

Supporting Group Policy

Module 02: How Group Policy Works



V1.0 Released: February 17, 2014

# Core Group Policy Physical Structure

Understanding where GPOs are stored and how they are structured can help you troubleshoot problems you might encounter when you implement Group Policy. Although GPOs can be linked to sites, domains, and OUs, they are stored only in the domain. As explained earlier, a GPO is a virtual object that stores its data in two locations: a Group Policy container and a Group Policy template.

## Group Policy Container

A Group Policy container is a location in Active Directory that stores GPOs and their properties. The properties of a GPO include both computer and user Group Policy information. The Policies container is the default location of GPOs. The path to the Policies container, in Lightweight Directory Access Protocol (LDAP) syntax, is **CN=Policies,CN=System,DC=Domain\_Name**, where the Domain\_Name values specify a fully qualified domain name (FQDN).

The Active Directory store contains the Group Policy container of each GPO in the domain. The Group Policy container contains attributes that are used to deploy GPOs to the domain, to OUs, and to sites within the domain. The Group Policy container also contains a link to the file system component of a GPO — the Group Policy template. Some of the information in a Group Policy container includes:

* **Version information.** Ensures that the information is synchronized with the Group Policy template information.
* **Status information.** Indicates whether the user or computer portion of the GPO is enabled or disabled.
* **List of components.** Lists (extensions) that have settings in the GPO. These attributes are **gPCMachineExtensionNames** and **gPCUserExtensionNames**.
* **File system path.** Specifies the Universal Naming Convention (UNC) path to the Sysvol folder. This attribute is **gPCFileSysPath**.
* **Functionality version.** Gives the version of the tool that created the GPO. Currently, this is version 2. This attribute is **gPCFunctionalityVersion**.
* **WMI filter.** Contains the distinguished name of the WMI filter. This attribute is **gPCWQLFilter**.

## System Container

Each Windows Server 2003 domain contains a System container. The System container stores per-domain configuration settings, including GPO property settings, Group Policy container settings, IP Security settings, and WMI policy settings. IP Security and WMI policy are deployed to client computers through the GPO infrastructure.

The following subcontainers of the System container hold GPO-related settings:

* **Policies.** This object contains groupPolicyContainer objects listed by their unique name. Each groupPolicyContainer object holds subcontainers for selected computer and user policy settings.
* **Domain, OUs and Sites.** These objects contain two GPO property settings, **gPLink** and **gPOptions**.
* **Default Domain Policy.** This object contains the **AppCategories** container, which is part of the Group Policy Software installation extension.
* **IP Security.** This object contains IP Security policy settings that are linked to a GPO. The linked IP Security policy is applied to the recipients (user or computer) of the GPO.
* **WMIPolicy.** This object contains WMI filters that can be applied to GPOs. WMI filters contain one or more Windows Query Language (WQL) statements.

## System\Policies Container

The System container is a top level container found in each domain naming context. It is normally hidden from view in the Active Directory Users and Computers snap-in but can be made visible by selecting “Advanced Features” from the snap-in View menu inside MMC. (Objects appear hidden in the Active Directory Users and Computers snap-in when they have the property showInAdvancedViewOnly = TRUE.) Group Policy information is stored in the Policies subcontainer of this container. Each GPO is identified by a GroupPolicyContainer object stored within the Policies container.

The Group Policy container is located in the **Domain\_Name/System/Policies** container. Each Group Policy container is given a common name (CN) and this name is also assigned as the container name. For example, the name attribute of a Group Policy container, might be: {923B9E2F-9757-4DCF-B88A-1136720B5AF2}, which is also assigned to the Group Policy container's CN attribute.

The default GPOs are assigned the same Group Policy container CN on all domains. All other GPOs are assigned a unique CN. The default GPOs and their Group Policy container common names are:

* Default Domain Policy: {31B2F340-016D-11D2-945F-00C04FB984F9}.
* Default Domain Controllers Policy: {6AC1786C-016F-11D2-945F-00C04fB984F9}.

Knowing the common names of the default GPOs will help you distinguish them from non-default GPOs.

# How a Group Policy Container is Named

Group Policy containers are named automatically when they are created. The CN of each Group Policy container is a GUID (Globally Unique Identifier). This is distinct from and unrelated to the Object GUID given to each Active Directory object. The CN is the name of the Group Policy container used to ensure uniqueness of Group Policy container names within the Policies container. There is no requirement for these GUIDs to be unique between domains (the Default Domain Policy and the Default Domain Controllers Policy GPOs each have identical GUIDs in all Active Directory installations). However, an Object GUID is always unique across all installations of the Active Directory store.

The following table shows permissions on Group Policy container:

**Default Group Policy Container Permissions**

|  |  |
| --- | --- |
| **Trustee** | **Access** |
| Authenticated Users | Read, Apply Group Policy |
| Domain Admins | Read, Write |
| System | Read, Write |

## GPO Attributes in the Policies CN

GPOs are created by instantiating the groupPolicyContainer class in the Active Directory schema and storing the resulting GPO in the System/Policies container of the Active Directory store. After creating a GPO, you can review its CN from the Active Directory Users and Computers snap-in by enabling the Advanced view and then expanding the Policies CN. You can review all GPO attributes and their values from the Active Directory Services Interface Editor snap-in, ADSI Edit.

