LGPO.exe v3.0

Local Group Policy Object Utility

Overview

LGPO.exe is a command-line utility that is designed to help automate management of Local Group Policy. It can import and apply settings from Registry Policy (Registry.pol) files, security templates, Advanced Auditing backup files, as well as from formatted "LGPO text" files and Policy Analyzer ".PolicyRules" XML files. It can export local policy to a GPO backup. It can export the contents of a Registry Policy file to the "LGPO text" format that can then be edited, and can build a Registry Policy file from an LGPO text file. (The syntax for LGPO text files is described later in this document.)

LGPO.exe has four command-line forms: for importing and applying settings to local policy – including to Multiple Local Group Policy Objects (MLGPO)[1]; for creating a GPO backup; for parsing a Registry Policy file and outputting "LGPO" text; for producing a Registry Policy file from an LGPO text file.

All output is written to LGPO.exe's standard output, and all diagnostic and error information is written to its standard error. Both can be redirected to files using standard command shell operations. To support batch file use, LGPO.exe's exit code is 0 on success and non-zero on any error.

LGPO.exe does not support Group Policy Preferences (GPP) at this time.

What's New in Version 3.0

Two new options were added in LGPO.exe. The first, /ef which enables Group Policy extensions referenced in the backup.xml. The second, /p which allows for importing settings directly from a .PolicyRules file which negates the need to have the actual GPOs on hand. Additionally, LGPO.exe /b and /g now capture locally-configured client-side extensions (CSEs) (which we had an issue with previously). Lastly, /b also correctly captures all user rights assignments, overcoming a bug in the underlying "secedit.exe /export" that fails to capture user rights assignments that are granted to no one.

Terms of Use

Terms of Use have been included in the download.

Importing and applying settings

In this mode, LGPO.exe applies the contents of the supplied input files to local policy. Note that it does not clear or remove pre-existing policy settings that are not specified in the input files, except for the **/ac** command.

The command-line syntax for this mode is:

[1] For more information about MLGPO, see https://docs.microsoft.com/previous-versions/windows/it-pro/windows-vista/cc766291(v=ws.10).

LGPO.exe *command* [...]

Where *command* is one or more of the following, each of which can be repeated:

/g *path*	Import settings from one or more Group Policy backups anywhere under the directory specified by *path*.
/p *path**lgpo.PolicyRules*	Import settings from a Policy Analyzer .PolicyRules file.
/m *path**registry.pol*	Import settings from a Registry Policy file into Computer (Machine) Configuration.
/u *path**registry.pol*	Import settings from a Registry Policy file into system-wide User Configuration.
/ua *path**registry.pol*	Import settings from a Registry Policy file into MLGPO User Configuration for Administrators.
/un *path**registry.pol*	Import settings from a Registry Policy file into MLGPO User Configuration for Non-Administrators.
/u:*username path**registry.pol*	Import settings from a Registry Policy file into MLGPO User Configuration for the specified, valid local account.
/s *path**GptTmpl.inf*	Apply the specified security template.
/a[**c**] *path**audit.csv*	Apply an Advanced Auditing backup (CSV) file. With **/ac**, LGPO.exe clears existing Advanced Auditing settings before applying the settings from the CSV file, and copies the file to the local group policy subdirectory so that the settings appear in the local group policy editor.
/e *name*|*guid*	Enable a Group Policy client side extension for local policy processing. Specify a GUID, or one of these names (case-insensitive): **zone** – Internet Explorer zone mapping extension; needed for Site-To-Zone Assignment List policy. **mitigation** – Mitigation Options extension; needed for the Untrusted Font Blocking policy (Windows 10). **audit** – Advanced Audit Policy Configuration; ensures that GpUpdate.exe also applies advanced audit policy settings. **LAPS** – Local Administrator Password Solution (LAPS) extension.[2] **DGVBS** – Device Guard virtualization-based security extension; needed for Credential Guard and for Device Guard (Windows 10). **DGCI** – Device Guard code integrity policy extension; needed for Device Guard (Windows 10).
/ef *path**backup.xml*	Enable GP extensions referenced in backup.xml from a GPO backup.
/t *path**lgpo.txt*	Apply registry-based commands from an "LGPO text" file.
/boot	Reboot after applying policies.
/v	Verbose output.
/q	Quiet output (no headers).

