines a failure or success for the message.

| Parameter | Value |
|---|---|
| *Entry* | GPO DN for the GPO being updated. |
| *attributes* | This field MUST specify the attributes **versionNumber** and either **gPCUserExtensionNames** (if user policy settings are being modified) or **gPCMachineExtensionNames** (if computer policy settings are being modified). The operation for each attribute specified MUST be "replace" as specified in [RFC2251].<br>The syntax of these attributes is specified in section 2.2.4. If the GUID is not already present from a prior update, **gPCUserExtensionNames** or **gPCMachineExtensionNames** MUST be updated with the extension GUID of the plug-in that modified the GPO. |

### 2.2.8.3 GPO Property Update Message

The GPO Property Update message MUST be an LDAP modifyRequest with the following parameters. The result of modifyRequest is a modifyResponse message in reply, as defined in [RFC2251] section 4.6. The **resultCode** field value determines a failure or success for the message.

| Parameter | Value |
|---|---|
| *Entry* | GPO DN for the GPO being updated. |
| *attributes* | MUST specify one or more of the attributes defined in section 2.2.4.<br>Semantics of these attributes are defined in section 2.2.4. The operation for each attribute specified MUST be "replace" as specified in [RFC2251]. |

### 2.2.8.4 SOM Property Update Message

The SOM Property Update message MUST be an LDAP modifyRequest with the following parameters. The result of modifyRequest is a modifyResponse message in reply, as defined in [RFC2251] section 4.6. The **resultCode** field value determines a failure or success for the message.

| Parameter | Value |
|---|---|
| *Entry* | SOM DN for the SOM being updated. |
| *attributes* | This field MUST specify one or more of the attributes:<br>▪ **gpLink**: A Directory String encoded in UTF-8 as defined in [RFC2252] section 6.10 specifying a list of GPOs that are associated with the SOM and the properties of the association. The format of this string is defined in section 2.2.2.<br>▪ **gpOptions**: An LDAP INTEGER specifying properties of the SOM.<br>The syntax of these attributes is defined in section 2.2.2. The operation for each attribute specified MUST be "replace" as specified in [RFC2251]. |

#### 2.2.8.5 GPO Deletion Message

The deletion of the Active Directory portion of the GPO MUST be accomplished through a series of LDAP delRequest messages with the following parameters.

| Parameter | Value |
|---|---|
| *entry* | GPO DN or GPO subcontainer DN |

## 2.3 Directory Service Schema Elements

The Group Policy: Core Protocol accesses the following Directory Service schema classes and attributes listed in the following table. For the syntactic specifications of the following <Class> or <Class> <Attribute> pairs, refer to: [MS-ADSC], [MS-ADA1], [MS-ADA2], and [MS-ADA3].

| Class | Attribute |
|---|---|
| domain | gPLink<br>gPOptions |
| groupPolicyContainer | displayName<br>flags<br>gPCFileSysPath<br>gPCFunctionalityVersion<br>gPCMachineExtensionNames<br>gPCUserExtensionNames<br>gPCWQLFilter<br>versionNumber |
| msWMI-Som | msWMI-Author<br>msWMI-ChangeDate<br>msWMI-CreationDate<br>msWMI-ID<br>msWMI-Name<br>msWMI-Parm1<br>msWMI-Parm2 |

| Class | Attribute |
| --- | --- |
| organizationalUnit | gPLink<br>gPOptions |
| site | gPLink<br>gPOptions |

# 3 Protocol Details

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

The following sections describe the state maintained on the Client and server that implement the Group Policy: Core Protocol.

## 3.1 Server Details

### 3.1.1 Server Abstract Data Model

The GP server has no knowledge of the Group Policy: Core Protocol. It is merely an LDAP and file server that stores generic objects. The GP server primarily stores information on managed objects and policies that must affect those objects.

The GP server keeps state in two conceptual stores: an LDAP server and a domain-based distributed file system. The LDAP server stores information on policy targets and the policies that affect those targets. The distributed file system part of the GP server is primarily intended for storing large streams of data (by convention, large data is considered to be anything over 100 kilobytes) that are not appropriate for a lightweight store (such as an LDAP server) or for storing data that has traditionally been accessed in files by Clients outside the context of the Group Policy: Core Protocol.

The LDAP server portion models policy in the following ways.

- Policy targets exist in the directory as individual user accounts and computer accounts.
- Thus policy targets must exist in organizational unit containers and domains.
- Sites must also be defined in the LDAP server.
- Each site, domain, and organizational unit has an attribute named **gpLink** that associates that site, domain, or organizational unit with a set of gpContainer objects that logically represent GPOs in the LDAP server.

Logically, GPOs exist in two different sections.

- User section: Contains all information that relates to user policies, which Clients are to retrieve as part of user policy mode. Group Policy Extensions store all server state for user policy settings within this section in formats of their own specifications.
- Computer section: Contains all information that relate to computer policies, which Clients are to retrieve as part of computer policy mode. Group Policy Extensions store all server state for computer policy settings within this section in formats of their own specifications.

Each of these sections corresponds to a policy mode.

- User Extension List: The list of Group Policy Extensions that stores settings in the User section of the GPO.
- Computer Extension List: The list of Group Policy Extensions that stores settings in the Computer section of the GPO.

GPOs themselves have the following structures on the GP server.

- GPO Active Directory storage: For each GPO to be communicated through the protocol, the following objects and attributes MUST be accessible under the LDAP path CN=Policies,CN=System,<DN for root of the domain> via LDAP. The "Policies" object is of the class "Container" (as defined in [MS-ADSC]); whereas the GPO object is of the class "groupPolicyContainer" (as defined in [MS-ADSC]). The CN attribute of the object MUST be a GUID that is unique in the domain.
- GPO user container: The container CN=User,<GPO DN> that stores all Active Directory information to be retrieved for Group Policy Extension sequences for user policy mode.
- GPO computer container: The container CN=Machine,<GPO DN> that stores all Active Directory information to be retrieved for Group Policy Extension sequences for computer policy mode.
- GPO domain-based distributed file system storage: The following file system information MUST be available on the GP server through file access as follows:
  - GPO path: A GPO path MUST be available for the GPO. For a given GPO, the GUID in the GPO DN and the GPO path MUST be the same.
  - GPO user path: The subdirectory <GPOPath>\User (where <GPO Path> is the GPO path) MUST exist; this subdirectory contains all user policy information that is stored in the file system.
  - GPO computer path: The subdirectory <GPOPath>\Machine (where <GPO Path> is the GPO path) MUST exist; this subdirectory contains all computer policy information that MUST be stored in the file system.

### 3.1.2 Timers

None.

### 3.1.3 Initialization

None.

### 3.1.4 Higher-Layer Triggered Events

None.

### 3.1.5 Message Processing Events and Sequencing Rules

None.

### 3.1.6 Timer Events

None.

### 3.1.7 Other Local Events

None.

## 3.2 Client Details

### 3.2.1 Client Abstract Data Model

The following sections describe the data that is stored on the Client for applying policy for a policy target.

#### 3.2.1.1 Cache of GPO Versions

A table of the following information MAY be present on the Client, indexed by GPO GUID, as part of the efficient implementation of the protocol:

GPO Container Version: GPO version that is stored in the Active Directory portion of the GPO.

**GPO File System Version**: GPO version that is stored in the SYSVOL file system portion of the GPO.

The Client SHOULD maintain this information (even across system restarts) after every Group Policy application protocol conversation so that Clients do not overwhelm the GP server. As part of any future policy application protocol sequence, these cached attributes can be compared against those attributes that are being retrieved for each GPO returned by the server. If both the version numbers are identical, this can be interpreted as meaning that there are no changes to administrative intent since the last policy application; thus, there is nothing new for the Client to enforce because it may be assumed to be in compliance from the earlier policy application.

Thus, as an optimization, the Client MAY choose not to include some of the Group Policy Extension sequences for some extensions to Group Policy to avoid unnecessary network usage and Client or server processing.<12>

#### 3.2.1.2 Default Policy Source Mode

The Default Policy Source Mode is used by the client to compute the Policy Source Mode. If specified, it MUST be one of the values specified in section 3.2.1.3.<13>

#### 3.2.1.3 Policy Source Mode

Policy Source Mode determines the **policy sources** used by the Client to compute the Filtered GPO list (Public) (section 3.2.1.5), computed on the Client by searching on the policy source's hierarchy in Active Directory to identify the associated set of GPOs.

Policy Source Mode is assigned one of the following values on the Client. The Client MUST choose policy sources as described for each Policy Source Mode:

- Normal mode: The Filtered GPO list (Public) MUST be computed using the policy source of the policy target account.
- Loopback replace mode: The Filtered GPO list (Public) MUST be computed using the policy source of the computer account and applied to the impersonated user.
- Loopback merge mode: The Filtered GPO list (Public) MUST be computed using two policy sources: the one for the computer account and the one for the policy target account. The GPO list obtained for the computer is appended to the GPO list for the user, and the merged list is applied to the impersonated user. The GPO list for the computer is applied later and therefore has precedence if it conflicts with settings in the user's list.

The Policy Source Mode computation is as specified in section 3.2.5.1.

#### 3.2.1.4 GPO List

A GPO list is a list of Group Policy objects that are associated with a specified policy target. The list is ordered by **GPO precedence** in descending order of priority. The following information MUST be maintained for each GPO:

1. **GPO versions**: A 32-bit integer that stores the GPO container version in the lower 16 bits and the GPO file system version in the upper 16 bits. For the user policy application mode of the policy application, the user GPO version part of the GPO container version and the GPO file system version MUST be maintained. For the computer policy application mode of the policy application, the machine GPO version part of the GPO container version and the GPO file system version MUST be maintained.

2. **Scoped GPO DN**: A Unicode string of the scoped GPO DN, prefixed with "LDAP://". For the user policy application mode of the policy application, the **user-scoped GPO DN** MUST be maintained. For the computer policy application mode of the policy application, the **computer-scoped GPO DN** MUST be maintained.

3. **Scoped GPO path**: A Unicode string of the scoped GPO path. For the user policy application mode of the policy application, the user-scoped GPO path MUST be maintained. For the computer policy application mode of the policy application, the **computer-scoped GPO path** MUST be maintained.

4. **GPO GUID**: The curly braced GUID string that identifies the GPO.

5. Display name: A human-readable directory string description of the GPO.

6. ExtensionList: An array of CSE GUIDs configured in the GPO. The ExtensionList is an array of CSE GUIDs from gPCMachineExtensionNames for the computer policy application mode and an array of CSE GUIDs from gPCUserExtensionNames for the user policy application mode as specified in section 2.2.4.

7. FunctionalityVersion: An integer that stores the functionality version of the GPO.

8. SecurityDescriptor: The security descriptor, as specified in [MS-DTYP] section 2.4.6, of the GPO.

9. WMI Filter: A Unicode string that stores the WMI filter that is associated with the GPO.

#### 3.2.1.5 Filtered GPO List (Public)

The Filtered GPO list (Public) is computed separately for each Group Policy Extension at policy application time (as specified in section 3.2.5.1.10). The Filtered GPO list (Public) contains only those GPOs that pass all the criteria specified in section 3.2.5.1 and is then shared with that specific Group Policy Extension. GPOs represented in this list have passed access checking and are a subset of those in abstract element **GPO List**.

#### 3.2.1.6 SOM List

A prioritized list of SOMs to which the specified policy target belongs. An SOM MUST be prioritized higher in the list compared to its parent SOMs. The following information MUST be maintained for each SOM:

- **SOM DN**: The DN of the SOM.

- **gpLink**: A directory string value of the **gpLink** attribute on the SOM.
- **gpOptions**: An integer value of the **gpOptions** attribute on the SOM.
- **SOM Object type**: One of the following values is assigned:
  - **GPLinkOrganizationalUnit**: SOM represents an organizational unit.
  - **GPLinkDomain**: SOM represents a domain.

#### 3.2.1.7 SOM GPLink List

A prioritized list of GPO DNs that is associated with a given SOM. The following information MUST be maintained for each object in the list:

- GPO DN: Distinguished name of GPO.
- Enforced: A Boolean to indicate whether the GPO DN is enforced.

#### 3.2.1.8 Enforced GPLink List

A prioritized list of enforced GPO DNs.

#### 3.2.1.9 Non-enforced GPLink List

A prioritized list of non-enforced GPO DNs.

#### 3.2.1.10 GPLink List

A prioritized list of GPO DNs.

#### 3.2.1.11 Allow-Enforced-GPOs-Only

For each SOM a Boolean value to indicates whether only enforced GPOs should be allowed.