Object attributes are either mandatory or optional, as defined in the Active Directory schema. The CN attribute is mandatory for the class **Container**, Group Policy container's parent class. Three attributes — instanceType, objectCategory, and objectClass — are mandatory for the class **Top**, CN's parent class. Thus the Group Policy container class inherits all four mandatory attributes. The following table describes the mandatory attributes:

**Mandatory Attributes of the groupPolicyContainer Class**

|  |  |
| --- | --- |
| **Name** | **Description** |
| CN | The common name of the GPO. This is in the form of a GUID to avoid GPO naming conflicts within the Policies container. |
| instanceType | An attribute that dictates how an object is instantiated from its class on a particular server. In this case, it describes how the groupPolicyContainer class is created into a GPO in the Active Directory. A GPO is assigned the instanceType value of 4. |
| objectCategory | An object class name, including the object's path, used to group object's of the instantiated class. For example, the objectCategory of a GPO in the contoso.com domain is: CN=Group-Policy-Container,CN=Schema,CN=Configuration,DC=contoso,DC=com. |
| objectClass | The list of classes from which this class is derived. For a GPO, the objectClass is Container, groupPolicyContainer, and top. |

There are also a number of optional attributes inherited from the top class, and others that are assigned directly to the Group Policy container. Many optional attributes are required in order for the Group Policy container to function properly. For example, the GPCFileSysPath optional attribute must be present or the Group Policy container will not be linked to its corresponding Group Policy template.

## GroupPolicyContainer Subcontainers

Within the GroupPolicyContainer there are a series of subcontainers. The first level of subcontainers — User and Machine — belong to the class Container. These two containers are used to separate some User-specific and Computer-specific Group Policy components.

## Group Policy Container-Related Attributes of Domain, Site, and OU Containers

Windows Server 2003 uses domain, DNS, site, and organizational unit classes to create domain, site, and OU container objects respectively. These objects contain two optional Group Policy container-related attributes, **gPLink** and **gPOptions**. The **gPLink** property contains the prioritized list of GPOs and the **gPOptions** property contains the Block Policy Inheritance setting.

The **gPLink** attribute holds a list of all Group Policy containers linked to the container and a number for each listed Group Policy container, that represents the **Enforced** (previously known as No Override) and **Disabled** option settings. The list appears in priority order from lowest to highest priority GPO.

The **gPOptions** attribute holds an integer value that indicates whether the Block Policy Inheritance option of a domain or OU is enabled (1) or disabled (0).

## Managing Group Policy Links for a Site, Domain, or OU

To manage GPO links to a site, domain, or OU, you must have read and write access to the **gPLink** and **gPOptions** properties. By default, Domain Admins have this permission for domains and organizational unit, and only Enterprise Admins and Domain Admins of the forest root domain can manage links to sites. Active Directory supports security settings on a per-property basis. This means that a non-administrator can be delegated read and write access to specific properties. In this case, if non-administrators have read and write access to the gPLink and gPOptions properties, they can manage the list of GPOs linked to that site, domain, or OU.

## How WMIPolicy Objects are Stored and Associated with Group Policy Container Objects

A single WMI filter can be assigned to a Group Policy container. The Group Policy container stores the distinguished name of the filter in **gPCWQLFilter** attribute. The Group Policy container locates the assigned filter in the System/WMIPolicy/SOM container. Each Windows Server 2003 domain stores its WMI filters in this Active Directory container. Each WMI filter stored in the SOM container lists the rules that define the WMI filter. Each rule is listed separately. For example, consider a WMI filter containing the following three WQL queries:

SELECT \* FROM Win32\_Product WHERE IdentifyingNumber = "{5E076CF2-EFED-43A2-A623-13E0D62EC7E0}"

SELECT \* FROM Win32\_Product WHERE IdentifyingNumber = "{242365CD-80F2-11D2-989A-00C04F7978A9}"

SELECT \* FROM Win32\_Product WHERE IdentifyingNumber = "{00000409-78E1-11D2-B60F-006097C998E7}"

Three WMI rules are defined in the details of the filter. Each rule contains a number of attributes, including the query language (WQL) and the WMI namespace queried by the rule.

## Group Policy Template

The majority of Group Policy settings are stored in the file system of the domain controllers. This part of each GPO is known as the Group Policy template. The GroupPolicyContainer object for each GPO has a property, GPCFileSysPath, which contains the UNC path to its related Group Policy template.

All Group Policy templates in a domain are stored in the **\\*domain\_name*\Sysvol\*domain\_name*\Policies** folder, where domain\_name is the FQDN of the domain. The Group Policy template for the most part stores the actual data for the policy extensions, for example Security Settings inf file, Administrative Template-based policy settings .adm and .pol files, applications available for the Group Policy Software installation extension, and potentially scripts.

## The Gpt.ini File

The Gpt.ini file is located at the root of each Group Policy template. Each Gpt.ini file contains GPO version information. Except for the Gpt.ini files created for the default GPOs, a display name value is also written to the file.

Each Gpt.ini file contains the GPO version number of the Group Policy template.

