Auditpol.exe

List the policies

Before you can use audit policies, you need to know which policies are available and whom they affect. Windows applies categories of auditing policies to specific users, so you actually have two concerns when discovering the current auditing configuration. The AuditPol /List command makes it possible to check users, auditing categories, and auditing subcategories as described in the following sections.

Auditpol list

1 out of 1 rated this helpful - Rate this topic

Published: April 17, 2012
Updated: April 17, 2012
Applies To: Windows 7, Windows 8, Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, Windows Vista
Lists audit policy categories and/or subcategories, or lists users for whom a per-user audit policy is defined.
For examples of how this command can be used, see Examples.

Syntax

```
auditpol /list
[/user|/category|subcategory[:<categoryname>|<{guid}>|*]]
[/v] [/r]
```

Parameters

Parameter	Description
/user	Retrieves all users for whom the per-user audit policy has been defined. If used with the /v parameter, the security identifier (SID) of the user is also displayed.
/category	Displays the names of categories understood by the system. If used with the /v parameter, the category globally unique identifier (GUID) is also displayed.
/subcategory	Displays the names of subcategories and their associated GUID.
/v	Displays the GUID with the category or subcategory, or when used with /user, displays the SID of each user.
/r	Displays the output as a report in comma-separated value (CSV) format.
/?	Displays help at the command prompt.

Remarks

For all list operations for the per-user policy, you must have Read permission on that object set in the security descriptor. You can also perform list operations by possessing the **Manage auditing and security log**(SeSecurityPrivilege) user right. However, this right allows additional access that is not necessary to perform the list operation.

Examples

To list all users who have a defined audit policy, type:

```
Auditpol /list /user
```

To list all users who have a defined audit policy and their associated SID, type:

```
Auditpol /list /user /v
```

To list all categories and subcategories in report format, type:

```
Auditpol /list /subcategory:* /r
```

To list the subcategories of the Detailed Tracking and DS Access categories, type:

```
Auditpol /list /subcategory:"Detailed Tracking","DS Access"
```

List audit users

To discover which users are audited, type **AuditPol /List /User** and press Enter. The output of this command provides a list of which users are audited, but not how they're being audited. To discover how the user is being audited, you type **AuditPol /Get /User:*UserName* / Category:*** and press Enter, where*UserName* is the user's name (see the "Get a Policy" section of the chapter for additional information). If you also want to know the user's Security Identifier (SID), type **AuditPol /List /User /V** and press Enter. The SID comes in useful for a number of purposes and ensures that you can uniquely identify the user to the system.

List audit categories

Many of the AuditPol commands require that you know a category. If you want information for all categories, you simply use the asterisk (*), but often the wildcard search returns far too much information to be useful unless you limit the output in some other way. Consequently, knowing the precise category you want is important in many situations. To obtain a basic category listing, type **AuditPol /List / Category** and press Enter. In most cases, the basic listing is all you need. However, if you plan to work with the category at a detailed level or want to search for its entry in the registry, you need a Globally Unique Identifier (GUID) that precisely identifies the category to the system. To obtain this information, type **AuditPol /List /Category /V** and press Enter.

List audit subcategories

Categories are divided into subcategories. For example, the Object Access category contains a subcategory

of File System (among other subcategories). You can choose to audit a user's access to the file system, without monitoring other kinds of Object Access, by specifying a subcategory. To obtain the subcategories of a specific category, type**AuditPol /List /Subcategory:"*CategoryName*"** and press Enter, where*CategoryName* is the name of any category you want to see.

If you want to see multiple categories, simply create a list separated by commas of category names. For example, to see the subcategories of the Account Logon and Account Management categories, you'd type **AuditPol /List /Subcategory:"Account Logon","Account Management"** and press Enter. To see all of the subcategories for every category, type **AuditPol /List /Subcategory:*** and press Enter. As with categories, subcategories have GUIDs. To see the GUIDs for the subcategories, add the /V command line switch.