The **/g** option imports settings from one or more Group Policy backups containing Registry Policy (registry.pol) files, security templates (GptTmpl.inf), Advanced Auditing backups (audit.csv), and backup.xml files that include references to GP client side extensions (CSEs) used by settings in the GPOs. The **/g** option searches the specified *path* directory for files with these exact names and imports them. Each registry.pol file must be in a "Machine" or "User" subdirectory. (The command-line options that

[2] LAPS: https://www.microsoft.com/en-us/download/details.aspx?id=46899

take filename parameters do not require that the filenames adhere to these restrictions.) Note that the **/g** option imports only into system-wide settings and does not support configuring MLGPO.

Use of this LGPO.exe mode requires administrative rights.

In this example (ignore the line wrap), LGPO.exe imports settings from two Registry Policy files into Computer Configuration, from two more Registry Policy files into system-wide User Configuration and the MLGPO User configuration for Non-Administrators, from a security template and from an Advanced Audit backup file after clearing existing auditing policy. It also enables the IE zone mapping extension so that Site-To-Zone Assignment List policies are properly processed. It writes verbose output to `lgpo.out` and any error information to `lgpo.err`.

```
LGPO.exe /e zone /m .\Win10\machine.pol /m .\Win10\IE11.pol /u .\Win10\AllUsers.pol /u
.\Win10\NonAdmin.pol /s .\Win10\GptTmpl.inf /ac .\Win10\audit.csv /v > lgpo.out 2>
lgpo.err
```

This next example searches C:\GPOBackups for files named registry.pol, GptTmpl.inf, or audit.csv and imports their contents into local policy, and also registers any machine or user CSEs referenced in backup.xml files into local policy. As with the previous example, it also writes verbose output to log files.

```
LGPO.exe /g C:\GPOBackups /v > lgpo.out 2> lgpo.err
```

This example does the same, importing settings from just one backed-up GPO. (Note that curly braces require special quoting in PowerShell commands.)

```
LGPO.exe /g C:\GPOBackups\{45CA52BB-19DE-487A-9CE8-0A95B18F6054} /v > lgpo.out 2>
lgpo.err
```

And this example demonstrates applying policy settings from a Policy Analyzer .PolicyRules file, which can include settings from one or more GPOs and associated CSEs:

```
LGPO.exe /p .\MSFT-Win10-v1909-FINAL.PolicyRules /v > lgpo.out 2> lgpo.err
```

These are the GUIDs of Group Policy client side extensions referenced in the Windows 10 v1607 ADMX files that might be needed with the **/e** option:

AppV Policy	{2BFCC077-22D2-48DE-BDE1-2F618D9B476D}
ConfigMgr User State Management Extension	{346193F5-F2FD-4DBD-860C-B88843475FD3}
Cortana Search	{29BBE2D5-DE47-4855-97D7-2745E166DC6D}
Device Guard: Code Integrity Policy	{FC491EF1-C4AA-4CE1-B329-414B101DB823}
Device Guard: Virtualization Based Security	{F312195E-3D9D-447A-A3F5-08DFFA24735E}
Disk Quotas	{3610eda5-77ef-11d2-8dc5-00c04fa31a66}
Internet Explorer Machine Accelerators	{CF7639F3-ABA2-41DB-97F2-81E2C5DBFC5D}
Internet Explorer User Accelerators	{7b849a69-220f-451e-b3fe-2cb811af94ae}
Internet Explorer Zonemapping	{4CFB60C1-FAA6-47f1-89AA-0B18730C9FD3}
LAPS	{D76B9641-3288-4f75-942D-087DE603E3EA}
Microsoft Offline Files	{C631DF4C-088F-4156-B058-4375F0853CD8}
Microsoft User Experience Virtualization	{169EBF44-942F-4C43-87CE-13C93996EBBE}
Mitigation Options: Process Mitigation Options	{4B7C3B0F-E993-4E06-A241-3FBE06943684}

Mitigation Options: Untrusted Font Blocking	{2A8FDC61-2347-4C87-92F6-B05EB91A201A}
QoS Packet Scheduler	{426031c0-0b47-4852-b0ca-ac3d37bfcb39}
Remote Desktop USB Redirection	{4bcd6cde-777b-48b6-9804-43568e23545d}
RemoteApp and Desktop Connections	{4D2F9B6F-1E52-4711-A382-6A8B1A003DE6}
TCPIP	{cdeafc3d-948d-49dd-ab12-e578ba4af7aa}
Windows Search Group Policy Extension	{7933F41E-56F8-41d6-A31C-4148A711EE93}
Windows To Go Hibernate Options	{C34B2751-1CF4-44F5-9262-C3FC39666591}
Windows To Go Startup Options	{BA649533-0AAC-4E04-B9BC-4DBAE0325B12}
Work Folders	{4d968b55-cac2-4ff5-983f-0a54603781a3}

Exporting local policy to a GPO backup

To export the computer's local policy in the form of a GPO backup with an optional GPO display name, run LGPO.exe with this command line syntax:

LGPO.exe /b *path* [**/n** *GPO-display-name*]

LGPO.exe creates a subdirectory under *path* with a newly-generated GUID for the directory name, and exports system-wide local policy settings into that backup directory. The *path* directory must exist and be writable. If you specify a GPO display name that contains spaces, it must be quoted. The name is shown in the Import Settings Wizard in Active Directory Group Policy Management. If you do not specify a display name, LGPO.exe uses "Local Policy Export."