#### 3.2.1.12 Policy Application Mode

Determines whether the policy application is for the logged-on User or the Computer.

#### 3.2.1.13 Group Policy Server

This is the fully qualified domain name (FQDN) of the domain controller prefixed by 2 backslashes (\\).

#### 3.2.1.14 Configured Computer Base Frequency

If configured, this value along with the Configured Computer Random Offset determines the frequency of policy application for the computer. The minimum value is 7 seconds and the maximum value is 45 days.<14>

#### 3.2.1.15 Configured Computer Random Offset

If specified, this value along with Configured Computer Base Frequency determines the frequency of policy application for the computer. The minimum value is 0 minutes and the maximum value is 1440 minutes.<15>

#### 3.2.1.16 Policy Target Domain Name

This is the fully qualified domain name (FQDN) of policy target.

#### 3.2.1.17 Computer Policy Refresh Interval

This is the frequency of policy application for the computer.

If specified, Configured Computer Base Frequency and Configured Computer Random Offset are added to determine this value.

If **Configured Computer Base Frequency** and **Configured Computer Random Offset** are not specified, this value is 5 minutes for clients that are domain controllers. For clients that are not domain controllers, this value is determined by adding 90 minutes to an offset value in the range of 0 to 30 minutes.

#### 3.2.1.18 Configured User Base Frequency

If specified, this value along with Configured User Random Offset determines the frequency of policy application for an interactively logged-on user. The minimum value is 7 seconds and maximum value is 45 days.<16>

#### 3.2.1.19 Configured User Random Offset

If specified, this value along with Configured User Base Frequency determines the frequency of policy application for an interactively logged-on user. The minimum value is 0 minutes and the maximum value is 1440 minutes.<17>

#### 3.2.1.20 User Policy Refresh Interval

This is the frequency of policy application for interactively logged-on users.

If specified, Configured User Base Frequency and Configured User Random Offset are added to determine this value.

If **Configured User Base Frequency** and **Configured User Random Offset** are not specified, this value is determined by adding 90 minutes to an offset value in the range of 0 to 30 minutes.

#### 3.2.1.21 Configured Disable Periodic Refresh

If specified, a Boolean value of TRUE indicates that periodic refresh is disabled for the computer and all interactively logged-on users, and FALSE indicates that periodic refresh is enabled for the computer and all interactively logged-on users.<18>

#### 3.2.1.22 Disable Periodic Refresh

A Boolean value of TRUE indicates that periodic refresh is disabled for the computer and all interactively logged-on users and FALSE indicates that periodic refresh is enabled for the computer and all interactively logged-on users.

If Configured Disable Periodic Refresh is specified, this value is same as the value of **Configured Disable Periodic Refresh**.

If **Configured Disable Periodic Refresh** is not specified, this value MUST be FALSE.

#### 3.2.1.23 GP Client AD Connection Handle

An ADConnection handle as defined in [MS-ADSO] section 6.2.3. This element is used each time a GP client communicates with a Group Policy Server over an Active Directory connection.

#### 3.2.1.24 Policy Application Event

A local event which indicates that policy application completed successfully.

### 3.2.2 Timers

Unless periodic refresh is disabled by Disable Periodic Refresh, the Client SHOULD have the following timers:

Computer Periodic Refresh timer: This timer SHOULD be triggered periodically to check for updated policy for the computer. The frequency of this timer is determined by the Computer Policy Refresh Interval.

User Periodic Refresh timer: This timer SHOULD be triggered periodically to check for updated policy for each user interactively logged on to the computer. This timer is maintained separately for each interactively logged on user. The frequency of this timer is determined by the User Policy Refresh Interval.

### 3.2.3 Initialization

None.

### 3.2.4 Higher-Layer Triggered Events

Each of the Group Policy Extensions MUST request and retrieve its settings during the policy application sequence. The request and retrieval are specific to each Group Policy Extension and are not specified in this document.

**Note** All of the Group Policy Extension messages can be considered to have the following logical input parameters. (An individual Group Policy Extension sequence MAY use every part of the input parameters to obtain its settings.) Refer to the specific Group Policy Extension sequence for the format of the data that is actually transmitted between the Client and any servers during the protocol sequence. A GPO state of New, Changed, or Deleted MAY be derived by comparing a complete **GPO DN list** from Active Directory against **Group Policy** processing results logged on the local machine during each policy application session. If the client-specific implementation does not support Group Policy processing results logging then all GPOs MUST be considered New or Changed in order to apply policy.

The logical parameters are:

- **New or Changed GPO list:** The **New or Changed GPO list** contains one entry for each GPO for which a Group Policy Extension will request and retrieve settings as well as the GPO path.
- **Deleted GPO list**: The **Deleted GPO list** contains one entry for each GPO for which a Group Policy Extension will request and retrieve settings as well as the GPO path.
- **A set of flags defining aspects of this policy application session:** these flag values are listed in the following table:.

| Value | Description |
|---|---|
| 0x00000001 | Computer Policy Application Mode. |
| 0x00000010 | Policy applying as a background process. |
| 0x00000020 | Policy applying across a slow link. |
| 0x00000040 | The Group Policy Extension SHOULD use verbose logging. |
| 0x00000080 | No changes were detected in the GPO List. |
| 0x00000100 | A change in link speed was detected in comparison to the previous policy application. |
| 0x00000200 | A change in logging was detected in comparison to the previous policy application. |
| 0x00000400 | A forced refresh of policy is being applied. |
| 0x00000800 | The computer is in maintenance or recovery(Safe) mode. |
| 0x00001000 | Policy applying as a foreground process. |

- **A security token enabling impersonation of the user for user policy application mode.**

The GPO DN list (New or Changed GPOs) passed to each Group Policy Extension's specific protocol sequence only contains those GPOs that are marked as containing those Group Policy Extensions (section 3.2.5.1.10). The GPO list does not contain GPOs that are noted by the Client as denied (section 3.2.5.1.6), or GPOs for which the WMI query returns no results and are considered denied (section 3.2.5.1.7). The GPO DN list (Deleted GPOs) passed to each Group Policy Extension specific protocol sequence contains only those GPOs that no longer apply but applied during the previous policy application session.

### 3.2.5 Message Processing Events and Sequencing Rules

#### 3.2.5.1 Policy Application

Policy application is comprised of the following steps:

- DC Discovery and AD Connection establishment
- DN Discovery
- Domain SOM Search
- Site Search
- GPO Search
- GPO Filter Evaluation
- **WMI** Filter Evaluation
- AD Connection termination
- Link Speed Discovery
- Extension Protocol Sequences

- Policy Application Notification

The following initialization steps MUST be completed before proceeding with the tasks listed above.

The GPO list, SOM list, GPLink list, SOM GPLink list, Enforced GPLink list, and Non-enforced GPLink list MUST be initialized to empty lists.

Allow-Enforced-GPOs-Only MUST be initialized to FALSE.

For Computer Policy Application Mode, the Policy Source Mode MUST be set to Normal.

For User Policy Application Mode, the Client enumerates all the domains in the same **forest** as the computer's domain by performing a local call consistent with the behavior as specified in the **DsrEnumerateDomainTrusts** method (as defined in [MS-NRPC] section 3.5.5.5.1) with the following parameters.

- NULL for *ServerName*.
- Value A for *Flags*.

If the method returns a non-zero error code, policy application MUST be terminated and an event SHOULD be logged using an implementation-specific mechanism.<19> Otherwise, if the Policy Target Domain Name (section 3.2.1.16) is not in the list of **DNS** domains found, then the Policy Source Mode MUST be set to Loopback replace mode. If the **Policy Target Domain Name** is in the list, the Policy Source Mode MUST be initialized to the Default Policy Source Mode.

The priority list of GPOs applicable to a policy target MUST be computed as specified in the following subsections (3.2.5.1.x).

1. If the Policy Source Mode is normal mode, the policy target and policy target domain MUST be used to compute the GPO list.
2. If the Policy Source Mode is loopback replace mode, the computer account name and computer domain MUST be used to compute the GPO list.
3. If the Policy Source Mode is loopback merge mode, step 1 MUST be run, followed by step 2.

#### 3.2.5.1.1 DC Discovery and AD Connection Establishment

The Client performs the following steps to discover and establish Active Directory connection with the DC. This series of steps is performed a second time if steps 2-6 fail the first time.

1. The Client locates a domain controller by invoking the **DsrGetDcNameEx2** method (as specified in [MS-NRPC] section 3.5.5.3.1) locally with the following parameters:
   - NULL for *ComputerName*.
   - NULL for *AccountName*.
   - 0 for *AllowableAccountControlBits*.
   - Policy Target Domain Name (section 3.2.1.16) for *DomainName*.
   - NULL for *DomainGuid*.
   - NULL for *SiteName*.

- Values B and R for *Flags* on the first iteration. Additionally, value A is also passed on the second iteration.

If the method returns a nonzero error code, policy application MUST be terminated. Otherwise, the **Group Policy Server** ADM element (specified in section 3.2.1.13) is populated with the value of the **DomainControllerName** field in the returned **DOMAIN_CONTROLLER_INFOW** structure.

2. The Client invokes the task "Initialize an ADConnection", as defined in [MS-ADSO] section 6.2.6.1.1, with the following parameters:

   - *TaskInputTargetName*: Value of **Group Policy Server** ADM element.
   - *TaskInputPortNumber*: 389
   - Store the new *TaskReturnADConnection* returned from the task as the GP Client AD Connection Handle ADM element.

   If the task returns failure and it is the first iteration, repeat from step 1. Otherwise, policy application MUST be terminated and an event SHOULD be logged using an implementation-specific mechanism.<20>

3. The Client invokes the task "Setting an LDAP Option on an ADConnection", as defined in [MS-ADSO] section 6.2.6.1.2, with the following parameters:

   - *TaskInputADConnection*: Value of the **GP Client AD Connection Handle** ADM element
   - *TaskInputOptionName*: LDAP_OPT_SIGN
   - *TaskInputOptionValue*: TRUE

   If the task returns failure and it is the first iteration, repeat from step 1. Otherwise, policy application MUST be terminated and an event SHOULD be logged using an implementation-specific mechanism.<21>

4. If Policy Application Mode is Computer, the Client invokes the task "Setting an LDAP Option on an ADConnection", as defined in [MS-ADSO] section 6.2.6.1.2, with the following parameters:

   - *TaskInputADConnection*: Value of the **GP Client AD Connection Handle** ADM element
   - *TaskInputOptionName*: LDAP_OPT_DNSDOMAIN_NAME
   - *TaskInputOptionValue*: Value of the Policy Target Domain Name ADM element

   If the task returns failure and it is the first iteration, repeat from step 1. Otherwise, policy application MUST be terminated and an event SHOULD be logged using an implementation-specific mechanism.<22>

5. The Client invokes the task "Establishing an ADConnection", as defined in [MS-ADSO] section 6.2.6.1.3, with the following parameter:

   - *TaskInputADConnection*: Value of the **GP Client AD Connection Handle** ADM element

   If the task returns FALSE, policy application MUST be terminated and an event SHOULD be logged using an implementation-specific mechanism.<23>

6. The Client invokes the task "Setting an LDAP Option on an ADConnection", as defined in [MS-ADSO] section 6.2.6.1.2, with the following parameters:

- *TaskInputADConnection*: Value of the **GP Client AD Connection Handle** ADM element
- *TaskInputOptionName*: LDAP_OPT_AUTH_INFO
- *TaskInputOptionValue*: For computer policy mode, SASL authentication with Kerberos credentials in Unicode form. For user policy mode, SASL using the GSS-SPNEGO mechanism with null credentials.

