



Configuring Variables

Overview

Configuration variables are named configuration data items that you create for repeated use in other configuration items such as file access control rules, network access control rules, and alerts. You can group files together, as well as network addresses, and network services. Once configured, you enter these global variables in corresponding fields for other CSA MC items.

You use configuration variables to help build the rules that form your policies. Using variables makes it easy for you to maintain policies by letting you make any necessary modifications in one place and having those changes instantiated across all rules and policies.

This section contains the following topics.

- [Where Variables are Used, page 9-2](#)
- [COM Component Sets, page 9-3](#)
- [COM Component Extract Utility, page 9-6](#)
- [Data Sets, page 9-7](#)
- [File Sets, page 9-10](#)
- [Network Address Sets, page 9-15](#)
- [Network Services, page 9-18](#)
- [Registry Sets, page 9-21](#)
- [Query Settings, page 9-25](#)

- [Localized Language Version Support, page 9-30](#)

Where Variables are Used

Figure 9-1 displays how variables relate to access control rules. In the diagram, variables (Event Sets, Query Settings, File Sets, Network Address Sets, Network Services, Registry Sets, COM Component Sets, and Data Sets) are shown on the left and the rule types they can be applied to are shown on the right.

Figure 9-1 Variable Use in Rules



Note

Using variables is optional (note that Application Classes are included in this diagram, but they are not optional). Nearly all the information used in variable configurations can also be entered directly into corresponding rule configuration fields. Variables are simply a tool meant to simplify the creation of rules, especially if the same configurations are used in multiple rules.

**Note**

You can use the Compare button in Variable list views to compare and merge similar variables. See [Comparing Configurations, page 5-27](#) for details on using the Compare tool.

See for [Chapter 10, “Event Logging and Alerts”](#) details on configuring Event Sets.

Display only Show All mode Option

Each individual variable page (including Application Classes) contains a Display only in Show All mode checkbox. If your lists of variables are growing too long in rule or application configuration pages, clicking the Display only in Show All mode checkbox on a variable page causes that variable to no longer appear in lists for that variable type. To display hidden items, you must go to the Admin Preferences page and choose another admin preferences that Always uses Show All mode or change the preference assigned to you. See [Configuring Role-Based Administration, page 2-5](#).

COM Component Sets

Configure COM component sets for use in COM component access control rules. COM objects are groupings of COM Program IDs (PROGID's) and/or COM Class IDs (CLSID's) under one common name. This name is then used in COM component access control rules to allow or deny access to the COM component set name. All COM components that match the entries of a given component set are relevant to the rule in which the set is used.

You can also use pattern matching when creating COM component sets. For example, entering "Word.*" would match "Word.Application" and "Word.Document".

CSA MC ships with several pre-configured COM component sets you can use as well.

**Note**

This is not available for UNIX configurations.



Note CSA MC provides a COM component import utility which installs with each Cisco Security Agent. Running this utility extracts all COM component PROGID's and CLSID's for software running on the system in question. See [page 9-6](#) for instructions.

To configure a COM component set, do the following.

-
- Step 1** From the menu bar **Configuration** drop-down list, move the mouse over **Variables**. A cascading menu with further selections appears.
- Step 2** Select **COM Component Sets** from the cascading menu. Any existing COM component set configurations are shown.
- Step 3** Click the **New** button to create a new COM component set. This takes you to the configuration view (see [Figure 9-2](#)).
- Step 4** In the available edit fields, enter the following information (See [Using the Correct Syntax, page 2-32](#). You can also click the Quick Help question mark beside each field for syntax information.):
- **Name**—This is a unique name for this COM component set. Generally, it's a good idea to adopt a naming convention that lets you quickly enter COM component set names in a corresponding rule configuration field.
 - **Description**—This is a line of text that is displayed in the list view helping you to identify this particular COM component set configuration.
 - **Display only in Show All mode**—If your lists of variables are growing too long in rule or application configuration pages, clicking the Display only in Show All mode checkbox on a variable page causes that variable to no longer appear in selection lists for that variable type. This feature works in conjunction with Admin Preference settings. You must go the Admin Preferences page to make the item reappear. See [Configuring Role-Based Administration, page 2-5](#).
- Step 5** **PROGID's/CLSID's matching**—Enter the COM component PROGID's or CLSID's here (one per line) to which you want to impose restrictions.
- By default, this field has an <all> entry indicating all PROGID's and CLSID's. When you click inside this field, the <all> disappears so that you can enter your own restrictions.

