

Command-Line Options

05/30/2018 • 7 minutes to read

The executable program that interprets packages and installs products is Msiexec.exe. Note that Msiexec also sets an error level on return that corresponds to [system error codes](#). Command-line options are case-insensitive.

The command-line options in the following table are available with Windows Installer 3.0 and earlier versions. The [Standard Installer Command-Line Options](#) are also available beginning with Windows Installer 3.0.

Option	Parameters	Meaning
/i	<i>Package ProductCode</i>	Installs or configures a product.
/f	[p o e d c a u m s v] <i>Package ProductCode</i>	Repairs a product. This option ignores any property values entered on the command line. The default argument list for this option is 'omus.' This option shares the same argument list as the REINSTALLMODE property. p - Reinstalls only if file is missing. o - Reinstalls if file is missing or an older version is installed. e - Reinstalls if file is missing or an equal or older version is installed. d - Reinstalls if file is missing or a different version is installed. c - Reinstalls if file is missing or the stored checksum does not match the calculated value. Only repairs files that have msidbFileAttributesChecksum in the Attributes column of the File table. a - Forces all files to be reinstalled. u - Rewrites all required user-specific registry entries. m - Rewrites all required computer-specific registry entries. s - Overwrites all existing shortcuts. v - Runs from source and re-caches the local package. Do not use the v reinstall option for the first installation of an application or feature.
/a	<i>Package</i>	Administrative installation option. Installs a product on the network.
/x	<i>Package ProductCode</i>	Uninstalls a product.
/j	[u m] <i>Package</i> or [u m] <i>Package/tTransform List</i> or [u m] <i>Package/gLanguageID</i>	Advertises a product. This option ignores any property values entered on the command line. u - Advertises to the current user. m - Advertises to all users of machine. g - Language identifier. t - Applies transform to advertised package.

Option	Parameters	Meaning
/L	<code>[i w e a r u c m o p v x + !]*</code> <i>Logfile</i>	<p>Writes logging information into a logfile at the specified existing path. The path to the logfile location must already exist. The installer does not create the directory structure for the logfile. Flags indicate which information to log. If no flags are specified, the default is 'iwearmo.'</p> <p>i - Status messages. w - Nonfatal warnings. e - All error messages. a - Start up of actions. r - Action-specific records. u - User requests. c - Initial UI parameters. m - Out-of-memory or fatal exit information. o - Out-of-disk-space messages. p - Terminal properties. v - Verbose output. x - Extra debugging information. Windows Installer 2.0: Not supported. The x option is available with Windows Installer version 3.0.3790.2180 and later.</p> <p>+ - Append to existing file. ! - Flush each line to the log. "*" - Wildcard, log all information except for the v and x options. To include the v and x options, specify "/!*vx".</p> <p>[!Note] For more information about all the methods that are available for setting the logging mode, see Normal Logging in the Windows Installer Logging section</p>
/m	<i>filename</i> <p>[!Note] The length of <i>filename</i> must be no more than eight characters.</p>	<p>Generates an SMS status .mif file. Must be used with either the install (-i), remove (-x), administrative installation (-a), or reinstall (-f) options. The ISMIF32.DLL is installed as part of SMS and must be on the path. The fields of the status mif file are filled with the following information:</p> <p>Manufacturer - Author Product - Revision Number Version - Subject Locale - Template Serial Number - not set Installation - set by ISMIF32.DLL to "DateTime" InstallStatus - "Success" or "Failed" Description - Error messages in the following order: 1) Error messages generated by installer. 2) Resource from Msi.dll if installation could not commence or user exit. 3) System error message file. 4) Formatted message: "Installer error %i", where %i is error returned from Msi.dll.</p>
/p	<i>PatchPackage</i> [<i>patchPackage2</i>]	<p>Applies a patch. To apply a patch to an installed administrative image you must combine the following options:</p> <p><i>/p</i> [<i>patchPackage2</i>] <i>/a</i></p>