7. After the Active Directory connection is initialized and the options are set, the Client invokes the "Performing an LDAP Bind on an ADConnection" task, as specified in [MS-ADSO] section 6.2.6.1.4, with the following parameter:

   - *TaskInputADConnection*: Value of the **GP Client AD Connection Handle** ADM element

   If the *TaskReturnStatus* returned is not 0 and it is the first iteration, repeat from step 1. Otherwise, policy application MUST be terminated and an event SHOULD be logged using an implementation-specific mechanism.<24>

#### 3.2.5.1.2 DN Discovery

The Client attempts to discover the policy target DN that should be used to query for the GPOs, as specified in section 2.2.1. If the computer's account is to be used, the computer account name MUST be specified in DS_NT4_ACCOUNT_NAME format. If the user account is to be used, the user account name MUST be specified in DS_NT4_ACCOUNT_NAME format. If this message is invalid, as specified in section 2.2, policy application MUST be terminated and an event SHOULD be logged using an implementation-specific mechanism.<25>

#### 3.2.5.1.3 Domain SOM Search

This step uses the domain controller name and the DN that was retrieved in section 3.2.5.1.2 for the Domain Scope of Management search. The policy target DN retrieved MUST be parsed to form the prioritized list of SOMs. The prioritized SOM list MUST store the SOM Object type (**GPLinkOrganizationalUnit** or **GPLinkDomain**) and the DN, and is populated as follows:

1. The policy target DN MUST be parsed to compute the parent DN.
2. The parent DN that is computed MUST be appended to the end of the SOM list.
3. If there is a parent DN, and if it does not start with "DC=", steps 1 and 2 MUST be repeated with the parent DN computed until the DN starts with "DC=".
4. All of the SOMs in the SOM list that don't start with "OU=" or "DC=" MUST NOT be added to the SOM list.

All of the SOMs in the domain that are discovered MUST be searched to retrieve the **gpLink** and **gpOptions** attributes as follows:

1. Disable LDAP_OPT_REFERRALS by passing abstract element "GP Client AD Connection Handle", setting an LDAP Option on an ADConnection.
2. An LDAP SearchRequest as specified in section 2.2.2 MUST be sent from the Client to the GP server, and the SearchResponse received MUST be verified to satisfy the specified requirements. The SearchResponse contains the **gpLink** and **gpOptions** attribute values for all of the SOMs.

If there are no SOMs to search for, the protocol sequence continues at section 3.2.5.1.4 Site Search. If Domain SOM Search fails, the entire protocol sequence MUST be terminated and an event SHOULD be logged using an implementation-specific mechanism.<26>

#### 3.2.5.1.4 Site Search

The site to which the Client computer belongs (the SiteName) is determined by invoking the DsrGetSiteName method (as specified in [MS-NRPC] section 3.5.5.3.6) locally with the following parameters:

- NULL for *ComputerName*.

If the method returns ERROR_NO_SITENAME, the remainder of this message MUST be skipped and the protocol sequence MUST continue at GPO Search (section 3.2.5.1.5) . The initial site named "Default-First-Site-Name" is documented in [MS-ADTS] section 7.1.1.2.2.1 that specifies the Site object. If the method returns any other nonzero error code, policy application MUST be terminated. If the method returns zero, then the DN of the configuration container of the domain MUST be searched for as follows:

1. An LDAP SearchRequest as specified in section 2.2.3 MUST be sent from the Client to the GP server, and the SearchResponse received MUST be verified to satisfy the specified requirements. The SearchResponse contains the configurationNamingContext attribute value. From this value and the SiteName value (the out parameter of the previous DsrGetSiteName method call), the site distinguished name (DN) MUST be computed by concatenating the strings "CN=", <the site name>, ",CN=Sites,", and <the DN of the configuration container>. This site DN MUST be used for the remainder of this message to retrieve the attributes of the site object.

2. Another LDAP SearchRequest, as specified in section 2.2.3, MUST be sent from the Client to the GP server to retrieve the **gpLink** and **gpOptions** attribute values.

If this message is invalid in any way, as specified in section 2.2.3, the entire Group Policy: Core Protocol policy application sequence MUST be terminated and an event SHOULD be logged using an implementation-specific mechanism.<27>

The retrieved **gpLink** attribute of the site contains LDAP DNs for GPOs that are associated with this site. The **gpLink**, **gpOptions**, site DN, and Object type (GPLinkSite) MUST be appended to the end of the SOM list.

#### 3.2.5.1.5 GPO Search

This message requires the success of all previous messages that have retrieved a scope of management and a **gpLink** that are associated with each of the SOMs, and have stored them in the SOM list. If this message is invalid, policy application MUST be terminated and an event SHOULD be logged using an implementation-specific mechanism.<28>

The following steps MUST be used to create a prioritized list of GPOs:

1. Set Allow-Enforced-GPOs-Only to FALSE.

2. For each SOM in the SOM list, beginning with the first SOM:

    1. Retrieve **gpLink** and **gpOptions** attributes of current SOM: searchRequest defined by baseObject: DN of SOM, scope: zero, filter: objectClass=*, attributes: **gpLink** and **gpOptions**.

    2. The **Client** MUST parse the **gpLink** value into a list of individual directory strings of the following format, as specified in section 2.2.2.

```
[<GPO DN>;<GPLinkOptions>]
```

For each directory string in the list, if the decimal representation of the **GPLinkOptions** bit field does not specify that the GPO DN MUST be ignored, an element MUST be appended to the end of SOM GPLink list as follows:

1. GPO DN field MUST be set to the GPO DN in the directory string.

2. The **Enforced** field MUST be set to TRUE, if the decimal representation of the **GPLinkOptions** bit field specifies that the GPO DN is an enforced GPO; otherwise, it MUST be set to FALSE.

3. For each element in the SOM GPLink list, beginning with the first element:

    1. If the **Enforced** field is FALSE, and Allow-Enforced-GPOs-Only is set to FALSE, the GPO DN MUST be prepended to the beginning of the Non-enforced GPLink list.

    2. The element MUST be removed from the SOM GPLink list.

4. For each element that remains in the SOM GPLink list, beginning with the first element:

    1. GPO DN MUST be appended to the end of the Enforced GPLink list.

    2. The element MUST be removed from the SOM GPLink list.

5. If the **gpOptions** value for the SOM is set to directory string "1", as specified in section 2.2.2, Allow-Enforced-GPOs-Only MUST be set to TRUE.

3. For each GPO DN in the Non-enforced GPLink list, beginning with the first element, GPO DN MUST be appended to the end of the GPLink list.

4. For each GPO DN in the Enforced GPLink list, beginning with the first element, GPO DN MUST be appended to the end of the GPLink list.

5. The list of GPO DNs MUST be grouped on the basis of domain. In each domain, all of the GPO DNs in that domain MUST be placed in the LDAP Filter portion of the protocol. For each group, attributes of GPOs MUST be queried as follows:

    1. An LDAP SearchRequest as specified in section 2.2.4 MUST be verified to satisfy the specified requirements.

    2. Many response buffers could return since the **Filter** field is a logical OR of all GPO DNs in the domain. When the searchResponse returns, every LDAPMessage response buffer, along with the current domain's LDAP handle used for generating the query, MAY be cached within the abstract element **GPLink List** for later use.

    3. If any of the LDAP SearchRequest Filter parameter GPO DNs are not in the domain specified by Policy Target Domain Name, then for each new domain:

        1. Execute the LDAP binding sequence described in DC Discovery and AD Connection Establishment (section 3.2.5.1.1).

        2. Disable LDAP_OPT_REFERRALS as described in [MS-ADSO] section 6.2.6.1.2, Setting an LDAP Option on an ADConnection.

        3. Repeat the sequence of steps 4.1-4.3 for querying the GPO attributes.

    4. For each GPO in group, the following file access sequences MUST be generated to retrieve the GPO file system version.

- File Open request for the gpt.ini file (described in section 2.2.4) stored on the Server.
- One or more file reads MUST be done until either the entire contents of the opened file are read or an error in reading occurs.
- A file close operation MUST then be issued.

If there are any errors in processing the previous messages, policy application MUST be terminated and an event SHOULD be logged using an implementation-specific mechanism.<29>

6. For each DN in the GPLink list, beginning with the first element:
   - If the GPO was returned in the LDAP searchResponse (it is expected that not all GPOs in the search will be returned due to access control issues or replication issues), an element MUST be added to GPO list as follows:
     1. The GPO Versions field MUST be updated by the GPO container version and GPO file system version, as specified in section 3.2.1.4.
     2. The Scoped GPO DN field MUST be set to a Unicode computer-scoped GPO DN or user-scoped GPO DN by prefixing "CN=Machine," or "CN=User," to the current DN of GPLink list.
     3. The Scoped GPO Path field MUST be set to a Unicode computer-scoped GPO path or **user-scoped GPO path** by appending "\Machine," or "\User," to the transformed value of the directory string attribute **gPCFileSysPath** in the LDAP searchResponse.
     4. The GPO GUID field MUST be set to the value of the **cn** attribute in the LDAP searchResponse.
     5. The **displayName** field MUST be set to the value of the **displayName** attribute in the LDAP searchResponse.
     6. The **ExtensionList** field MUST be set to an array of curly braced GUID strings formed by parsing CSE GUID from the value of the **gPCMachineExtensionNames** or **gPCUserExtensionNames** attributes in the LDAP searchResponse.
     7. The **FunctionalityVersion** field MUST set the value of the **gPCFunctionalityVersion** attribute in the LDAP searchResponse.
     8. The **SecurityDescriptor** field MUST be set to the value of the **ntSecurityDescriptor** attribute in the LDAP searchResponse.
     9. The **WMI Filter** field MUST be updated with the value of the **gpcWQLFilter** attribute, if present in the LDAP searchResponse.
   - If the GPO was not returned in the LDAP searchResponse, the GPO MUST be ignored.
7. For each GPO in the GPO list, beginning with the first element:
   1. The checks specified in section 3.2.5.1.6 MUST be performed.
   2. If the represented GPO passes access checking:
      1. WMI filter evaluation (section 3.2.5.1.7) MAY be performed.
      2. If the represented GPO is considered allowed, append it to abstract element Filtered GPO list (Public).

If there are any errors in processing the previous messages, policy application MUST be terminated and an event SHOULD be logged using an implementation-specific mechanism.<30>

#### 3.2.5.1.6 GPO Filter Evaluation

In this step, the Client MUST process the GPOs as follows:

1. Check for the functionality version of the GPO. If the **gPCFunctionalityVersion** field of the Group Policy Object Search message (as defined in [MS-ADA1] section 2.278) is not set to 2, the GPO MUST NOT be included in the rest of the protocol sequence. The GPO MUST be considered denied.

2. Check whether the GPO has been disabled. The GPO MUST be considered denied in either of the following two cases:

   - If the Flags field of the results of the Group Policy Object Search message has the 0x00000001 flag set, and if the policy application is part of user policy mode.

   - If the Flags field of the results of the Group Policy Object Search message has the 0x00000002 flag set, and if the policy application is part of computer policy mode.

3. Perform security filtering. The **ntSecurityDescriptor** attribute that is retrieved for each GPO MUST be treated as a security descriptor, as specified in [MS-ADLS] section 2.253. This security descriptor, **discretionary access control list (DACL)**, MUST be checked for an **access control entry (ACE)** that grants the extended right ApplyGroupPolicy (as specified in [MS-ADTS] section 5.1.3.2.1) to an Active Directory security group for which the policy target account is a member. If the right is denied by an ACE for which the policy target account is a member, the GPO is to be considered denied. Otherwise, the entry grants that right, and that GPO is to be considered allowed.

4. Checks for an empty GPO: GPO MUST be considered denied if the GPO versions consisting of GPO container version and GPO file system version are both 0. The GPO attribute **versionNumber** stores the 32-bit container version in Active Directory.

#### 3.2.5.1.7 WMI Filter Evaluation

The Client MUST process the GPO to evaluate the WMI filter as follows:

1. Client MUST parse the **gPCWQLFilter** attribute in the GPO structure and extract the WMI filter ID and domain name of the WMI filter.

2. Client MUST make a WMI Filter Search as specified below with the WMI filter ID and domain name that was computed in step 1. If this step fails due to a failure that is returned from the LDAP messages, the WMI filter evaluation MUST be skipped, and the GPO MUST be assumed to be allowed.

   - An LDAP SearchRequest as specified in section 2.2.5 MUST be sent from the Client to the GP server, and the SearchResponse received MUST be verified to satisfy the specified requirements.

3. The WQL query filter that is retrieved in the LDAP **msWMI-Parm2** attribute MUST be evaluated by locally invoking the **IWbemServices::ExecQuery** method (as specified in [MS-WMI] section 3.1.4.3.18) with the following parameters:

   - The value of the **msWMI-Parm2** attribute for the *strQuery* parameter.

- WBEM_FLAG_RETURN_IMMEDIATELY and WBEM_FLAG_FORWARD_ONLY for the *lFlags* parameter.
- NULL for the *pCtx* parameter.

If the method call is successful, the Client MUST invoke the enumerator methods (specified in [MS-WMI] section 3.1.4.4) on the returned **IEnumWbemClassObject** object (in the *ppEnum* parameter) and ensure that there is at least one CIM object returned in the query result set.