Get a policy

Listing a policy simply tells you that the policy exists but doesn't tell you the policy setting. Getting a policy won't tell you that the policy exists—you must already know that the policy exists. However, it does tell you how the policy is configured. Even though listing and getting may sound a lot alike, the two are completely different. The AuditPol / Get command is all about discovering the system settings.

It's also important to understand that audit policies are configured at two levels. First, you can configure an audit policy at the system level, which means that the policy affects everyone. Second, you can configure an audit policy for a specific user, which means that the policy affects only that user. The AuditPol /Get /User command tells you about specific user settings, while the AuditPol /Get /Category and AuditPol /Get /Subcategory commands tell you about system-level settings.

A special setting level affects the system directly when an audit event occurs. For example, the CrashOnAuditFail option causes the system to crash when the auditing system fails for some reason. This is a safety feature because it ensures that no one can turn off auditing and then continue to use the system unless they use the standard methods to do so and have the proper rights. The following sections describe all of these AuditPol /Get command scenarios.

Get audit users

The AuditPol /Get /User command obtains information about a specific user. In most cases, you want to know a user's full rights, so you'll type **AuditPol /Get /User:*UserName* /Category:***, where *UserName* is the name of the user, and press Enter. However, you can specify a particular category to discover information about just that category or you can use the /Subcategory command line switch to be even more selective and discover information about just one setting. The output you see contains three columns: the name of the category or subcategory, the inclusive setting, and the exclusive setting.

NOTE: When you set a user audit policy, it's either inclusive or exclusive. An inclusive policy is one that adds to the system-level settings. For example, if you audit the user's failure to log on to the system, it's an inclusive policy because it's in addition to any system-level settings. However, if the system normally monitors logon failures, but you don't want to check a particular user, then you'd create an exclusive policy. Even though everyone else is monitored, this particular user is excluded from the policy. It's unusual to create exclusive policies—inclusive policies are far more common.

You may need to output the user settings in a form that you can import into a database. In this case, you'd add the /R command line switch to create Comma Separated Value (CSV) output. For example, if you need

to retrieve the settings for user Jamal and put them in a CSV file, you'd type **AuditPol /Get /User:Jamal /Category:* /R > AuditPol.CSV** and press Enter.

Get audit categories

The AuditPol /Get /Category command obtains the system-wide settings for both categories and subcategories. Of course, you can choose to obtain a specific category by using the category name in place of *. For example, to obtain the Logon/Logoff category, you type **AuditPol /Get /Category:"Logon/ Logoff"** and press Enter. As with the user information, you can output the categories to CSV format using the /R command line switch.

Get audit subcategories

Use the AuditPol /Get /Subcategory command when you need to obtain the system-wide setting for a single subcategory. For example, to retrieve the status of the Logon subcategory, you'd type **AuditPol /Get / Subcategory:"Logon"** and press Enter. Unlike the /Category command line switch, you can't use * with the /Subcategory command line switch.

Get audit options

The AuditPol /Get /Option command retrieves audit policy settings that affect the system as a whole when certain audit policy events occur. The following list describes each of these options:

- **CrashOnAuditFail:** When you enable this setting, it forces the system to crash should the auditing system become unable to log events. The advantage to this setting is that it forces everyone to use the auditing policies you set. However, the disadvantage is that an outsider could use this option to force the

server to crash and cause an apparent Distributed Denial of Service (DDoS) attack. You need to use this setting with care. After this event occurs, only administrators can log on to the system. The administrator must fix whatever caused the crash before the system will allow anyone to log back on. This setting is generally useful on client systems, but not recommended for servers.