Use of this LGPO.exe mode requires administrative rights. It's important to note that this operation backs up only local policy, not all applied policies and settings.

The GPO backup incorporates results from "secedit.exe /export," "auditpol.exe /backup," and the Machine and User registry.pol files under System32\GroupPolicy. If either registry.pol does not exist, LGPO.exe creates an empty registry.pol for the backup. All locally-registered CSEs are incorporated into the backup's backup.xml file. Note that the **/b** option does not back up MLGPO configuration settings.

Parsing a Registry Policy file to LGPO text

The format of Registry Policy files is a documented, binary file format[3], normally produced by Group Policy editors such as GpEdit.msc. Registry Policy files typically contain Group Policy settings from Administrative Templates, Windows Defender Firewall, AppLocker, Public Key Policies, and more. However, there have not been any good viewers or editors for directly manipulating those files. LGPO.exe defines a custom, Notepad-editable "LGPO text" file format to specify registry-based settings. LGPO.exe can read the content of a Registry Policy file and output it in LGPO text format. You can redirect this output to a file, edit it, and then import the modifications directly into local group policy using the **/t** option described earlier, or produce a new Registry Policy file incorporating your changes using the syntax described later in this document. You can also combine the LGPO text from multiple Registry Policy files into a single LGPO text file and product a "merged" Registry Policy file with it.

[3] Documentation: https://docs.microsoft.com/previous-versions/windows/desktop/Policy/registry-policy-file-format

LGPO.exe /parse [/q] {**/m**|**/u**|**/ua**|**/un**|**/u:***username*} *path**registry.pol*

Registry Policy files do not contain information indicating whether they are for Computer or User configuration. Use **/m** to indicate that the file should be interpreted as Computer Configuration, or one of the **/u** options to indicate User Configuration:

/m	Computer Configuration.
/u	System-wide User Configuration.
/ua	MLGPO User Configuration for Administrators.
/un	MLGPO User Configuration for Non-Administrators.
/u:*username*	MLGPO User Configuration for the specified, valid local account.

Specifying the configuration is important if you later apply the settings using the **/t** option described earlier in this document. LGPO.exe writes the LGPO text to standard output, where you can redirect it to a file. You can also skip the redirection and output the content to the console for immediate viewing. LGPO.exe writes diagnostic and error information to standard error. With the **/q** option it writes to standard error only to report errors.

In this simple example, LGPO.exe converts the contents of the local policy's Computer Configuration registry.pol file and produces an LGPO text file from it. Standard error is not redirected, so diagnostic and error information is shown in the command shell console.

```
LGPO.exe /parse /m C:\Windows\System32\GroupPolicy\Machine\registry.pol > regpol.txt
```

Building a Registry Policy file from LGPO text

With the **/r** and **/w** options, you can build a new Registry Policy file from an LGPO text file.

LGPO.exe /r *path**lgpo.txt* **/w** *path**registry.pol* **[/v]**

Note that because Registry Policy files do not contain information indicating whether they are for Computer or User configuration, those indicators in the LGPO text file are not used. The **/v** option produces verbose output.

LGPO text file format

The registry-based policy input files are Notepad-editable text files, and can be Unicode (little-endian – the Windows default) or ANSI text. Unicode input files must have a Byte Order Marker (BOM) in the first two bytes of the file. Most Windows tools that create Unicode files (including Notepad) automatically insert the correct BOM in the file.

A file can consist of any number of entries. Each entry consists of four consecutive lines:

```
Configuration
Registry Key
Value Name
Action
```

Configuration specifies whether the setting is for Computer Configuration, User Configuration, or an MLGPO User Configuration. A single LGPO text file can contain settings targeting multiple

configurations. The `Configuration` line is case-insensitive, cannot have leading or trailing whitespace, and must be one of the following:

`Computer`	Computer Configuration.
`User`	System-wide User Configuration.
`User:Administrators`	MLGPO User Configuration for Administrators.
`User:Non-Administrators`	MLGPO User Configuration for Non-Administrators.
`User:`*`username`*	MLGPO User Configuration for the named local account.