When entering PROGID's, use syntax as shown in the following example:

```
Outlook.Application
```

When entering CLSID's (uppercase hexadecimal), using the following syntax (You must include the brackets shown here.):

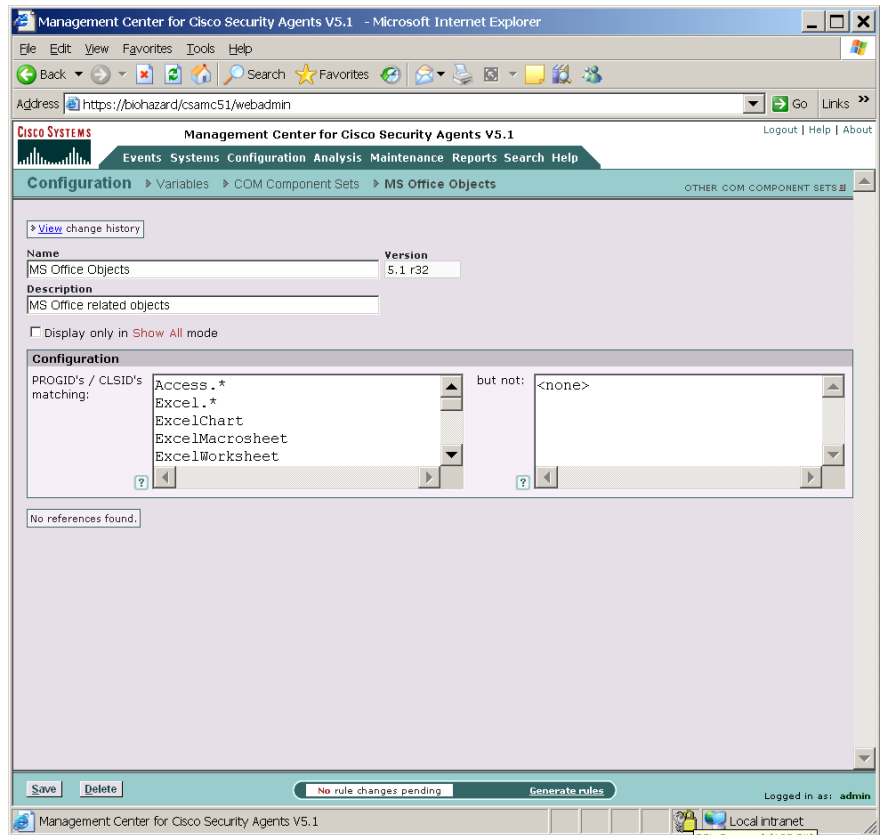
```
{000209FF-0000-0000-C000-000000000046}
```

Step 6 **but not**—Make exceptions to PROGID's or CLSID's you've entered in the PROGID's/CLSID's matching field.

By default, this field has a <none> entry indicating no exceptions. When you click inside this field, the <none> disappears so that you can enter your own exceptions.

When all required information is entered, click the **Save** button to save your COM component set in the CSA MC database.

Figure 9-2 COM Component Set Configuration View



153538

COM Component Extract Utility

CSA MC provides a COM component extraction utility, called `extract_com`, which installs in the `Cisco\CSAgent\bin` directory with each Cisco Security Agent. Running this utility extracts all COM component PROGID's and CLSID's for software installed on the system in question and places this data into a text file. You can cut and paste these ID's from the text file into your COM component sets and access rules.

See [Using the COM Extract Utility, page 12-9](#).

Data Sets

Configure data sets for use in data access control rules. Data sets are groupings of data strings under one common name. These strings represent a set of patterns that will be matched against the URI portion of HTTP requests. The name of the data set is then used in rules that control data access permissions and restrictions. All the data parameters that exist under that name are then applied to the rule where the name is used.

CSA MC ships with several pre-configured Data Sets you can use. The pre-configured data sets group patterns to match based upon the following:

- functional associations of meta-characters (e.g. "(" and ")")
- examples of known classes of attacks
- Web server specific exploits

The following is an example of an HTTP request attempting to execute an attack by invoking a command shell to obtain a directory listing. A data set of this syntax, *cmd.exe*, would stop not only this exploit but any other exploit trying to make use of a command shell.

```
GET /scripts/..%255c%255c../winnt/system32/cmd.exe?/c+dir
```

**Note**

Not all pre-configured data sets are used in pre-configured policies. For example, some attack fingerprints or command arguments might be acceptable on one deployment of a web server, but not be acceptable for a different deployment. Therefore, pre-configured data sets used in shipped policies may require modification if legitimate, but blocked meta-characters are being used by a web server.

**Note**

Additionally, modifying the preconfigured data sets allows you to block a pattern which specifically matches a new/old exploit or attack.