If the WMI filter cannot be evaluated due to some local error on the Client, policy application MUST be terminated and an event SHOULD be logged using an implementation-specific mechanism.<31>

4. If the WMI query returns no results, the GPO MUST be considered denied; otherwise, the GPO MUST be considered allowed.

#### 3.2.5.1.8 AD Connection Termination

The Client performs the termination of the Active Directory connection with the Group Policy Server by invoking the "Performing an LDAP Unbind on an ADConnection" task defined in [MS-ADSO] section 6.2.6.1.5, with the following parameter:

- *TaskInputADConnection*: Value of the **GP Client AD Connection Handle** ADM element.

#### 3.2.5.1.9 Link Speed Discovery

The Client attempts to estimate the speed of the link between the Client and the domain controller, as specified in section 2.2.6. The domain controller used MUST be the domain controller discovered in section 3.2.5.1.1.

#### 3.2.5.1.10 Extension Protocol Sequences

The Client MUST use the abstract element Filtered GPO List (Public), which contains different classes of settings used to invoke the Group Policy Extension specific to each class of settings. A failure in any Group Policy Extension sequence MUST NOT affect the execution of other Group Policy Extensions. Applicability is determined as specified in section 3.2.1.5. The Group Policy Registry Extension MUST always execute first. All remaining Group Policy Extensions registered on the client MUST be loaded and executed in ascending order by Group Policy Extension GUID.

Additionally, each Group Policy Extension's specific protocol sequence MUST only contain those GPOs that are marked as containing those Group Policy Extensions; the **gPCUserExtensionNames** and **gPCMachineExtensionNames** attributes retrieved from the GPO search for each GPO contains CSE GUIDs that correspond to the Group Policy Extension sequences that, if present, are invoked for that GPO.

As a result, each Group Policy Extension sequence only generates traffic that references GPOs in which that Group Policy Extension's CSE GUID was present in the **gPCUserExtensionNames** attribute for the user policy mode, and only those GPOs with the CSE GUID present in **gPCMachineExtensionNames** for the computer policy mode.

The behavior of a given Group Policy Extension is specific to each Group Policy Extension and is specified in the documentation of that Group Policy Extension. A failure in any Group Policy Extension sequence does not cause the policy application sequence to fail. Failure simply means that Clients are not able to enforce settings that are associated with that specific Group Policy Extension. For example, if the Group Policy: IP Security (IPSec) Protocol Extension (as specified in [MS-GPIPSEC]) sequence fails, the computer will not be configured according to the network

administrator's IP security policy settings. This might mean that the computer cannot access some network resources that are secured through IP security. Other Group Policy Extensions are not directly affected by the failure of the Group Policy: IP Security (IPSec) Protocol Extension. For example, if the Group Policy: IP Security (IPSec) Protocol Extension fails, the Group Policy: Scripts Extension Encoding (as specified in [MS-GPSCR]) protocol sequence MUST still be invoked by the Client.

If the determined Link Speed (section 3.2.5.1.9) is below an implementation defined threshold, an implementation SHOULD NOT invoke any Group Policy Extension sequence that is bandwidth intensive.<32>

An implementation-specific means SHOULD be provided to allow for the addition of Group Policy Extensions.

#### 3.2.5.1.11 Policy Application Notification

The Client MUST raise the Policy Application Event.

### 3.2.5.2 GPO Processing Order

The Filtered GPO List is ordered according to the following processing order.

1. The Local Group Policy Object.<33>
2. Group Policy Objects linked to Site.
3. Group Policy Objects linked to Domain.
4. Group Policy Objects linked to Organizational Units: Group Policy Objects that are linked to the organizational unit that is highest in the Active Directory hierarchy are processed first, then Group Policy Objects that are linked to its child organizational unit, and so on. Finally, the Group Policy Objects that are linked to the organizational unit that contains the user or computer are processed.

## 3.2.6 Timer Events

Computer Periodic Refresh timer: When the Computer Periodic Refresh timer expires, the Client SHOULD set the Policy Application Mode to Computer and attempt to apply the policy, as described in section 3.2.5.1. The Client SHOULD also restart the timer.

User Periodic Refresh timer: When the User Periodic Refresh timer expires, the Client SHOULD set the Policy Application Mode to User and attempt to apply the policy for that user as described in section 3.2.5.1. The Client SHOULD also restart the timer.

## 3.2.7 Other Local Events

The Policy Application Mode is initialized to the Computer at the time the computer boots, or to the User at the time a user logs on. If **Policy Application Mode** is Computer, the Policy Target Domain Name (section 3.2.1.16) is initialized to the value specified by **DomainName.FQDN** (as specified in [MS-DISO] section 4.3.1.1). If **Policy Application Mode** is User, the **user principal name (UPN)** for the user is obtained by implementation specific means.<34> The domain portion of this UPN is then assigned to **Policy Target Domain Name**. If **DomainName.FQDN** is NULL and policy target is **computer account**, the policy application MUST NOT be invoked. Policy application SHOULD be invoked at the following times:

1. When the computer boots.

2. When a user logs on.

3. Periodic timer expiration as specified in section 3.2.2.

4. User initiation, that is, a user MAY manually initiate a process that causes the Group Policy: Core Protocol to immediately attempt to get the Client's state in compliance with the most recent policy settings that are stored in GPOs on the server.

5. A computer regains network connectivity to a Group Policy Server after a prior policy application failure due to the lack of network connectivity to a Group Policy Server.<35>

If Computer Periodic Refresh timer (specified in section 3.2.2) is present, it SHOULD be started at the time the computer boots. If User Periodic Refresh timer (specified in section 3.2.2) is present, it SHOULD be started at the time a user logs in.

## 3.3 Administrative Tool Details

### 3.3.1 Abstract Data Model

The administrative tool abstract data model contains the GP server model described in section 3.1.1. It also contains the following concepts: Group Policy Protocol Administrative Tool (section 3.3.1.1), Group Policy Extension Administrative Plug-In (section 3.3.1.2), and Administered GPO (section 3.3.1.3).

#### 3.3.1.1 Group Policy Protocol Administrative Tool

The Group Policy Protocol Administrative Tool is an entity that determines the abstract data model for GPOs, except for the abstract data models of the Group Policy Extensions of the GPO. It operates on a particular GPO, the Administered GPO.

#### 3.3.1.2 Group Policy Extension Administrative Plug-In

The Group Policy Extension Administrative Plug-In is an entity that determines a specific Group Policy Extension for updating and reading that Group Policy Extension's settings to and from a GPO, but it does not understand how to alter a GPO. However, it is capable of invoking the Group Policy: Core Protocol's Group Policy Extension update sequence. In common usage, the Group Policy Protocol administrative tool invokes the plug-in.

#### 3.3.1.3 Administered GPO (Public)

The Administered GPO ADM element is the Group Policy Object administered by the Administrative Tool.

### 3.3.2 Timers

None.

### 3.3.3 Initialization

When the Group Policy Protocol administrative tool starts, the administrator selects a GPO GUID to edit a GPO or enters a new GPO GUID to create a GPO. When creating a new GPO, the GPO Creation Message (section 2.2.8.1) MUST be used. When the GPO is being edited, the tool MUST attempt to access that GPO and read the GPO's user and computer Group Policy Extension lists to determine the Group Policy Extension administrative tool plug-ins that are needed to read or write settings in the GPO. It does this by using a GPO Read Administration (section 2.2.7) message.

After this action, the Group Policy Protocol administrative tool MAY invoke the correct Group Policy Extension, depending on user input, that direct the tool to show some (or all) of the settings, or to allow those settings to be changed.

### 3.3.4 Higher-Layer Triggered Events

#### 3.3.4.1 Group Policy Creation

The Group Policy Creation occurs whenever an administrator uses a Group Policy Administration tool to create a GPO. This triggers a GPO Creation (section2.2.8.1) message.

Parameters to the Group Policy Creation event are:

| Parameter | Description |
|---|---|
| DN of domain naming context | The distinguished name (DN) for the domain where the new Group Policy object will be created. |

#### 3.3.4.2 Group Policy Property Update

The Group Policy property update occurs whenever an administrator uses a Group Policy Extension's Policy Administration protocol to change properties on a GPO. This triggers a GPO Property Update (section2.2.8.3) message.

#### 3.3.4.3 The SOM Property Update

The scope of management (SOM) property update occurs whenever an administrator uses a Group Policy Extension's Policy Administration protocol to change Group Policy properties on an SOM. This triggers an SOM Property Update Message (section 2.2.8.4).

Parameters to the **SOM Property Update** event are:

| Parameter | Description |
|---|---|
| SOM DN for the SOM being updated | The distinguished name (DN) for the object defining the scope of management to be updated. |
| attribute name | A string representing the attribute on the object referenced by the SOM DN. For example, "gpOptions". |
| attribute value | The value to be used to update the attribute named in the attribute name parameter. The data type of the attribute value depends on the data type of the attribute on the object referenced by the SOM DN. For example, the attribute "gpOptions" is defined as an integer. |

#### 3.3.4.4 Group Policy Extension Update

The Group Policy Extension settings update occurs whenever an administrator uses a Group Policy Extension's Policy Administration protocol to change a Group Policy Extension's settings in a GPO. This triggers a GPO Extension Update (section 2.2.8.2) message. The GPO attribute **versionNumber** is incremented according to the current value associated with the "Version" key in the <GPO path>\gpt.ini file.

Parameters to the Group Policy Extension Update event are the following:

| Parameter | Description |
|---|---|
| GPO DN | The distinguished name for the Group Policy Object that was updated. |
| Is User Policy | A Boolean value to indicate that this update is for user policy mode. If FALSE, this update is for computer policy mode. |
| CSE GUID | The Client-side extension's GUID of the form {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}. |
| TOOL GUID | The Administrative extension plug-in's GUID of the form {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}. |

#### 3.3.4.5 Version Number Update

GPO container version: The user GPO version part of GPO container version MUST be incremented if the user policy settings are being modified. The machine GPO version part of GPO container version MUST be incremented if computer policy settings are being modified. If, after the increment, the user GPO version or machine GPO version is 0, the value MUST be set to 1.

GPO file system version: The user GPO version part of GPO file system version MUST be incremented if the user policy settings are being modified. The machine GPO version part of the GPO file system version MUST be incremented if the computer policy settings are being modified. If, after the increment, the user GPO version or machine GPO version is 0, the value MUST be set to 1.

If the user policy settings are being modified then both the GPO container version and the GPO file system version for the user GPO version part MUST be updated as mentioned previously.

If the computer policy settings are being modified then both the GPO container version and the GPO file system version for the machine GPO version part MUST be updated as mentioned previously.

#### 3.3.4.6 Group Policy Deletion

The Group Policy Deletion occurs whenever an administrator uses a Group Policy Administration tool to delete a GPO. This triggers a GPO Deletion Message (section 2.2.8.5). The parameter to the Group Policy Deletion event is a DN of the GPO to be deleted.

### 3.3.5 Message Processing Events and Sequencing Rules

None.

#### 3.3.5.1 GPO Creation

Creation of a GPO requires the creation of a groupPolicyContainer Active Directory object on the GP server and a corresponding directory on the GP server's SYSVOL share. The creation of the Active Directory portion of the GPO MUST be accomplished through an LDAP addRequest message (as described in the specification of the GPO Creation Message, section 2.2.8.1) from the Client to the Server. Prior to the creation of the Active Directory portion of the GPO, the parent Active Directory policies container is created through an LDAP addRequest message.

1. Create Policies container as shown in an existing message specified in section 2.2.8.1. If the container exists, the "object already exists" error MUST be ignored. Other than the "object already exists" error, if the **resultCode** field of the addResponse message is nonzero, this protocol sequence MUST be terminated.

2. Attempt to retrieve the GPO container as shown in a new message specified in section 2.2.8.1.

3. If the object does not exist, create GPO container as shown in an existing message specified in section 2.2.8.1. If the **resultCode** field of the addResponse message is nonzero, this protocol sequence MUST be terminated.

   The result of a groupPolicyContainer addRequest is an addResponse message in reply, as defined in [RFC2251] section 4.7. The **resultCode** field value determines a failure or success for the message. Success is indicated when the value of the addResponse message's **resultCode** is 0. Any other **resultCode** value indicates a failure.

The result of SearchRequest in a GPO Creation Message (section 2.2.8.1) is a reply from the GP server with one LDAP searchResponse message that contains one searchResultEntry, as specified in [RFC2251] section 4.5.2.

The searchResultEntry MUST contain an attributes field with the values in Active Directory for the **ntSecurityDescriptor** attribute of the GPO object that was searched for.