[General]

Version=65539

Normally, this is identical to the version-number property of the corresponding GroupPolicyContainer object. It is encoded in the same way — as a decimal representation of a 4 byte hexadecimal number, the upper two bytes of which contain the GPO user settings version and the lower two bytes contain the computer settings version. In this example the version is equal to 10003 hexadecimal giving a user settings version of 1 and a computer settings version of 3.

Storing this version number in the Gpt.ini allows the CSEs to check if the client is out of date to the last processing of policy settings or if the currently applied policy settings (cached policies) are up-to-date. If the cached version is different from the version in the Group Policy template or Group Policy container, then policy settings will be reprocessed.

## Group Policy Template Subfolders

The Group Policy template folder contains the following subfolders:

* **Machine.** Includes a Registry.pol file that contains the registry settings to be applied to computers. When a computer initializes, this Registry.pol file is downloaded and applied to the **HKEY\_LOCAL\_MACHINE** portion of the registry. The **Machine** folder can contain the following subfolders (depending on the contents of the GPO):
  + **Scripts\Startup.** Contains the scripts that are to run when the computer starts up.
  + **Scripts\Shutdown.** Contains the scripts that are to run when the computer shuts down.
  + **Applications.** Contains the advertisement files (.aas files) used by the Windows installer. These are applied to computers.
  + **Microsoft\Windows NT\Secedit.** Contains the Gpttmpl.inf file, which includes the default security configuration settings for a Windows Server 2003 domain controller.
  + **Adm.** Contains all of the .adm files for the GPO.
* User. Includes a Registry.pol file that contains the registry settings to be applied to users. When a user logs on to a computer, this Registry.pol file is downloaded and applied to the HKEY\_CURRENT\_USER portion of the registry. The User folder can contain the following subfolders (depending on the contents of the GPO):
  + **Applications.** Contains the advertisement files (.aas files) used by the Windows installer. These are applied to users.
  + **Documents and Settings.** Contains the Fdeploy.ini file, which includes status information about the Folder Redirection options for the current user's special folders.
  + **Microsoft\RemoteInstall.** Contains the OSCfilter.ini file, which holds user options for operating system installation through Remote Installation Services.
  + **Microsoft\IEAK.** Contains settings for the Internet Explorer Maintenance snap-in.
  + **Scripts\Logon.** Contains all the user logon scripts and related files for this GPO.
  + **Scripts\Logoff.** Contains all the user logoff scripts and related files for this GPO.

The User and Machine folders are created at install time, and the other folders are created as needed when policy is set.

The permissions of each Group Policy template reflect the read and write permissions applied to the GroupPolicyContainer through the Group Policy Object Editor. These permissions are automatically maintained and are shown in the following table.

**Default Group Policy Template Permissions**

|  |  |
| --- | --- |
| **Trustee** | **Access** |
| Authenticated Users | Read and Execute |
| Administrators | Full Control |
| Group Policy Creator Owners | Read and Execute |
| Creator Owner | Full Control (Subfolders and Files only) |
| System | Full Control |

## Group Policy Object Editor use of Sysvol

Each policy setting changed in a GPO causes at least two files to be rewritten — the GPT.ini and the file holding the changed setting. Making many changes to a GPO can cause a lot of network traffic as Sysvol replicates these changes. This congestion should only occur on a local area network where Sysvol replication occurs frequently. Across wide area network links, the inter-site replication schedule will cause these changes to be amalgamated into a smaller amount of traffic (for example, four changes to the Registry.pol file will result in only a single file replication).

## The Local Group Policy Object

The Local GPO has no Active Directory component. Information stored in the Group Policy container of an Active Directory GPO is instead stored in the Group Policy template of a Local GPO. The Group Policy template of a Local GPO is located in the **Windows\system32\GroupPolicy** folder. The Gpt.ini file in this GroupPolicy folder must hold more management information than its counterpart in a domain-based GPO because there is no Active Directory component to hold this information. The following table shows the attributes for the Group Policy template.ini file.

**Local GPO GPT.INI Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| gPCUserExtensionNames | Includes a list of GUIDs that tells the client-side engine which CSEs have User data in the GPO.  The format is: [{GUID of CSE}{GUID of MMC extension}{GUID of second MMC extension if appropriate}][repeat first section as appropriate]. |
| GPCMachineExtensionNames | Includes a list of GUIDs that tells the client-side engine which CSEs have computer data in the GPO. |
| Options | Refers to GPO options such as User portion disabled or Computer portion disabled. |

The following extensions are disabled in a Local GPO:

* Group Policy Software installation extension
* Folder Redirection

The following extensions have reduced functionality in a Local GPO:

* Public Key policies; EFS only; there are not any options for trust lists or auto enrollment.
* Security Settings; there are not any options for Restricted Groups or File System, Registry or Service Access Control Lists (ACLs).

# Core Group Policy Processes and Interactions

Application of GPOs to targeted users and computers relies on many interactive processes. This section explains how GPOs are applied and filtered to Active Directory containers such as sites, domains, and OUs. It includes information about how the Group Policy engine processes GPOs in conjunction with CSEs. In addition, it explains how Group Policy is replicated among domain controllers.

