

LGPO.exe v1.00 – Local Group Policy Object Utility

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. 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; 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.

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 settings that are not specified in the input files, with the exception of the **/ac** command.

The command-line syntax for this mode is:

LGPO.exe *command* [...]

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

/g <i>path</i>	Import settings from one or more Group Policy backups anywhere under the directory specified by <i>path</i> .
/m <i>path\registry.pol</i>	Import settings from a Registry Policy file into Computer (Machine) Configuration.
/u <i>path\registry.pol</i>	Import settings from a Registry Policy file into User Configuration.
/s <i>path\GptTmp1.inf</i>	Apply the specified security template.
/a[c] <i>path\audit.csv</i>	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 <i>name guid</i>	Enable a Group Policy client side extension for local policy processing. Specify a GUID, or one of these names: 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 gpupdate also applies advanced audit policy settings.
/t <i>path\lgpo.txt</i>	Apply registry-based commands from an “LGPO text” file.
/boot	Reboot after applying policies.
/v	Verbose output.
/q	Quiet output (no headers).

Registry Policy files, security templates, and Advanced Auditing backup files are typically named “registry.pol,” “GptTmpl.inf,” and “audit.csv,” respectively, so the syntax above uses these names. The files you use do not have to conform to this pattern, though. The **/g** option searches the specified *path* directory for files with these exact names and imports them. “Registry.pol” files must be in a “Machine” or a “User” directory.

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, another Registry Policy file into User Configuration, a security template and 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\user.pol /s
.\win10\GptTmpl.inf /ac .\win10\audit.csv /v > lgpo.out 2> lgpo.err
```

This command searches C:\GPOBackups for files named registry.pol, GptTmpl.inf, and audit.csv files and imports them all into local policy. Like the previous example, it also enables the IE zone mapping extension and writes verbose output to log files.

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

These are the GUIDs of Group Policy client side extensions referenced in the Windows 10 (TH1) ADMX files that might be needed with the **/e** option:

```
{2A8FDC61-2347-4C87-92F6-B05EB91A201A} MitigationOptions
{346193F5-F2FD-4DBD-860C-B88843475FD3} ConfigMgr User State Management Extension.
{3610eda5-77ef-11d2-8dc5-00c04fa31a66} Microsoft Disk Quota
{426031c0-0b47-4852-b0ca-ac3d37bfc39} QoS Packet Scheduler
{4bcd6cde-777b-48b6-9804-43568e23545d} Remote Desktop USB Redirection
{4CFB60C1-FAA6-47f1-89AA-0B18730C9FD3} Internet Explorer Zonemapping
{4d968b55-cac2-4ff5-983f-0a54603781a3} Work Folders
{7b849a69-220f-451e-b3fe-2cb811af94ae} Internet Explorer User Accelerators
{BA649533-0AAC-4E04-B9BC-4DBAE0325B12} Windows To Go Startup Options
{C34B2751-1CF4-44F5-9262-C3FC39666591} Windows To Go Hibernate Options
{C631DF4C-088F-4156-B058-4375F0853CD8} Microsoft Offline Files
{cdea9c3d-948d-49dd-ab12-e578ba4af7aa} TCP/IP
{CF7639F3-ABA2-41DB-97F2-81E2C5DBFC5D} Internet Explorer Machine Accelerators
```

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 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.

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.

Parsing a Registry Policy file to LGPO text

The format of Registry Policy files is a [documented](#), binary file format, normally produced by Group Policy editors such as GpEdit.msc. 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.

LGPO.exe /parse [/q] {/m|/u} *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 **/u** to indicate User Configuration. This is important if you later apply the settings using the **/t** option described earlier. 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 a Registry Policy 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 .\win10\machine.pol > win10Machine.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 must be either “Computer” or “User”, for Computer or User Configuration, respectively. It’s case-insensitive, but leading/trailing whitespace is not allowed.

Registry Key specifies the name of a registry key (not including the base key); e.g.,
SOFTWARE\Policies\Microsoft\some policy

Value Name is the name of the registry value to modify. The value (Default) can be used to denote the key’s default value. (A dummy value such as “*” should be used for the CREATEKEY and DELETEALLVALUES actions.)

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

DELETE	Deletes the value (reverting a policy to “not configured”)
DWORD: <i>n</i>	Sets the value to a REG_DWORD value <i>n</i> . E.g., DWORD: 1 Values can be specified in hexadecimal by prepending “0x”; e.g., DWORD: 0x1000
SZ: <i>text</i>	Sets the value to a REG_SZ (text) value <i>text</i> . E.g., SZ: Authorized users only!
EXSZ: <i>text</i>	Sets the value to a REG_EXPAND_SZ (expandable text) value <i>text</i> . E.g., EXSZ: %USERPROFILE%\Desktop
MULTISZ: <i>text</i>	Sets a multi-string value. Use the character sequence \0 to separate multiple strings. Example: MULTISZ: One\0Two\0Three
BINARY: <i>data</i>	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.)
------------------------	--

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 User Configuration
User
Software\Policies\Microsoft\Internet Explorer\Main
Use FormSuggest
SZ:no
```

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

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