After the groupPolicyContainer object is created, create the machine and user container objects:

1. Attempt to retrieve the machine container as shown in a message, as specified in section 2.2.8.1.
2. If the object does not exist, create machine container as shown in a message, as specified in section 2.2.8.1. If the **resultCode** field of the addResponse message is nonzero, this protocol sequence MUST be terminated.
3. Attempt to retrieve the user container as shown in a message, as specified in section 2.2.8.1.
4. If the object does not exist, create user container as shown in a message, as specified in section 2.2.8.1. If the **resultCode** field of the addResponse message is nonzero, this protocol sequence MUST be terminated.

The following messages make up the remainder of the GPO Creation messages:

1. File Status request for the directory GPO Path. If the GPO Path exists, the sequence MUST be terminated.
2. Create Directory request for the directory GPO Path.
3. Modify the security descriptor on the directory to the owner, primary group, and DACL as specified in the **ntSecurityDescriptor** GPO attribute using an implementation-specific method.<36>
4. Create File request for the file GPO path\gpt.ini.
5. Write File request to write the contents as outlined in section 2.2.4 with the required section, "General"; the key, "Version"; and the value, 0 (integer).
6. Create Directory request for the directory user scoped GPO path.
7. Create Directory request for the directory computer-scoped GPO path.

Any failures from the file operations mean that the overall GPO Creation Message (section 2.2.8.1) is invalid, and the sequence previously mentioned MUST be terminated.

### 3.3.5.2 GPO Extension Update

Whenever an administrative tool invokes a Group Policy Extension plug-in for a GPO and that plug-in modifies the GPO, the Group Policy Extension plug-in invokes the GPO Extension Update sequence,

which produces the LDAP modifyRequest message (as described in the specification of the GPO Extension Update Message, section 2.2.8.2) from the Client to the Server.

If the value of the modifyResponse message's **resultCode** is integer 0, it indicates success. Any other **resultCode** value indicates a failure.

The GPO File System Version Update (section 3.3.5.4) file access messages make up the remainder of the GPO Extension Update message.

### 3.3.5.3 GPO Property Update

Whenever an administrative tool modifies the properties of a GPO, it produces the LDAP modifyRequest message (as described in the specification of the GPO Property Update Message section 2.2.8.3) from the Client to the Server.

If the value of the modifyResponse message's **resultCode** is integer 0, it indicates success. Any other **resultCode** value indicates a failure.

When the **nTSecurityDescriptor** attribute is modified in the GPO Property Update Message (section 2.2.8.3), the following file access message is included in the GPO Property Update message:

Modify the security descriptor on the directory to the value of the **nTSecurityDescriptor** GPO attribute using an implementation specific method.<37>

The GPO File System Version Update (section 3.3.5.4) file access messages make up the remainder of the GPO Property Update message.

### 3.3.5.4 GPO File System Version Update

The following file access messages make up the GPO file system version update sequence:

1. Open the file GPO path\gpt.ini for read/write access.
2. Read the contents of the Value corresponding to Key "Version".
3. Increment the GPO file system version.
    1. For user policy mode, increment the upper 16-bit version; or
    2. For computer policy mode, increment the lower 16-bit version.
4. Write the GPO file system version as the value corresponding to key "Version".
5. Close the file.

### 3.3.5.5 SOM Property Update

Whenever an administrative tool modifies the properties of the SOM, it produces the LDAP modifyRequest message (as described in the specification of the SOM Property Update Message, section 2.2.8.4) from the Client to the Server.

If the value of the modifyResponse message's **resultCode** is integer 0, it indicates success. Any other **resultCode** value indicates a failure.

### 3.3.5.6 GPO Deletion

Deletion of a GPO requires the deletion of the GPO's Active Directory object on the GP server and a corresponding directory on the GP server's SYSVOL share. The deletion of the Active Directory portion of the GPO MUST be accomplished through an LDAP delRequest message as described in the specification of the GPO Deletion Message (section 2.2.8.5) from the Client to the Server.

The result of delRequest is a delResponse message in reply, as defined in [RFC2251] section 4.8. The **resultCode** field value determines a failure or success for the message. Success is indicated when the value of the delResponse message's **resultCode** is 0. Any other **resultCode** value indicates a failure.

1. Open directory file at <GPO path>
2. Enumerate contents of current directory
3. For each directory entry
   - If entry is a directory file
     - Repeat steps 2 and 3, enumerating contents of subdirectory
     - Delete directory file
   - Else
     - Delete file
4. Delete directory file at <GPO path>

A GPO is an Active Directory container so an LDAP delRequest message MUST be sent for all Active Directory objects contained in the GPO and recursively for each subcontainer and all Active Directory objects contained in the subcontainer before it is sent for the GPO. Starting at the GPO, an LDAP SearchRequest MUST be sent to the GP server with the following parameters:

| Parameter | Value |
|---|---|
| *baseObject* | LDAP DN for the current container (starting with the GPO DN). |
| *Scope* | MUST be set to 1. Search all entries in the first level below the *baseObject* excluding the *baseObject*. |
| *derefAliases* | MUST be set to 0 (neverDerefAliases). |
| *sizeLimit* | No limit is set (this MUST be set to 0). |
| *timeLimit* | MAY be 0 (infinite). |
| *typesOnly* | MUST be set to 0 (FALSE). |
| *Filter* | The following LDAP filter (using the representation as specified in [RFC2254]) MUST be used:<br>(objectClass=*) |
| *attributes* | objectClass |

For each returned object, if the **objectClass** attribute is equal to "container", the object DN MUST be used as the *baseObject* for an LDAP SearchRequest recursively until the GPO contains no objects.

If the **objectClass** attribute is not equal to "container", an LDAP delRequest message MUST be sent for the object. The final LDAP delRequest message MUST be for the GPO DN. If the **resultCode** field of a delResponse message is nonzero, the error condition must be logged.

The following steps make up the remainder of GPO Deletion:

1. A domain SOM search as defined in section 2.2.2 except for these fields:

| Parameter | Value |
|---|---|
| *baseObject* | LDAP DN for the root of the domain. |
| *Scope* | MUST be the whole subtree (2). |
| *Filter* | The following LDAP filter (using the representation as specified in [RFC2254]) MUST be used:<br>(&(\|(objectcategory=domaindns)(objectcategory=organizationalUnit))(gplink=*)) |

2. A site search as defined in the first part section 2.2.3 which retrieves the **configurationNamingContext**. The second search is identical except for these fields:

| Parameter | Value |
|---|---|
| *baseObject* | cn=Sites,<LDAP DN for the configurationNamingContext of the domain.> |
| *Scope* | MUST be the whole subtree (2). |
| *Filter* | The following LDAP filter (using the representation as specified in [RFC2254]) MUST be used:<br>(objectCategory=site) |

3. For each SOM object returned in Step 1, a SOM property update message for attribute **gPLink** removing the GPO DN from the list of linked GPO objects.

4. For each Site object returned in Step 2, a SOM property update message for attribute **gPLink** removing the GPO DN from the list of linked GPO objects.

### 3.3.5.7 GPO Link Creation, Update and Deletion

Whenever an administrative tool creates, updates or deletes a link from a GPO to a SOM, the "gpLink" attribute on the SOM is updated, which produces an LDAP modifyRequest message from the Client to the Server (as described in the specification of the SOM Property Update Message, section 2.2.8.4).

The result of modifyRequest is a modifyResponse message in reply, as defined in [RFC2251] section 4.6. The **resultCode** field value determines a failure or success for the message. If the value of the modifyResponse message's **resultCode** is integer 0, it indicates success. Any other **resultCode** value indicates a failure.

Parameters to the GPO Link Creation and GPO Link Deletion event are:

| Parameter | Description |
|---|---|
| SOM DN for the SOM being updated | The distinguished name for the object defining the SOM to be updated. |

| Parameter | Description |
|---|---|
| GPO DN | The distinguished name for the GPO to be created or deleted from the GPO Link list. |
| GPO Link list | A list of GPO and GPO Link Options as defined in the "gpLink" attribute on the object referred to by the SOM DN. |
| GPO Link position | In a GPO Link Creation, a GPO Link position MAY be specified which defines the link order for the GPO. |
| GPO Link Options | In a GPO Link Creation, GPO Link Options MAY be specified. The value for GPO Link Options is defined in the "gpLink" description in section 2.2.2. |

For a GPO Link Creation, the GPO DN and GPO Link Options specified MUST be inserted in the GPO Link list at the position specified by the GPO Link position parameter or at the end of the GPO Link List if GPO Link position is unspecified. A string as defined in the "gpLink" description in section 2.2.2 MUST be created by enumerating the GPO Link list elements in order. The string is used as the attribute value on the SOM Property Update as defined below.

For a GPO Link Deletion, the GPO DN specified MUST be located by comparing the GPO DN parameter against the GPO DN in each GPO Link in the GPO Link list. If the GPO DN is located in the list, it MUST be removed from the GPO Link list. A string as defined in the "gpLink" description in section 2.2.2 MUST be created by enumerating the GPO Link list elements in order. The string is used as the attribute value on the SOM Property Update as defined below.

The link order for a GPO can be updated by combining a GPO Link Deletion and a GPO Link Creation at the desired GPO Link position.

Parameters to the SOM Property Update event are:

| Parameter | Description |
|---|---|
| SOM DN for the SOM being updated | The distinguished name for the object defining the scope of management to be updated. |
| attribute name | The string "gpLink". |
| attribute value | The format of this string is defined in the "gpLink" description in section 2.2.2. |

### 3.3.6 Timer Events

None.

### 3.3.7 Other Local Events

None.

# 4 Protocol Examples

This section provides examples of how to use Group Policy to perform a representative subset of functions.

## 4.1 Domain SOM Search and Reply Messages

The following sections describe the message exchange with a Group Policy Server in order to obtain the SOMs for a computer account, as described in section 2.2.2 (steps 2 and 3).

In this example, the process is initiated with a search message query sent to the Group Policy Server and ends with receipt of two SOMs for the specified account.

### 4.1.1 Domain SOM Search Message

This section describes the initial Search message sent to the Group Policy Server to obtain the SOM for a computer account, as described in section 2.2.2 (step 2).

In this example, the computer policy target account is identified by its Computer-Scoped GPO DN, "CN=LABSERVER,OU=ComputersOU,DC=test,DC=contoso,DC=com". The computer policy target account is located in the OU, "OU=ComputersOU,DC=test,DC=contoso,DC=com" and the root of the domain is "DC=test,DC=contoso,DC=com".

This message has the following form.

| Parameter | Value |
|---|---|
| **baseObject** | DC=test,DC=contoso,DC=com |
| **scope** | 2 |
| **derefAliases** | 0 |
| **sizeLimit** | 0 |
| **timeLimit** | 240 |
| **typesOnly** | 0 |
| **filter** | (\|(distinguishedName=OU=ComputersOU,DC=test,DC=contoso,DC=com)(distinguishedName=DC=test,DC=contoso,DC=com)) |
| **attributes** | gpLink, gpOptions |

### 4.1.2 Domain SOM Reply Message

This section describes the message received from the Group Policy Server in response to the query message sent in section 4.1.1 of this example. This response contains the SOMs for a computer account, as described in section 2.2.2 (step 3).

The computer policy target account's Computer-Scoped GPO DN is "CN=LABSERVER,OU=ComputersOU,DC=test,DC=contoso,DC=com". The reply contains two SOMs. In the first, searchResultEntry has the value "1" for the OU, "OU=ComputersOU,DC=test,DC=contoso,DC=com". In the second, searchResultEntry has the value "2" for the domain root, "DC=test,DC=contoso,DC=com".

This message has the following form.

| searchResultEntry | Attribute | Value |
|---|---|---|
| 1 | DN | OU=ComputersOU,DC=test,DC=contoso,DC=com |
| 1 | gpLink | [LDAP://cn={D57B125B-5E65-48DF-A123-CF6262607BB6},cn=policies,cn=system,DC=test,DC=contoso,DC=com;0] |
| 1 | gPOptions | 0 |
| 2 | DN | DC=test,DC=contoso,DC=com |
| 2 | gpLink | [LDAP://CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System,DC=test,DC=contoso,DC=com;0] |
| 2 | gPOptions | 0 |

## 4.2 Site Search Messages

The following sections describe the message exchange with a Group Policy Server in which a Client requests and receives the site where a computer account is located. This procedure is described in section 2.2.3.

In this example, the process is initiated with a Site Search message sent to the Group Policy Server. The server replies with the configurationNamingContext. This configurationNamingContext is used in a second Site Search message. The process ends with receipt of the scope of management for the site to which the computer account belongs.