## Group Policy Processing Rules

GPOs that apply to a user or computer do not all have the same precedence. Settings that are applied later can override settings that are applied earlier. Group Policy settings are processed in the following order:

* **Local Group Policy object.** Each computer has exactly one Group Policy object that is stored locally. This processes for both computer and user Group Policy processing.
* **Site.** Any GPOs that have been linked to the site that the computer belongs to are processed next. Processing is in the order that is specified by the administrator, on the Linked Group Policy Objects tab for the site in GPMC. The GPO with the lowest link order(最小的ORDER) is processed last, and therefore has the highest precedence.
* **Domain.** Processing of multiple domain-linked GPOs is in the order specified by the administrator, on the Linked Group Policy Objects tab for the domain in GPMC. The GPO with the lowest link order is processed last, and therefore has the highest precedence.
* **Organizational units.** GPOs that are linked to the organizational unit that is highest in the Active Directory hierarchy are processed first, then GPOs that are linked to its child organizational unit are processed, and so on. Finally, the GPOs that are linked to the organizational unit that contains the user or computer are processed.

To summarize, the Local GPO is processed first, and the organizational unit to which the computer or user belongs (the one that it is a direct member of) is processed last. All of this processing is subject to the following conditions:

* WMI or security filtering that has been applied to GPOs.
* Any domain-based GPO (not Local GPO) can be enforced by using the **Enforce** option so that its policies cannot be overwritten. Because an Enforced GPO is processed last, no other settings can write over the settings in that GPO. If you have more than one Enforced GPO, it's possible to set the same setting in each GPO to a different value, in which case, the link order of the GPOs determines which one contains the final settings.
* At any domain or organizational unit, Group Policy inheritance can be selectively designated as **Block Inheritance.** However, because enforced GPOs are always applied, and cannot be blocked, blocking inheritance does not prevent policy from **Enforced** GPOs from applying.

Every computer has a single Local GPO that is always processed regardless of whether the computer is part of a domain or is a stand-alone computer. The Local GPO can't be blocked by domain-based GPOs. However, settings in domain GPOs always take precedence since they are processed after the Local GPO.

## Targeting GPOs

The site, domain, and OU links from a GPO are used as the primary targeting principle for defining which computers and users should receive a GPO. Security filtering and WMI filtering can be used to further reduce the set of computers and users to which the GPO will apply. The Group Policy engine uses the following logic in processing GPOs: If a GPO is linked to a domain, site, or OU that applies to the user or computer, the Group Policy engine must then determine whether the GPO should be added to its GPO list for processing. A GPO is blocked from processing in the following circumstances:

* **The GPO is disabled.** You disable either or both the computer or user components of a GPO from its Policy Properties dialog box.
* **The computer or user does not have permission to read and apply the GPO.** You control permission to a GPO through security filtering, as explained in the following section.
* **A WMI filter applied to a GPO evaluates to false on the client computer.** A WMI filter must evaluate to true before the Group Policy engine will allow it to be processed, as explained in the following section.

## Security Filtering

Security filtering is a way of refining which users and computers will receive and apply the settings in a GPO. By using security filtering to specify that only certain security principals within a container where the GPO is linked apply the GPO, you can narrow the scope of a GPO so that it applies only to a single group, user, or computer. Security filtering determines whether the GPO as a whole applies to groups, users, or computers; it cannot be used selectively on different settings within a GPO.

In order for the GPO to apply to a given user or computer, that user or computer must have both **Read** and **Apply Group Policy (AGP)** permissions on the GPO, either explicitly, or effectively though group membership.

By default, all GPOs have **Read** and **AGP** both **Allowed** for the Authenticated Users group. The Authenticated Users group includes both users and computers. This is how all authenticated users receive the settings of a new GPO when it is applied to an organizational unit, domain or site. Therefore, the default behavior is for every GPO to apply to every Authenticated User. By default, Domain Admins, Enterprise Admins, and the local system have full control permissions, without the Apply Group Policy access-control entry (ACE). However, administrators are members of Authenticated Users, which means that they will receive the settings in the GPO by default.

These permissions can be changed to limit the scope to a specific set of users, groups, or computers within the organizational unit, domain, or site. The Group Policy Management Console manages these permissions as a single unit, and displays the security filtering for the GPO on the GPO Scope tab. In GPMC, groups, users, and computers can be added or removed as security filters for each GPO.

## How Security Filtering is Processed

Before processing a GPO, the Group Policy engine checks the Access Control List ACL associated with the GPO. If an ACE(Access control entry) on a GPO denies a security principal to which the computer or user belongs, either the Apply Group Policy or Read permission, the Group Policy engine does not add the GPO to its list of GPOs to process. Additionally, an ACE on a GPO must allow the appropriate security principal both Apply Group Policy and Read permissions in order for the Group Policy engine to add the GPO to the GPO processing list.

If appropriate permissions are granted to the GPO, it is added to the list of GPOs to download.

In general, **Deny** ACEs should be avoided because you can achieve the same results by granting or not granting **Allow** permissions.

## Windows Management Instrumentation (WMI) Filtering

WMI makes data about a target computer available for administrative use. Such data can include hardware and software inventory, settings, and configuration information. For example, WMI exposes hardware configuration data such as CPU, memory, disk space, and manufacturer, as well as software configuration data from the registry, drivers, file system, Active Directory, the Windows Installer service, networking configuration, and application data.