Option	Parameters	Meaning
/q	n b r f	<p>Sets user interface level.</p> <p>q , qn - No UI</p> <p>qb - Basic UI. Use qb! to hide the Cancel button.</p> <p>qr - Reduced UI with no modal dialog box displayed at the end of the installation.</p> <p>qf - Full UI and any authored FatalError, UserExit, or Exit modal dialog boxes at the end.</p> <p>qn+ - No UI except for a modal dialog box displayed at the end.</p> <p>qb+ - Basic UI with a modal dialog box displayed at the end. The modal box is not displayed if the user cancels the installation. Use qb+! or qb!+ to hide the Cancel button.</p> <p>qb- - Basic UI with no modal dialog boxes. Please note that /qb+ - is not a supported UI level. Use qb-! or qb!- to hide the Cancel button.</p> <p>Note that the ! option is available with Windows Installer 2.0 and works only with basic UI. It is not valid with full UI.</p>
/? or /h		Displays copyright information for Windows Installer.
/y	module	<p>Calls the system function DllRegisterServer to self-register modules passed in on the command line. Specify the full path to the DLL. For example, for MY_FILE.DLL in the current folder you can use:</p> <p>msiexec /y .\MY_FILE.DLL</p> <p>This option is only used for registry information that cannot be added using the registry tables of the .msi file.</p>
/z	module	<p>Calls the system function DllUnRegisterServer to unregister modules passed in on the command line. Specify the full path to the DLL. For example, for MY_FILE.DLL in the current folder you can use:</p> <p>msiexec /z .\MY_FILE.DLL</p> <p>This option is only used for registry information that cannot be removed using the registry tables of the .msi file.</p>
/c	<p>Advertises a new instance of the product. Must be used in conjunction with /t. Available starting with the Windows Installer version that is shipped with Windows Server 2003 and Windows XP with Service Pack 1 (SP1).</p>	
/n	ProductCode	<p>Specifies a particular instance of the product. Used to identify an instance installed using the multiple instance support through a product code changing transforms. Available starting with the Windows Installer version shipped with Windows Server 2003 and Windows XP with SP1.</p>

The options /i, /x, /f[p|o|e|d|c|a|u|m|s|v], /j[u|m], /a, /p, /y and /z should not be used together. The one exception to this rule is that patching an [administrative installation](#) requires using both /p and /a. The options /t, /c and /g should only be used with /j. The options /l and /q can be used with /i, /x, /f[p|o|e|d|c|a|u|m|s|v], /j[u|m], /a, and /p. The option /n can be used with /i, /f, /x and /p.

To install a product from A:\Example.msi, install the product as follows:

```
msiexec /i A:\Example.msi
```

Only [public properties](#) can be modified using the command line. All property names on the command line are interpreted as uppercase but the value retains case sensitivity. If you enter **MyProperty** at a command line, the installer overrides the value of MYPROPERTY and not the value of **MyProperty** in the Property table. For more information, see [About Properties](#).

To install a product with PROPERTY set to VALUE, use the following syntax on the command line. You can put the property anywhere except between an option and its argument.

Correct syntax:

```
msiexec /i A:\Example.msi PROPERTY=VALUE
```

Incorrect syntax:

```
msiexec /i PROPERTY=VALUE A:\Example.msi
```

Property values that are literal strings must be enclosed in quotation marks. Include any white spaces in the string between the marks.

```
msiexec /i A:\Example.msi PROPERTY="Embedded White Space"
```

To clear a public property by using the command line, set its value to an empty string.

```
msiexec /i A:\Example.msi PROPERTY=""
```

For sections of text set apart by literal quotation marks, enclose the section with a second pair of quotation marks.

```
msiexec /i A:\Example.msi PROPERTY="Embedded ""Quotes"" White Space"
```

The following example shows a complicated command line.

```
msiexec /i testdb.msi INSTALLLEVEL=3 /l* msi.log COMPANYNAME="Acme ""Widgets"" and ""Gizmos.""
```

The following example shows advertisement options. Note that switches are not case-sensitive.

```
msiexec /JM msisample.msi /T transform.mst /LIME logfile.txt
```

The following example shows you how to install a new instance of a product to be advertised. This product is authored to support multiple instance transforms.

```
msiexec /JM msisample.msi /T :instance1.mst;customization.mst /c /LIME logfile.txt
```

The following example shows how to patch an instance of a product that is installed using multiple instance transforms.

```
msiexec /p msipatch.msp;msipatch2.msp /n {00000001-0002-0000-0000-624474736554} /qb
```

When you apply patches to a specific product, the /i and /p options cannot be specified together in a command line. In this case, you can apply patches to a product as follows.

```
msiexec /i A:\Example.msi PATCH=msipatch.msp;msipatch2.msp /qb
```

The [PATCH](#) property cannot be set in a command line, when /p option is used. If the **PATCH** property is set when the /p option is used, the value of **PATCH** property is ignored and overwritten.