### 4.2.1 Site Search configurationNamingContext Request Message

This section describes the initial query message sent to the Group Policy Server to obtain the configurationNamingContext for the computer policy target account "CN=LABSERVER,OU=ComputersOU,DC=test,DC=contoso,DC=com". This procedure is described in step 2 in section 2.2.3.

| Parameter | Value |
|---|---|
| baseObject | Zero-length string |
| scope | 0 |
| derefAliases | 0 |
| sizeLimit | 1 |
| timeLimit | 240 |

| Parameter | Value |
|---|---|
| typesOnly | 0 |
| filter | (objectClass=*) |
| attributes | configurationNamingContext |

### 4.2.2 Site Search configurationNamingContext Reply Message

This section describes the response to the Site Search message sent in section 4.2.1. In this case, the configurationNamingContext attribute returned from the Group Policy Server is "CN=Configuration,DC=test,DC=contoso,DC=com".

### 4.2.3 Site Search SOM Request Message

This section describes the second Site Search message sent to the Group Policy Server. This message requests the scope of management of the site in which the computer account is located. The baseObject value contains the distinguished name of the site.

The site distinguished_name is computed from the site name combined with the configurationNamingContext value obtained in section 4.2.2. The site to which the Client computer belongs (the SiteName) is detailed in [MS-DISO] section 4.3.1.1. In this example, the computer policy target account belongs to the site "NA-WA-RED". In this example, the distinguished_name is "CN=NA-WA-RED,CN=Sites,CN=Configuration,DC=test,DC=contoso,DC=com". For more details, see step 4 in section 2.2.3.

| Attribute | Value |
|---|---|
| baseObject | CN=NA-WA-RED,CN=Sites,CN=Configuration,DC=test,DC=contoso,DC=com |
| scope | 0 |
| derefAliases | 0 |
| sizeLimit | 0 |
| timeLimit | 240 |
| typesOnly | 0 |
| filter | (objectClass=*) |
| attributes | gpLink, gpOptions |

The response to the preceding message is identical to the response for the Domain SOM Search message as described in the example in section 4.1.2.

## 4.3 GPO Search Message and Reply

The following sections describe the message exchange with a Group Policy Server in which a Client requests and receives the site where a computer account is located. This procedure is described in section 2.2.4.

In this example, the process is initiated with a Site Search message sent to the Group Policy Server. The server replies with the configurationNamingContext. This configurationNamingContext is used in

a second Site Search message. The process ends with receipt of the SOM for the site to which the computer account belongs.

### 4.3.1 GPO Search Message

The following table lists the values of attributes that are sent to the GP server in the Group Policy Object Search message, as specified in step 2 of section 2.2.4. The query is sent requesting all the GPOs matching under the subtree of "cn=policies,cn=system,DC=test,DC= contoso,DC=com". The GPO paths passed in the filter are as obtained in the response for section 4.1.2 and reply for section 4.2.3.

| Attribute | Value |
|---|---|
| baseObject | cn=policies,cn=system,DC=test,DC= contoso,DC=com |
| scope | 2 |
| derefAliases | 0 |
| sizeLimit | 0 |
| timeLimit | 240 |
| typesOnly | 0 |
| filter | (\|(distinguishedName=CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System ,DC=test,DC=contoso,DC=com)(distinguishedName=cn={D57B125B-5E65-48DF-A123-CF6262607BB6},cn=policies,cn=system,DC=test,DC=contoso,DC=com)) |
| attributes | nTSecurityDescriptor, cn, displayName, gPCFileSysPath, versionNumber, gPCMachineExtensionNames, gPCUserExtensionNames, gPCFunctionalityVersion, flags, and gPCWQLFilter. |

### 4.3.2 GPO Search Reply Message

The following table lists the values of attributes that are returned from an LDAP (for more information regarding LDAP, see [RFC2251]) search that was part of a GPO Search, as specified in section 2.2.4. In this example, the GPO that is returned is named Default Domain Policy and has a GPO GUID of 31B2F340-016D-11D2-945F-00C04FB984F9. The Computer section of the GPO contains settings for the Group Policy: Registry Extension Encoding (as specified in [MS-GPREG]) and the Group Policy: Scripts Extension Encoding (as specified in [MS-GPSCR]) client-side plug-ins, in addition to the Administrative Templates plug-in (as specified in [MS-GPREG]) and the Scripts Extension administrative plug-in (as specified in [MS-GPSCR]). The User section of the GPO contains settings for the Group Policy Registry Extension Administrative plug-in and the Administrative Templates administrative plug-in.

| Attribute | Value |
|---|---|
| cn | {31B2F340-016D-11D2-945F-00C04FB984F9} |
| displayName | Default Domain Policy |
| gPCFileSysPath | \\jdoe_pc.test.contoso.com\sysvol\ jdoe_pc.test.contoso.com\Policies\{31B2F340-016D-11D2-945F-00C04FB984F9} |

| Attribute | Value |
|---|---|
| versionNumber | 65537 |
| gPCMachineExtensionNames | [{35378EAC-683F-11D2-A89A-00C04FBBCFA2}{0F6B957E-509E-11D1-A7CC-0000F87571E3}][ {42B5FAAE-6536-11d2-AE5A-0000F87571E3} {40B6664F-4972-11D1-A7CA-0000F87571E3}] |
| gPCUserExtensionNames | [{35378EAC-683F-11D2-A89A-00C04FBBCFA2}{0F6B957E-509E-11D1-A7CC-0000F87571E3}] |
| gPCFunctionalityVersion | 2 |
| flags | 0 |
| gPCWQLFilter | Not set |

## 4.4 WMI Filter Search and Reply Messages

### 4.4.1 WMI Filter Search Message

The following table lists the values of attributes that are sent to the Group Policy Server for WMI filter Search message, as specified in step 2 of section 2.2.5. The WMI filter is obtained from the value of attribute gPCWQLFilter in the GPO Search Reply Messages, as specified in 2.2.4. This is the WMI FILTER ID value as described under the **gPCWQLFilter** attribute.

| Attribute | Value |
|---|---|
| baseObject | CN={A5B195C1-7D26-451B-9819-0A92F10EFEB9},CN=SOM,CN=WMIPolicy,CN=System,DC=test,DC=contoso,DC=com |
| scope | 0 |
| derefAliases | 0 |
| sizeLimit | 0 |
| timeLimit | 0 |
| typesOnly | 0 |
| filter | (objectclass=*) |
| attributes | msWMI-ID, msWMI-Name, msWMI-Parm1, msWMI-Author, msWMI-ChangeDate, msWMI-CreationDate, and msWMI-Parm2. |

### 4.4.2 WMI Filter Search Response Message

The following table lists the values of attributes that are received from the Group Policy Server as a response for WMI Filter Search message, as specified in step 3 of section 2.2.5.

| Attribute | Value |
|---|---|
| msWMI-ID | {A5B195C1-7D26-451B-9819-0A92F10EFEB9} |
| msWMI-Name | Test WMI Filter |

| Attribute | Value |
|---|---|
| msWMI-Parm1 | Description of Test WMI filter |
| msWMI-Author | Admin@test.contoso.com |
| msWMI-ChangeDate | 20070723220731.328000-000 |
| msWMI-CreationDate | 20070723220731.328000-000 |
| msWMI-Parm2 | 1;3;10;18;WQL;root\CIMv2;Select * from Win32_Printer; |

## 4.5 GPO Read Administration Request and Reply Messages

The following section describes the message exchange with a Group Policy Server in order to obtain settings and state of an individual GPO.

In this message, the query is for the "Default Domain Policy" with the GPO DN of "CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System,DC=test,DC=contoso,DC=com".

| Attribute | Value |
|---|---|
| baseObject | CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System,DC=test,DC=contoso,DC=com |
| scope | 0 |
| derefAliases | 0 |
| sizeLimit | 0 |
| timeLimit | 0 |
| typesOnly | 0 |
| filter | (objectClass=*) |
| attributes | nTSecurityDescriptor, cn, displayName, gPCFileSysPath, versionNumber, gPCMachineExtensionNames, gPCUserExtensionNames, gPCFunctionalityVersion, flags, and gPCWQLFilter |

The response for the preceding message is identical to the reply described in section 4.3.2.

## 4.6 GPO Creation Message

In this example, a GPO with the DN of "CN={1FE2ABF4-613E-4980-BA93-74F7B206A6C1},CN=Policies,CN=System,DC=test,DC=contoso,DC=com" is created, and the new GPOs GUID is "{1FE2ABF4-613E-4980-BA93-74F7B206A6C1}". This message is described in section 2.2.8.1.

| Parameter | Value |
|---|---|
| entry | CN={1FE2ABF4-613E-4980-BA93-74F7B206A6C1},CN=Policies,CN=System,DC=test,DC=contoso,DC=com |
| attributes | objectClass: groupPolicyContainer |

| Parameter | Value |
|---|---|
| | versionNumber: 0<br>Flags: 0 |

On successful creation of the GPO, the Client issues messages to create the user and machine subcontainers as shown in the following table.

User container: (the attributes field MUST contain one attribute: objectClass).

| Parameter | Value |
|---|---|
| entry | CN=user,CN={1FE2ABF4-613E-4980-BA93-74F7B206A6C1},CN=Policies,CN=System,DC=test,DC=contoso,DC=com |
| attributes | objectClass: container |

Machine container: (the attributes field MUST contain one attribute: objectClass).

| Parameter | Value |
|---|---|
| entry | CN=machine,CN={1FE2ABF4-613E-4980-BA93-74F7B206A6C1},CN=Policies,CN=System,DC=test,DC=contoso,DC=com |
| attributes | objectClass: container |

An LDAP SearchRequest MUST be sent to the GP server with the following parameters.

| Parameter | Value |
|---|---|
| baseObject | CN={1FE2ABF4-613E-4980-BA93-74F7B206A6C1},CN=Policies,CN=System,DC=test,DC=contoso,DC=com |
| scope | 0 |
| derefAliases | 0 |
| sizeLimit | 0 |
| timeLimit | 0 |
| typesOnly | 0 |
| filter | (objectclass=*) |
| attributes | **nTSecurityDescriptor** |

This is followed by the creation of GPO on the GP server's SYSVOL share. In this example, the name of the GP server machine is GPSvr1.test.contoso.com.

The following operations are involved:

- Create Directory request for directory: \\GPSvr1.test.contoso.com \sysvol\test.contoso.com\Policies\{1FE2ABF4-613E-4980-BA93-74F7B206A6C1}
- Modify the security descriptor on the directory to the value of the **ntSecurityDescriptor** Active Directory GPO attribute using an implementation-specific method.<38>

- Create File request for file: \\GPSvr1.test.contoso.com \sysvol\test.contoso.com\Policies\{1FE2ABF4-613E-4980-BA93-74F7B206A6C1}\gpt.ini
- Write File request to write the contents of file: \\GPSvr1.test.contoso.com \sysvol\test.contoso.com\Policies\{1FE2ABF4-613E-4980-BA93-74F7B206A6C1}\gpt.ini, as outlined in section 2.2.4 with the required section, "General"; the key, "Version"; and the value, "0".
- Close request for the opened file.
- Create Directory request for directory: \\GPSvr1.test.contoso.com \sysvol\test.contoso.com\Policies\{1FE2ABF4-613E-4980-BA93-74F7B206A6C1}\User
- Create Directory request for directory: \\GPSvr1.test.contoso.com \sysvol\test.contoso.com\Policies\{1FE2ABF4-613E-4980-BA93-74F7B206A6C1}\Machine

## 4.7 GPO Extension Update Message

In this example, user policy settings are being updated for the GPO with the DN of " CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System ,DC=test,DC=contoso,DC=com" as described in section 2.2.8.2.

| Parameter | Value |
|---|---|
| entry | CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System ,DC=test,DC=contoso,DC=com |
| attributes | *Attribute names and values for this message:*<br>versionNumber: 65537<br>gPCUserExtensionNames: [{35378EAC-683F-11D2-A89A-00C04FBBCFA2}{0F6B957E-509E-11D1-A7CC-0000F87571E3}] |

This is followed by the update to the GP server's SYSVOL share. In this example, the name of the GP server machine is GPSvr1.test.contoso.com.

The following operations are involved:

- Open for the file: \\GPSvr1.test.contoso.com \sysvol\test.contoso.com\Policies\{31B2F340-016D-11D2-945F-00C04FB984F9}\gpt.ini
- Write to write "65537" as the value corresponding to key, "Version".
- Close for the opened file.