WMI filtering allows you to filter the application of a GPO by attaching a Windows Query Language query to a GPO. The queries can be written to query WMI for multiple items. If the query returns true for all queried items, then the GPO will be applied to the target user or computer.

When a GPO that is linked to a WMI filter is applied on the target computer, the filter is evaluated on the target computer. If the WMI filter evaluates to false, the GPO is not applied (except if the client computer is running Windows 2000, in which case the filter is ignored and the GPO is always applied). If the WMI filter evaluates to true, the GPO is applied.

The WMI filter is a separate object from the GPO in the directory. A WMI filter must be linked to a GPO in order to apply. Each GPO can have only one WMI filter; however the same WMI filter can be linked to multiple GPOs. WMI filters, like GPOs, are stored only in domains. A WMI filter and the GPO it is linked to must be in the same domain.

## How Windows Management Instrumentation (WMI) Filtering is Processed

If, after security filtering, appropriate permissions are granted to the GPO, it is added to the list of GPOs to download. Upon download, the Group Policy engine reads the **gPCWQLFilter** attribute in the Group Policy container to determine if a WMI filter is applied to the GPO. If so, the WMI filter, which contains one or more WQL statements, is evaluated. If the statement evaluates to true, then the GPO is processed. There are tradeoffs in using WMI filters because they can increase the amount of time it takes to process policy especially if the filter to be evaluated takes a long time to process.

## Windows Management Instrumentation (WMI)Filtering Scenarios

Sample uses of WMI filters include:

* Services. Computers where DHCP is turned on.
* Registry. Computers that have this registry key populated.
* Hardware inventory. Computers with a Pentium III processor.
* Software inventory. Computers with Visual Studio .NET installed.
* Hardware configuration. Computers with network interface cards (NICs) on interrupt level 3.
* Software configuration. Computers with multi-casting turned on.
* Associations. Computers that have any services dependent on Systems Network Architecture (SNA) service.

Client support for WMI filters exists only on Windows XP, Windows Server 2003, and later operating systems. Windows 2000 clients will ignore any WMI filter and the GPO is always applied, regardless of the WMI filter. WMI filters are only available in domains that have at least one Windows Server 2003 domain controller.

## Application of Group Policy

Application of Group Policy involves a series of processes, beginning with user and computer logon.

## Initial Processing of Group Policy

Group Policy for computers is applied at computer startup. For users, Group Policy is applied when they log on. In Windows 2000, the processing of Group Policy is synchronous, which means that computer Group Policy is completed before the logon dialog box is presented, and user Group Policy is completed before the shell is active and available for the user to interact with it. As explained in the following section, Windows XP with Fast Logon-enabled (which is the default setting) allows users to log on while Group Policy is processed in the background.)

## Synchronous and Asynchronous Processing

Synchronous processes can be described as a series of processes where one process must finish running before the next one begins. Asynchronous processes, on the other hand, can run on different threads simultaneously because their outcome is independent of other processes.

You can change the default processing behavior by using a policy setting for each GPO so that processing is asynchronous instead of synchronous. For example, if the policy has been set to remove the **Run** command from the **Start** menu, it is possible under asynchronous processing that a user could logon prior to this policy taking effect, so the user would initially have access to this functionality.

## Fast Logon in Windows XP Professional

By default in Windows XP Professional, the Fast Logon Optimization feature is enabled for both domain and workgroup members. This means that policy settings apply asynchronously when the computer starts and when the user logs on. This process of applying policies is similar to a background refresh process. As a result, users can logon and begin using the Windows shell faster than they would with synchronous processing. Fast Logon Optimization is always off during logon under the following conditions:

* When a user first logs on to a computer.
* When a user has a roaming user profile or a home directory for logon purposes.
* When a user has synchronous logon scripts.

Note that under the preceding conditions, computer startup can still be asynchronous. However, because logon is synchronous under these conditions, logon does not exhibit optimization. The following table compares policy processing of Windows 2000 and Windows XP client computers.

**Default Policy Processing for Client Computers**

|  |  |  |
| --- | --- | --- |
| **Client** | **Application at startup/log on** | **Application at refresh** |
| Windows 2000 | Synchronous | Asynchronous |
| Windows XP Professional | Asynchronous | Asynchronous |

Windows XP clients support Fast Logon Optimization in any domain environment. Fast Logon Optimization can be disabled with the following policy setting:

**Computer Configuration\Administrative Templates\System\Logon\ Always wait for the network at computer startup and logon.**

Note that Fast Logon Optimization is not a feature of Windows Server 2003.

## Folder Redirection and Software Installation Policies

Note that when Fast Logon Optimization is on, a user might need to log on to a computer twice before folder redirection policies and software installation policies are applied. This occurs because the application of these types of policies requires the synchronous policy application. During a policy refresh (which is asynchronous), the system sets a flag indicating that the application of folder redirection or a software installation policy is required. The flag forces synchronous application of the policy at the user's next logon.