`Registry Key` specifies the name of a registry key (not including the base key). It should not be quoted even if it contains spaces. For example:
`SOFTWARE\Policies\Microsoft\some policy`

`Value Name` is the name of the registry value to modify. It should not be quoted even if it contains spaces. The value `(Default)` can be used to denote the key's default value. A dummy value such as **`*`** should be used for the CREATEKEY, DELETEALLVALUES, and CLEAR actions. For the DELETEKEYS action, the Value Name entry is a semicolon-delimited list of subkeys to delete from the named Registry Key.

`Action` specifies what action to take, and must look like one of the following:

`DELETE`	Deletes the value (reverting a policy to "not configured"). This inserts a command into the registry.pol file that deletes the named value each time policy is re-applied.

`DWORD:`*`n`*	Sets the value to a REG_DWORD value *n*. E.g.,
	`DWORD:1`
	Values can be specified in hexadecimal by prepending "0x"; e.g.,
	`DWORD:0x1000`

`QWORD:`*`n`*	Sets the value to a REG_QWORD value *n*. E.g.,
	`QWORD:1`
	Values can be specified in hexadecimal by prepending "0x"; e.g.,
	`QWORD:0x1000`

`SZ:`*`text`*	Sets the value to a REG_SZ (text) value *text*. E.g., `SZ:Authorized users only!`
`EXSZ:`*`text`*	Sets the value to a REG_EXPAND_SZ (expandable text) value *text*. E.g., `EXSZ:%USERPROFILE%\Desktop`
`MULTISZ:`*`text`*	Sets a multi-string value. Use the character sequence \0 to separate multiple strings. Example: `MULTISZ:One\0Two\0Three`
`BINARY:`*`data`*	Sets a binary value. Use comma-separated, two-digit hex values on a single line. Example: `BINARY:00,ff,01,fe,02,fd,03,fc`
`CREATEKEY`	Create the key, but do not create any values. (Use **`*`** on the Value Name line.)
`DELETEALLVALUES`	Delete all values from the registry key. (Use **`*`** on the Value Name line.)
`DELETEKEYS`	Deletes one or more subkeys from the named Registry Key. The Value Name line is a semicolon-delimited list of subkeys to delete.

CLEAR	Removes the named key and any commands associated with the key from policy entirely. Note that all other commands (including the Delete command) each insert a command into the policy file. CLEAR deletes commands associated with a key, as well as the key's values and subkeys, from the policy file. The CLEAR has effect only when used with the /t command-line switch.

Because the Action must be specified on one line, the SZ, EXSZ, and MULTISZ string specifiers each support the escape sequences \r, \n, and \\, to indicate carriage return, line feed, and backslash, respectively.

The four lines of an entry must be on consecutive lines. Entries can be separated by blank lines or by comment lines. Blank lines cannot contain any whitespace. Comment lines must begin with a semicolon character. Here is some sample content:

```
; Revert the Intranet zone's "Java Permissions" setting to "not configured"
Computer
Software\Policies\Microsoft\Windows\CurrentVersion\Internet Settings\Zones\1
1C00
DELETE

; Set the Trusted Sites zone's "Java Permissions" setting to "High Safety"
Computer
Software\Policies\Microsoft\Windows\CurrentVersion\Internet Settings\Zones\2
1C00
DWORD:0x10000

; Enable "Prevent ignoring certificate errors"
Computer
Software\Policies\Microsoft\Windows\CurrentVersion\Internet Settings
PreventIgnoreCertErrors
DWORD:1

; Set IE's "Disable AutoComplete for forms" in MLGPO User Config for non-admins
User:Non-Administrators
Software\Policies\Microsoft\Internet Explorer\Main
Use FormSuggest
SZ:no

; Set "Prohibit access to the Control Panel" in MLGPO User Config for
; the local account, "KioskAccount"
User:KioskAccount
Software\Microsoft\Windows\CurrentVersion\Policies\Explorer
NoControlPanel
DWORD:1

; Removes all "allowed remote assistance helpers"
Computer
Software\Policies\Microsoft\Windows NT\Terminal Services\RAUnsolicit
*
DELETEALLVALUES

; Create an empty registry key
Computer
SOFTWARE\Policies\Microsoft\WindowsFirewall\DomainProfile\Logging
*
CREATEKEY

; Remove two policies ("QoS Rule 1" and "QoS Rule 2") from QoS policy
Computer
Software\Policies\Microsoft\Windows\QoS
QoS Rule 1;QoS Rule 2
```

DELETEKEYS