- **FullPrivilegeAuditing:** When you tell the system to audit privileges, it normally does so for most privileges, but it leaves out a few commonly-used privileges to keep the event log from quickly overflowing, such as the following privileges:

  -- Generate security audit (SeAuditPrivilege)
  -- Bypass traverse checking (SeChangeNotifyPrivilege) debug programs (SeDebugPrivilege)
  -- Create a token object (SeCreateTokenPrivilege)
  -- Replace process-level token (SeAssignPrimaryTokenPrivilege)
  -- Generate security audits (SeAuditPrivilege)
  -- Back up files and directories (SeBackupPrivilege)
  -- Restore files and directories (SeRestorePrivilege)

Enabling this setting forces the system to audit all privilege changes except SeAuditPrivilege. You can't audit the SeAuditPrivilege because it would cause an endless loop — every access to the audit system generates this privilege and therefore every entry to the log would generate another SeAuditPrivilege event.

- **AuditBaseObjects and AuditBaseDirectories:** Kernel objects come in two forms: container objects and base objects. The AuditBaseObjects policy affects base objects, those that can't contain object objects such as semaphores and mutexes. The AuditBaseDirectories policy affects container objects, those that

can contain other objects, such as directories. Many kernel objects are unnamed and rely only on a handle that's accessible to just the process that created the object for access. Unnamed kernel objects are secure, but they don't allow interprocess communication, which is often necessary in applications. Named kernel objects do allow interprocess communication, but they present security risks because another process (other than those that should use the named process) can interact with the kernel object should it discover the object's name. Setting either of these options forces the operating system to assign a System Access Control List (SACL) to the named objects so that the auditing system can monitor them. The normal use for these settings is to detect and thwart squatting attacks. A problem with these settings is that you normally must reboot the system before the changes you make take effect.

You use these options individually. For example, to obtain the status of the CrashOnAuditFail, you type **AuditPol /Get /Option:CrashOnAuditFail** and press Enter. Unlike other audit policy settings, options are either enabled or disabled.

Set a policy

Setting a policy is the act of creating a new entry for the system or a particular user. These settings work as stated in the "Get a Policy" section of the chapter. When you create a new policy, the user or the system as a whole is monitored for the success or failure of certain actions. You can also enable or disable audit options that perform a task based on an audit event (such as crashing the system when someone tries to override the audit system). The following sections describe how to set an audit policy.

Set audit users

The AuditPol /Set /User command controls settings made to a specific user. When working with users, you

must remember that you can create inclusive settings that add to the system-level settings or exclusive settings that remove auditing from the system-level settings. Audits can affect failures and successes. You can also enable or disable a setting. For example, to set a user account to add (inclusive) failure auditing to the Object Access category, you'd type **AuditPol /Set /User:*Username*/Category:"Object Access" /Include /Failure:Enable,** where *Username*is the name of the user, and press Enter.

All user-level settings follow this same pattern. You provide the username, a category or subcategory, whether the setting is inclusive or exclusive, whether the auditing is for a success or failure, and whether the setting is enabled or disabled. As another example, let's say you want to create an exclusion for a user for Logon subcategory auditing for both success and failure. In this case, you'd type **AuditPol /Set /User:*Username*/Subcategory:"Logon" /Exclude /Failure:Enable /Success:Enable** and press Enter.

Set audit categories

The AuditPol /Set /Category command controls settings made to the system as a whole. Unlike user-level settings, you simply set the policy to monitor success or failure. There isn't any concept of inclusion or exclusion. For example, to audit Account Logon failures, you'd type**AuditPol /Set /Category:"Account Logon" /Failure:Enable** and press Enter. AuditPol sets all of the subcategories for the entire Account Logon category to audit failures.

Set audit subcategories

The AuditPol /Set /Subcategory command controls settings made to the system as a whole, just like the category-level command. However, this command lets you set the individual subcategory entries, rather than an entire category. For example, you might want to failure audit the Credential Validation subcategory

of the Account Logon category. To perform this task, you type **AuditPol /Set /Subcategory:"Credential Validation" /Failure:Enable**and press Enter.

Set audit options

The AuditPol /Set /Option command controls the audit policy options described in the "Get Audit Options" section of the chapter. You either enable or disable these options. For example, to enable the CrashOnAuditFail option, you type **AuditPol /Set /Option:CrashOnAuditFail /Value:Enable** and press Enter.