## 4.8 GPO Property Update Message

In this example the displayName attribute of the GPO with DN of " CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System,DC=test,DC=contoso,DC=com" is being updated as described in section 2.2.8.3.

| Parameter | Value |
|---|---|
| entry | CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System,DC=test,DC=contoso,DC=com |
| attributes | *Attribute name and value for this message:* |

| Parameter | Value |
|---|---|
| | displayName: Finance Department GPO |

If there is an update to the security descriptor of the GPO, that update needs to be propagated to the GP Server's SYSVOL share. In this example, the name of the GP server machine is GPSvr1.test.contoso.com

The following operation is involved to update the security descriptor of the GPO on the SYSVOL share:

Modify the security descriptor on the directory to the value of the **nTSecurityDescriptor** GPO attribute using an implementation-specific method.<39>

## 4.9 SOM Property Update Message

In this example, the GPO with the DN of "CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System,DC=test,DC=contoso,DC=com" associated with the SOM of "OU=Finance OU,DC=test,DC=contoso,DC=com" is being enforced. This message is described in section 2.2.8.4.

| Parameter | Value |
|---|---|
| entry | OU=Finance OU,DC=test,DC=contoso,DC=com |
| attributes | *Attribute name and value for this message:*<br>gpLink: [LDAP://CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System,DC=test,DC=contoso,DC=com;2] |

## 4.10 Sample gpt.ini File

The content of a sample gpt.ini file is as follows:

```
[General]
Version=9437184
```

# 5 Security

## 5.1 Security Considerations for Implementers

Implementers should note that the server might (and often does) have nearly the same policy application sequence with multiple Clients, which means that the protocol is not suitable for communicating confidential information that SHOULD be disclosed to only one computer (or to only one user) unless other security measures have been taken (such as a physical security mechanism, IP security, and so on).

Examples of such confidential information are passwords, asset account identifiers, and government-issued identification numbers. Even with additional security measures, the Group Policy: Core Protocol is not intended to transmit such sensitive information and thus SHOULD only be used to transmit administrative intentions to multiple Client computers.

Implementers should note that the GPO is made up of Active Directory objects under GPO DN and file system objects (files and directories) under the domain-based DFS path GPO path. Access to both the GPO DN and GPO path of a GPO MUST be secured to secure access to a GPO.

Implementers should note that a person with the appropriate permission on the GP server can modify the GPO settings. As specified in section 3.2.5.1.1, the Client locates the GP server - a domain controller (as specified in section 3.2.1.13) by invoking the **DsrGetDcNameEx2** method ([MS-NRPC] section 3.5.5.2.1) A domain controller, by definition, is a trusted third party for the domain, as explained in [MS-DISO] section 3.1.

## 5.2 Index of Security Parameters

| Security parameter | Section |
|---|---|
| LDAP signing | 1.5 |
| Kerberos authentication for computer policy application | 2.2 |
| SPNEGO authentication for user policy application | 2.2 |

# 6 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Microsoft Windows NT® operating system
- Microsoft Windows® 2000 operating system
- Windows® XP operating system
- Windows Server® 2003 operating system
- Windows Vista® operating system
- Windows Server® 2008 operating system
- Windows® 7 operating system
- Windows Server® 2008 R2 operating system

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

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

<1> Section 1.6: If the Client is running the Windows operating system, the Group Policy: Core Protocol is only applicable if that Client uses one of the following versions of Windows:

- Windows 2000
- Windows XP
- Windows Server 2003
- Windows Vista
- Windows Server 2008
- Windows 7
- Windows Server 2008 R2

<2> Section 1.6: If the GP server is running the Windows operating system, the GP server uses one of the following versions of Windows:

- Windows 2000
- Windows Server 2003
- Windows Server 2008

- Windows Server 2008 R2

<3> Section 1.6: Some functionality of the Group Policy: Core Protocol is not available in some versions of Windows; those exceptions are noted in the relevant sections of this document.

<4> Section 2.2.1: When the formatDesired field is set to 1, Windows Group Policy Clients only ask for one DS_NAME_RESULT_ITEMW value in the array in DS_NAME_RESULTW. If a value other than 1 is specified in formatDesired, Windows-based servers return names according to the values that are specified in [MS-DRSR] section 4.1.4.1.3. The Windows Group Policy Clients referred to here must be using one of the following versions of Windows:

- Windows 2000
- Windows XP
- Windows Vista
- Windows Server 2003
- Windows Server 2008
- Windows 7
- Windows Server 2008 R2

<5> Section 2.2.2: The *timeLimit* option is 0 (infinite) in Windows versions:

- Windows XP
- Windows 2000
- Windows Server 2003

and is 240 (seconds) in Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2.

<6> Section 2.2.3: The *timeLimit* option is 0 (infinite) in the following Windows versions:

- Windows XP
- Windows Server 2003
- Windows 2000

and is 240 (seconds) in Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2.

<7> Section 2.2.3: The *timeLimit* option is 0 (infinite) in the following Windows versions:

- Windows XP
- Windows Server 2003
- Windows 2000

and is 240 (seconds) in Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2.

<8> Section 2.2.4: The *timeLimit* option is 0 (infinite) in the following Windows versions:

- Windows XP
- Windows Server 2003
- Windows 2000

and is 240 (seconds) in Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2.

<9> Section 2.2.5: This message is only generated by Clients that run

- Windows XP
- Windows Server 2003
- Windows Vista
- Windows Server 2008
- Windows 7
- Windows Server 2008 R2

<10> Section 2.2.6: Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2 use normal protocol traffic to determine link speed. Windows 2000, Windows XP, and Windows Server 2003 use ICMP to determine the link speed between the Client and the domain controller. The following algorithm is used to determine the link speed when ICMP is used.

1. An ICMP Echo request with a packet size between 500–2,048 bytes is formed.
2. The request is sent to the domain controller three times, and the round-trip time for each of the echo responses is computed.
3. The packet size divided by average response time is used as the estimate of the link speed between the Client and the domain controller.

<11> Section 2.2.7: In Windows, the administrative tool specifies no attributes. This causes the GPserver to return the entire GPO and all its attributes.

<12> Section 3.2.1.1: The following Windows-based Clients make use of this optimization:

- Windows 2000
- Windows XP
- Windows Server 2003
- Windows Vista
- Windows Server 2008
- Windows 7
- Windows Server 2008 R2

<13> Section 3.2.1.2: In Windows, the default value of User Policy Source Mode is read from the machine-specific Registry Policy file in the following location. If that value is missing, the default value of User Policy Source Mode is Normal Mode.

- **Key:** Software\Policies\Microsoft\Windows\System
- **Value:** UserPolicyMode
- **Type:** REG_WORD
- **Size:** 4
- **Data:**
  - Normal mode: 0x0
  - Loopback merge mode: 0x1
  - Loopback replace mode: 0x2

<14> Section 3.2.1.14: In Windows, an administrator may configure the Configured Computer Base Frequency by setting the base frequency value (in minutes) in the computer-specific Registry Policy file in the following location. If a value of 0 is configured, Windows ignores it and uses 7 seconds as the base frequency value.

**Key:** Software\Policies\Microsoft\Windows\System

**Value:** GroupPolicyRefreshTimeDC (for computers that are domain controllers)

GroupPolicyRefreshTime (for computers that are not domain controllers)

**Type:** REG_WORD

**Size:** 4

**Data:** A number in the range 0 – 64800 (decimal).

<15> Section 3.2.1.15: In Windows, an administrator may configure Configured Computer Random Offset by setting the offset value (in minutes) in the computer-specific Registry Policy file in the following location.

**Key:** Software\Policies\Microsoft\Windows\System

**Value:** GroupPolicyRefreshTimeOffsetDC (for computers that are domain controllers)

GroupPolicyRefreshTimeOffset (for computers that are not domain controllers)

**Type:** REG_WORD

**Size:** 4

**Data:** A number in the range 0 – 1440 (decimal).

<16> Section 3.2.1.18: In Windows, an administrator may configure the Configured User Base Frequency by setting the base frequency value (in minutes) in the user-specific Registry Policy file in the following location. If a value of 0 is configured, Windows ignores it and uses 7 seconds as the base frequency value.

**Key:** Software\Policies\Microsoft\Windows\System

**Value:** GroupPolicyRefreshTime

**Type:** REG_WORD

**Size:** 4

**Data:** A number in the range 0 – 64800 (decimal).

<17> Section 3.2.1.19: In Windows, an administrator may configure the Configured User Random Offset by setting the offset value (in minutes) in the user-specific Registry Policy file in the following location.

**Key:** Software\Policies\Microsoft\Windows\System

**Value:** GroupPolicyRefreshTimeOffset

**Type:** REG_WORD

**Size:** 4

**Data:** A number in the range 0 – 1440 (decimal).

<18> Section 3.2.1.21: In Windows, periodic refresh of group policy is enabled by default. An administrator may modify the default behavior by configuring Configured Disable Periodic Refresh in the computer-specific Registry Policy file in the following location.

**Key:** Software\Microsoft\Windows\CurrentVersion\Policies\System

**Value:** DisableBkGndGroupPolicy

**Type:** REG_WORD

**Size:** 4

**Data:**

- Disable periodic refresh: 0x1
- Enable periodic refresh: 0x0

<19> Section 3.2.5.1: When policy application is terminated, Windows clients log an event to a Windows Event Log.

<20> Section 3.2.5.1.1: When policy application is terminated, Windows clients log an event to a Windows Event Log.

<21> Section 3.2.5.1.1: When policy application is terminated, Windows clients log an event to a Windows Event Log.

<22> Section 3.2.5.1.1: When policy application is terminated, Windows clients log an event to a Windows Event Log.

<23> Section 3.2.5.1.1: When policy application is terminated, Windows clients log an event to a Windows Event Log.

<24> Section 3.2.5.1.1: When policy application is terminated, Windows clients log an event to a Windows Event Log.

<25> Section 3.2.5.1.2: When policy application is terminated, Windows clients log an event to a Windows Event Log.

<26> Section 3.2.5.1.3: When policy application is terminated, Windows clients log an event to a Windows Event Log.

<27> Section 3.2.5.1.4: When policy application is terminated, Windows clients log an event to a Windows Event Log.

<28> Section 3.2.5.1.5: When policy application is terminated, Windows clients log an event to a Windows Event Log.

<29> Section 3.2.5.1.5: When policy application is terminated, Windows clients log an event to a Windows Event Log.

<30> Section 3.2.5.1.5: When policy application is terminated, Windows clients log an event to a Windows Event Log.

<31> Section 3.2.5.1.7: When policy application is terminated, Windows clients log an event to a Windows Event Log.

<32> Section 3.2.5.1.10: By default, Windows Clients (versions Windows 2000, Windows XP, and Windows Server 2003) do not invoke the Software Installation, as specified in [MS-GPSI], and Folder Redirection, as specified in [MS-GPFR], if the link speed is less than 500 kilobytes per second. An administrator can use Group Policy to modify the threshold speed and the set of Group Policy Extensions to be skipped.

<33> Section 3.2.5.2: In Windows, the Local Group Policy Object is stored in the local file system under <Root-Windows-Directory>\System32\GroupPolicy (for example, C:\Windows\System32\GroupPolicy). Once created, the Local Group Policy Object persists until deleted.

<34> Section 3.2.7: Windows-based Clients determine the FQDN of a user account by calling the **GetUserNameEx** method with the following parameters:

- The decimal value 12 for *NameFormat*.
- A pointer to the output buffer for *lpNameBuffer*.
- The size of the output buffer.

Upon success, the method returns a string in the output buffer, whose format is "<FQDN>\<User Name>". The string is parsed to obtain the FQDN.

<35> Section 3.2.7: In Windows, clients invoke policy application when a computer regains network connectivity to a Group Policy Server after a prior policy application failure due to the lack of network connectivity to a Group Policy Server.

This information is applicable to the following product versions of Windows:

- Windows Vista
- Windows Server 2008
- Windows 7
- Windows Server 2008 R2

<36> Section 3.3.5.1: Windows uses the SetNamedSecurityInfo Win32 API.

<37> Section 3.3.5.3: Windows uses the SetNamedSecurityInfo Win32 API.

<38> Section 4.6: Windows uses the SetNamedSecurityInfo Win32 API.

<39> Section 4.8: Windows uses the SetNamedSecurityInfo Win32 API.