## Time Limit for Processing of Group Policy

Under synchronous processing, there is a time limit of 60 minutes for all of Group Policy to finish processing on the client computer. Any CSEs that are not finished after 60 minutes are signaled to stop, in which case the associated policy settings might not be fully applied.

## Background Refresh of Group Policy

In addition to the initial processing of Group Policy at startup and logon, Group Policy is applied subsequently in the background on a periodic basis. During a background refresh, a CSE will only reapply the settings if it detects that a change was made on the server in any of its GPOs or its list of GPOs.

In addition, software installation and folder redirection processing occurs only during computer startup or user logon. This is because background processing could cause undesirable results. For example, in software installation, if an application is no longer assigned, it is removed. If a user is using the application while Group Policy tries to uninstall it or if an assigned application upgrade takes place while someone is using it, errors would occur. Although the Scripts CSE is processed during background refresh, the scripts themselves only run at startup, shutdown, logon, and logoff, as appropriate.

## Periodic Refresh Processing

By default, Group Policy is processed every 90 minutes with a randomized delay of up to 30 minutes — for a total maximum refresh interval of up to 120 minutes.

Group Policy can be configured on a per-extension basis so that a particular extension is always processed during processing of policy even if the GPOs haven't changed. Policy settings for each extension are located in **Computer Configuration\Administrative Templates\System\Group Policy**.

## On-Demand Processing

You can also trigger a background refresh of Group Policy on demand from the client. However, the application of Group Policy cannot be pushed to clients on demand from the server.

## Refreshing Policy from the Command Line

You can update or refresh Group Policy settings manually through a command line tool. On Windows XP and Windows Server 2003, you can use Gpupdate.

## Group Policy and Slow Links

When Group Policy detects a slow link, it sets a flag to indicate to CSEs that a policy setting is being applied across a slow link. Individual CSEs can determine whether or not to apply a policy setting over the slow link. The default settings are as follows:

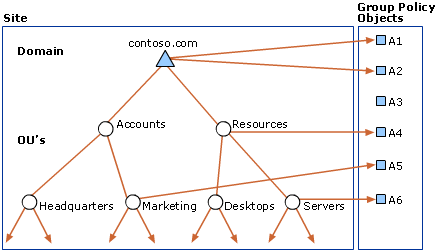
**Default Slow Link Settings**

|  |  |
| --- | --- |
| **Extension** | **Default Setting** |
| Security Settings | On (and cannot be turned off) |
| Administrative Templates | On (and cannot be turned off) |
| Software Installation | Off |
| Scripts | Off |
| Folder Redirection | Off |

## Group Policy Loopback Support

Group Policy is applied to the user or computer, based on where the user or computer object is located in Active Directory. However, in some cases, users might need policy applied to them, based on the location of the computer object, not the location of the user object. The Group Policy loopback feature gives you the ability to apply User Group Policy, based on the computer that the user is logging onto. The following figure shows a sample site, domain, and OU structure and is followed by a description of the changes that can occur with loopback processing.

**Sample Active Directory Structure**



Normal user Group Policy processing specifies that computers located in the Servers organizational unit have the GPOs A3, A1, A2, A4, and A6 applied (in that order) during computer startup. Users of the Marketing organizational unit have GPOs A3, A1, A2, and A5 applied (in that order), regardless of which computer they log on to.

In some cases this processing order might not be what you want. An example is when you do not want applications that have been assigned or published to the users of the Marketing organizational unit to be installed while they are logged on to the computers in the Servers organizational unit. With the Group Policy loopback feature, you can specify two other ways to retrieve the list of GPOs for any user of the computers in the Servers organizational unit:

* Merge mode. In this mode, the computer’s GPOs have higher precedence than the user's GPOs. In this example, the list of GPOs for the computer is A3, A1, A2, A4, and A6, which is added to the user's list of A3, A1, A2, A5, resulting in A3, A1, A2, A5, A3, A1, A2, A4, and 9A6 (listed in lowest to highest priority).
* Replace mode. In this mode, the user's list of GPOs is not gathered. Only the list of GPOs based upon the computer object is used. In this example, the list is A3, A1, A2, A4, and A6.

The loopback feature can be enabled by using the **User Group Policy loopback processing mode** policy under **Computer Settings\Administrative settings\System\Group Policy.**

The processing of the loopback feature is implemented in the Group Policy engine. When the Group Policy engine is about to apply user policy, it looks in the registry for a computer policy, which specifies which mode user policy should be applied in.

# How the Group Policy Engine Processes Client-Side Extensions

Client-side extensions are the components running on the client system that process and apply the Group Policy settings to that system. There are a number of extensions that are pre-installed in Windows Server 2003. Other Microsoft applications and third party application vendors can also write and install additional extensions to implement Group Policy management of these applications.

The default Windows Server 2003 CSEs are listed in the following table:

**Default Windows Server 2003 CSEs**

|  |  |  |
| --- | --- | --- |
| **Client-Side Extension** | **Active Directory Component** | **Sysvol Component** |
| Software Installation | PackageRegistration objects | .aas files |
| Security Settings |  | Gptmpl.inf |
| Folder Redirection |  | fdeploy.ini |
| Scripts |  | Scripts.ini |
| IP Security | IPSec Policy objects |  |
| Internet Explorer Maintenance |  | .ins + branding .inf files. |
| Administrative Templates |  | Registry.pol and .adm files |
| Disk Quota |  | Registry.pol |
| EFS Recovery |  | Registry.pol |
| Remote Installation |  | Oscfilter.ini |
| Wireless Network Policies |  | Registry.pol and .adm files |
| QoS Packet Scheduler |  | Registry.pol and .adm files |

## Client-Side Extension Operation

CSEs are called by the Winlogon process at computer startup, user logon and at the Group Policy refresh interval. CSEs are registered with Winlogon in the registry. This registration information includes a DLL and a DLL entry point (function call) by which the CSE processing can be initiated. The Winlogon process uses these to trigger Group Policy processing.

Each extension can opt not to perform processing at any of these points (for example, avoid processing during background refresh).

## Client-Side Extensions Registered with WinLogon

Each of the CSEs is registered under the following key:

**HKEY\_LOCAL\_MACHINE\Software\Microsoft\WindowsNT\CurrentVersion\Winlogon\GPExtensions**

Each extension is identified by a key named after the GUID of the extension. These extensions are shown in the following table.

**CSE Extensions**

|  |  |
| --- | --- |
| **Extension GUID** | **Extension Name** |
| 25537BA6-77A8-11D2-9B6C-0000F8080861 | Folder Redirection |
| 35378EAC-683F-11D2-A89A-00C04FBBCFA2 | Administrative Templates Extension |
| 3610EDA5-77EF-11D2-8DC5-00C04FA31A66 | Disk Quotas |
| 426031c0-0b47-4852-b0ca-ac3d37bfcb39 | QoS Packet Scheduler |
| 42B5FAAE-6536-11D2-AE5A-0000F87571E3 | Scripts |
| 827D319E-6EAC-11D2-A4EA-00C04F79F83A | Security |
| A2E30F80-D7DE-11d2-BBDE-00C04F86AE3B | Internet Explorer Maintenance |
| B1BE8D72-6EAC-11D2-A4EA-00C04F79F83A | EFS Recovery |
| C6DC5466-785A-11D2-84D0-00C04FB169F7 | Software Installation |
| E437BC1C-AA7D-11D2-A382-00C04F991E27 | IP Security |

## Preferences and Policy Configuration

Manipulating these registry values directly is not recommended. Most of the items in which you might need to change the behavior of an extension (such as forcing a CSE to run over a slow link), are available as Group Policy settings. These can be found in the Group Policy Object Editor in the following location:

**Computer Settings\Administrative Templates\System\Group Policy**

The behavior can be changed for the following CSEs:

* Administrative Templates (Registry-based policy)
* Internet Explorer Maintenance
* Software Installation
* Folder Redirection
* Scripts
* Security
* IP Security
* EFS recovery
* Disk Quotas

## Order of Extension Processing

Administrative Templates policy settings are always processed first. Other extensions are processed in an indeterminate order.

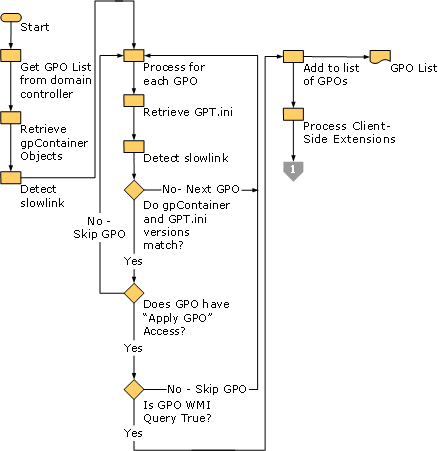
## Policy Application Processes

There are two primary milestones that the Group Policy engine uses for GPO processing:

* Creating the list of GPOs targeted at the user or computer.
* Invoking the relevant CSEs to process the policy settings relevant to them within the GPO list.

The following figure shows the steps required to reach the first milestone in GPO processing, GPO list creation.

**GPO List Creation**

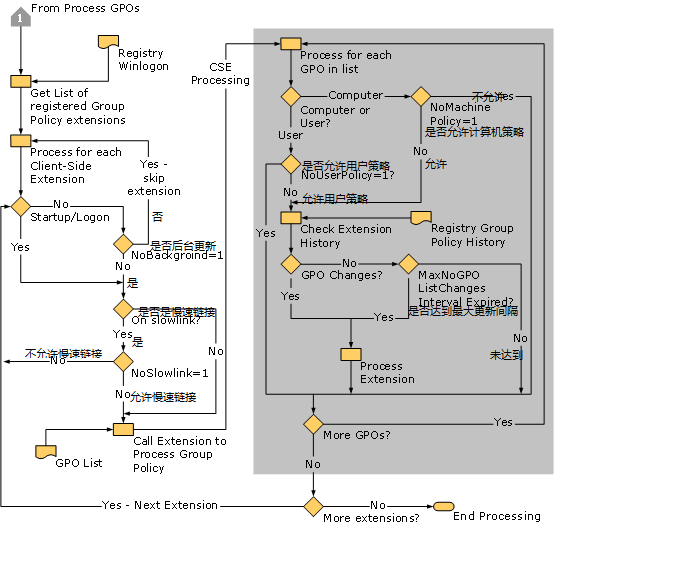


Creating the GPO list involves the following steps:

1. Query the Active Directory for the **gPLink** and **gPOptions** properties in the Site and Domain hierarchies to which the user or computer object belongs.
2. Query the Active Directory for the GroupPolicyContainer objects referenced in the **gPLink** properties.
3. Evaluate security filtering to determine if the user or computer have the Apply Group Policy access permission to the GPO.
4. Evaluate the WMI query against the WMI repository on the client computer to determine if the computer meets the query requirements.