# 7 Change Tracking

This section identifies changes that were made to the [MS-GPOL] protocol document between the January 2011 and February 2011 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

The revision class **New** means that a new document is being released.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.
- An extensive rewrite, addition, or deletion of major portions of content.
- The removal of a document from the documentation set.
- Changes made for template compliance.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **Editorial** means that the language and formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical or language changes were introduced. The technical content of the document is identical to the last released version, but minor editorial and formatting changes, as well as updates to the header and footer information, and to the revision summary, may have been made.

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

- New content added.
- Content updated.
- Content removed.
- New product behavior note added.
- Product behavior note updated.
- Product behavior note removed.
- New protocol syntax added.
- Protocol syntax updated.
- Protocol syntax removed.
- New content added due to protocol revision.
- Content updated due to protocol revision.
- Content removed due to protocol revision.
- New protocol syntax added due to protocol revision.

- Protocol syntax updated due to protocol revision.
- Protocol syntax removed due to protocol revision.
- New content added for template compliance.
- Content updated for template compliance.
- Content removed for template compliance.
- Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated.**

Some important terms used in the change type descriptions are defined as follows:

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.
- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

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

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| 1.1 Glossary | 56633<br>Updated the description of the "GPO" term. | N | Content updated. |
| 1.1 Glossary | 58172<br>Added "domain naming context (domain NC)" to the list of terms defined in [MS-GLOS]. | Y | Content updated. |
| 2.2.2 Domain SOM Search | 56634<br>Updated flag value meanings. | N | Content updated. |
| 2.2.2 Domain SOM Search | 63074<br>Updated the gpOptions attribute format description. | N | Content updated. |
| 2.2.2 Domain SOM Search | 63141<br>Updated the gpLink attribute format description. | N | Content updated. |
| 2.2.4 GPO Search | 62043<br>Updated the gPCMachineExtensionNames attribute format description. | Y | Content updated. |
| 2.2.8.1 GPO Creation Message | 56911<br>Updated LDAP messages. | N | Content updated. |
| 2.2.8.1 GPO Creation | 56911<br>Specified that an administrative tool MUST generate the | Y | Content updated. |

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| Message | GUID portion of the new GPO DN by using a specific GUID-generation algorithm. Clarified how containers and GPO existence MUST be checked. | | |
| 2.2.8.1 GPO Creation Message | 58172<br>Updated content by changing "root domain naming context" to "domain naming context". | Y | Content updated. |
| 2.2.8.1 GPO Creation Message | 56922<br>Removed unnecessary LDAP Search request involving retrieval of "Policies" container. | N | Content updated. |
| 2.2.8.1 GPO Creation Message | 56911<br>Specified section for [C706] reference. | N | Content updated. |
| 2.2.8.2 GPO Extension Update Message | 58446<br>Clarified that the update of gPCUserExtensionNames or gPCMachineExtensionNames occurs only when a GUID is not present from a prior update. | Y | Content updated. |
| 2.2.8.2 GPO Extension Update Message | 60640<br>Specified the result of the modifyRequest message. | Y | Content updated. |
| 2.2.8.3 GPO Property Update Message | 60640<br>Specified the result of the modifyRequest message. | Y | Content updated. |
| 2.2.8.4 SOM Property Update Message | 60640<br>Specified the result of the modifyRequest message. | Y | Content updated. |
| 3.2.1.1 Cache of GPO Versions | 63178<br>Revised section title. Updated the Cache of GPO Versions description. | Y | Content updated. |
| 3.2.1.3 Policy Source Mode | 58426<br>Changed "GPO list" to "Filtered GPO List (Public)". | N | Content updated. |
| 3.2.1.3 Policy Source Mode | 63048<br>Updated the Loopback replace mode and Loopback merge mode descriptions. | N | Content updated. |
| 3.2.1.4 GPO List | 62790<br>Changed "GUID" to "curly braced GUID string" in the GPO GUID description | N | Content updated. |
| 3.2.1.4 GPO List | 62803<br>Added lower and upper bit information to the GPO versions description. | N | Content updated. |
| 3.2.1.4 GPO List | 62815<br>Removed the GPO DN entry and added that the Scoped GPO DN is a Unicode string with a prefix. | N | Content updated. |

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| 3.2.1.4 GPO List | 62862<br>Removed the GPO path entry and added that the Scoped GPO path is a Unicode string. | N | Content updated. |
| 3.2.1.4 GPO List | 62865<br>Added that the FunctionalityVersion is an integer. | N | Content updated. |
| 3.2.1.4 GPO List | 62917<br>Added that the WMI Filter is a Unicode string. | N | Content updated. |
| 3.2.1.5 Filtered GPO List (Public) | 58426<br>Changed "Filtered GPO list" to "Filtered GPO List (Public)". | N | Content updated. |
| 3.2.1.5 Filtered GPO List (Public) | 63137<br>Updated content with respect to validation of GPOs in the list. | N | Content updated. |
| 3.2.1.6 SOM List | 62432<br>Added SOM Object type to the list of information that each SOM must maintain. | N | Content updated. |
| 3.2.1.7 SOM GPLink List | 59125<br>Updated list. | N | Content updated. |
| 3.2.4 Higher-Layer Triggered Events | 56039<br>Specified information that must be passed by GPOL to a GP Extension plug-in. | Y | Content updated. |
| 3.2.4 Higher-Layer Triggered Events | 59422<br>Revised the logical parameters list. | Y | Content updated. |
| 3.2.5.1 Policy Application | 58432<br>Updated formatting and clarified that it is the Filtered GPO list that is ordered. | Y | Content updated. |
| 3.2.5.1 Policy Application | 58432<br>Moved description of Filtered GPO List ordering. | N | Content updated. |
| 3.2.5.1.1 DC Discovery and AD Connection Establishment | 57281<br>Added a step specifying Kerberos authentication for user policy mode. | Y | Content updated. |
| 3.2.5.1.2 DN Discovery | 62420<br>Replaced "the DN" with "the policy target DN". | Y | Content updated. |
| 3.2.5.1.3 Domain SOM Search | 56667<br>Clarified the searching, filtering and prioritizing of SOMs to be added to the SOM list. Updated the processing rules for the gpLink and gpOptions attributes. | Y | Content updated. |
| 3.2.5.1.3 Domain SOM | 58337<br>Added an enumeration for the SOM attribute. | Y | Content updated. |

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| Search | | | |
| 3.2.5.1.3 Domain SOM Search | 58337<br>Updated the description for disabling LDAP_OPT_REFERRALS. | N | Content updated. |
| 3.2.5.1.3 Domain SOM Search | 62420<br>Replaced "the DN" with "the policy target DN" in two places. | Y | Content updated. |
| 3.2.5.1.3 Domain SOM Search | 62611<br>Updated SOM search failure processing rules. | Y | Content updated. |
| 3.2.5.1.4 Site Search | 62639<br>Updated the processing rules for the retrieved gpLink attribute. | Y | Content updated. |
| 3.2.5.1.4 Site Search | 61784<br>Revised description of the processing sequence when the method returns ERROR_NO_SITENAME. | Y | Content updated. |
| 3.2.5.1.5 GPO Search | 56667<br>Added an initial step setting the Allow-Enforced-GPOs-Only to FALSE. Updated the processing rules for retrieving an SOM in the SOM List. | Y | Content updated. |
| 3.2.5.1.5 GPO Search | 57300<br>Clarified when the LDAP searchResponse and LDAP handle used for the SearchRequest may be cached. Updated the processing rules for determining the response buffer. | N | Content updated. |
| 3.2.5.1.5 GPO Search | 62743<br>Updated the description of the GPLinkOptions bit field in the steps that describe acting on each SOM in SOM list. | N | Content updated. |
| 3.2.5.1.5 GPO Search | 62686<br>Replaced three occurrences of "string" with "directory string". | N | Content updated. |
| 3.2.5.1.5 GPO Search | 62705<br>Revised the gpOptions value processing rules. | Y | Content updated. |
| 3.2.5.1.5 GPO Search | 63142<br>Updated the GPO list loop processing instructions. | Y | Content updated. |
| 3.2.5.1.5 GPO Search | 63783<br>Switched the construction order of steps 2.2.1 with 2.3.1. | N | Content updated. |
| 3.2.5.1.5 GPO Search | 63863<br>Switched steps 3 (Processing an Enforced GPLink list) and 4 (processing a Non-enforced GPLink list) so that Processing a Non-enforced GPLink list now comes before Processing an Enforced GPLink list, | Y | Content updated. |

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| 3.2.5.1.5 GPO Search | 62990<br>Clarified the way in which an element is added to the GPO list. | N | Content updated. |
| 3.2.5.1.10 Extension Protocol Sequences | 56039<br>Clarified how applicablity to the policy target is determined, by referring to the Filtered GPO List (Public) section. | Y | Content updated. |
| 3.2.5.1.10 Extension Protocol Sequences | 58231<br>Clarified the use of the Filtered GPO List. | N | Content updated. |
| 3.2.5.1.10 Extension Protocol Sequences | 61189<br>Specified the order in which Group Policy Extensions are invoked and executed. | Y | Content updated. |
| 3.2.5.2 GPO Processing Order | 58432<br>Added new section. | Y | Content updated. |
| 3.3.4 Higher-Layer Triggered Events | 58239<br>Removed duplicate text. | N | Content updated. |
| 3.3.4.1 Group Policy Creation | 5823<br>Added section. | Y | New content added. |
| 3.3.4.1 Group Policy Creation | 58172<br>Updated content to define the parameters used in Group Policy Creation. | Y | Content updated. |
| 3.3.4.2 Group Policy Property Update | 58239<br>Added section. | Y | New content added. |
| 3.3.4.3 The SOM Property Update | 58239<br>Added section. | Y | New content added. |
| 3.3.4.3 The SOM Property Update | 58173<br>Updated content regarding the parameters for the SOM Property Update event. | Y | Content updated. |
| 3.3.4.4 Group Policy Extension Update | 58239<br>Added section. | Y | New content added. |
| 3.3.4.4 Group Policy Extension Update | 58174<br>Updated section name, and added table of parameters. | N | Content updated. |
| 3.3.4.4 Group Policy | 58174<br>Specified how the GPO attribute "versionNumber" is | Y | Content updated. |

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| Extension Update | incremented. Added the "Is User Policy" parameter and updated the descriptions of the "CSE GUID" and "TOOL GUID" parameters. | | |
| 3.3.4.6 Group Policy Deletion | 58041<br>Added section. | N | Content updated. |
| 3.3.5.1 GPO Creation | 56911<br>Added message processing rules. | N | Content updated. |
| 3.3.5.1 GPO Creation | 56922<br>Updated topic information with respect to GPO creation. | N | Content updated. |
| 3.3.5.1 GPO Creation | 56938<br>Added GPO Creation message. | N | Content updated. |
| 3.3.5.1 GPO Creation | 60036<br>Specified that the result of SearchRequest in a GPO Creation Message is a reply from the GP server with one LDAP searchResponse message that contains one searchResultEntry. | N | Content updated. |
| 3.3.5.1 GPO Creation | 56935<br>Updated the procedure for creating container objects. | N | Content updated. |
| 3.3.5.1 GPO Creation | 56922<br>Removed unnecessary retrieval request involving "Policies" container. | N | Content updated. |
| 3.3.5.1 GPO Creation | 57709<br>Moved the paragraph about an addResponse message into the last item in the list of steps involved in creating a groupPolicyContainer object. | N | Content updated. |
| 3.3.5.1 GPO Creation | 59932<br>Specified how the security descriptor is modified on the directory. | Y | Content updated. |
| 3.3.5.1 GPO Creation | 60792<br>Specified conditions under which the protocol sequence is terminated. | Y | Content updated. |
| 3.3.5.2 GPO Extension Update | 60460<br>Removed the description of the modifyRequest message result. | N | Content updated. |
| 3.3.5.3 GPO Property Update | 60460<br>Removed the description of the modifyRequest message result. | N | Content updated. |
| 3.3.5.4 GPO File System Version Update | 57633<br>Added step to increment the version value. | Y | Content updated. |

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| 3.3.5.5 SOM Property Update | 60460<br>Removed the description of the modifyRequest message result. | N | Content updated. |
| 3.3.5.6 GPO Deletion | 57709<br>Replaced "Delete every file and subdirectory in GPO path. Any failures from file operations should be logged" with a procedure for deleting files and subdirectories. | N | Content updated. |
| 3.3.5.6 GPO Deletion | 60793<br>Specified the delResponse message processing rule. | Y | Content updated. |
| 3.3.5.7 GPO Link Creation, Update and Deletion | 57982<br>Added section. | Y | Content updated. |

# 8 Index