Once the GPO list is created, the Group Policy engine and the CSEs work together to process Group Policy template components. The following figure shows the steps required to determine which CSEs to call.

**Determining CSEs to Call**



Determining which CSEs to call involves the following steps:

1. Retrieve the list of CSEs registered with Winlogon.
2. Check to see whether it is appropriate to run a particular CSE (for example, whether background processing or slow link processing is enabled for the extension).
3. Check the CSE history against list of Applied GPOs. GPOs with new version numbers and GPOs that have settings relevant to the CSE (that is, they have the CSE extension GUID in the Group Policy container gpcUserExtension or gpcMachineExtension properties) are added to the Changed GPO List. GPOs no longer in the Applied GPO List are added to the Deleted GPO List.
4. Check to see whether the appropriate CSE should be processing policy settings for the user or the computer.
5. Check the version number listed in the GPO against its recorded version history in the registry to determine whether the GPO needs reprocessing.

If all of the version numbers are unchanged the MaxNoGPOListChanges interval might have expired; if so, the CSE processes policy settings without regard to an unchanged version number.

Steps 3 through 5 are repeated by each CSE for all GPOs in the GPO list. After one CSE is done, the next CSE that needs to run repeats the entire process.

Group Policy updates are dynamic and occur at specific intervals. If there have been no changes to Group Policy, the client computer still refreshes the security policy settings at regular intervals for the GPO.

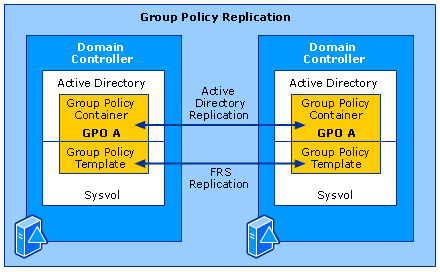
If no changes are discovered, GPOs are not processed. Security policies have a periodic force apply every 16 hours. For security policies, there is a value that sets a maximum limit of how long a client can function without reapplying non-changed GPOs. By default, this setting is every 16 hours plus a randomized delay of up to 30 minutes. Even when GPOs that contain security policy settings do not change, the policy is reapplied every 16 hours.

# Group Policy Replication

In a domain that contains more than one domain controller, Group Policy information takes time to propagate, or replicate, from one domain controller to another. Low bandwidth network connections between domain controllers slow replication. The Group Policy infrastructure has mechanisms to manage these issues.

Each GPO is stored partly in the Sysvol on the domain controller and partly in Active Directory. GPMC and Group Policy Object Editor present and manage the GPO as a single unit. For example, when you set permissions on a GPO in GPMC, GPMC is actually setting permissions on objects in both Active Directory and the Sysvol. It is not recommended that you manipulate these separate objects independently outside of GPMC and the Group Policy Object Editor. As shown in the following figure, it is important to understand that these two separate components of a GPO rely on different replication mechanisms. The file system portion is replicated through FRS, independently of the replication handled by Active Directory. Only the Sysvol subfolder (%systemroot%\SYSVOL\sysvol) is shared and replicated. Sysvol was designed to allow multiple domain’s Sysvols to be replicated in the same tree — each domain’s Sysvol is contained under a subfolder of the Sysvol share. For the current domain, a copy of the domain’s Sysvol subtree is also stored directly under the **%systemroot%\SYSVOL\domain** folder.

## Group Policy Replication



FRS is a multi-master replication service that synchronizes folders between two or more Windows Server 2003 or Windows 2000 systems. Modified files are queued for replication at the point the file is closed. In the case of conflicting modifications between two copies of an FRS replica, the file with the latest modification time will overwrite any other copies. This is referred to as a “last-writer-wins” model.

FRS replication topology configuration is stored as a combination of child objects of each FRS replica partner (in the FRS Subscriptions subcontainer) and objects within another hidden subcontainer of the domain System container. Replication links between systems are maintained as FRS subscription objects. These objects specify the replica partner and the replication schedule. It is possible to view the schedule by browsing to an FRS subscription object and viewing the properties. The replica partner is stored as the object GUID of the computer account of that partner.

Windows 2000 Server and Windows Server 2003 use File Replication Service (FRS) to replicate SYSVOL, whereas Windows Server 2008 uses the newer DFS Replication service when in domains that use the Windows Server 2008 domain functional level, and FRS for domains that run older domain functional levels.

For more information, see [FRS Overview](http://go.microsoft.com/fwlink/?LinkId=121763), or [Overview of DFS Replication](http://go.microsoft.com/fwlink/?LinkId=121762)

# Group Policy Best Practices

优化组策略性能

<http://technet.microsoft.com/zh-cn/magazine/2008.01.gpperf.aspx>