To configure a data set, do the following.

Step 1

From the menu bar **Configuration** drop-down list, move the mouse over **Variables**. A cascading menu with further selections appears.

- Step 2** Select **Data Sets** from the cascading menu. Any existing data set configurations are shown.
- Step 3** Click the **New** button to create a new data set. This takes you to the data set configuration view.
- Step 4** In the available edit fields, enter the following information. (Note that you can click the Quick Help question mark beside each field for syntax information.):
- **Name**—This is a unique name for this data set. Generally, it’s a good idea to adopt a naming convention that lets you quickly enter data set names in a corresponding rule configuration field.
 - **Description**—This is a line of text that is displayed in the list view helping you to identify this particular data set configuration.
 - **Display only in Show All mode**—If your lists of variables are growing too long in rule or application configuration pages, clicking the Display only in Show All mode checkbox on a variable page causes that variable to no longer appear in selection lists for that variable type. This feature works in conjunction with Admin Preference settings. You must go the Admin Preferences page to make the item reappear. See [Configuring Role-Based Administration, page 2-5](#).
- Step 5** **Patterns matching**—Enter the data strings here (one per line) to which you want to impose restrictions. By default, this field has an <all> entry indicating all strings. When you click inside this field, the <all> disappears so that you can enter your own data. This pattern is used by HTTP Web servers to match against the requested URI (Uniform Resource Identifier) to enforce allow/deny Data access control rules.

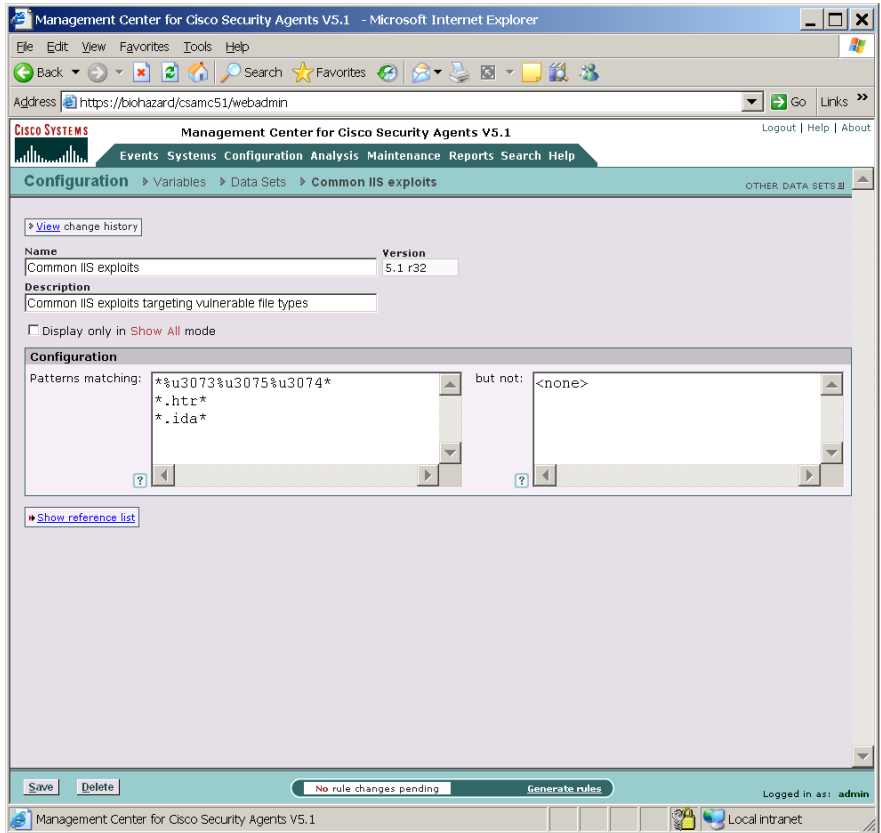


Note When entering data patterns, the “*” character is a generic wildcard specification.

- Step 6** **but not**—Make exceptions to the data strings you’ve entered in the directories matching field. By default, this field has a <none> entry indicating no exceptions. When you click inside this field, the <none> disappears so that you can enter your own exceptions.
- Step 7** When all required information is entered, click the **Save** button to save your data set in the CSA MC database.

You can now enter this data set name by clicking the **Insert Data Set** link in the data access control rule files field.

Figure 9-3 Data Set Configuration View



153539

File Sets

Configure file sets for use in file access control rules and application classes. File sets are groupings of individual files and directories under one common name. This name is then used in rules that control directory and file permissions and restrictions. All the parameters that exist under that name are then applied to the rule where the name is used.

CSA MC ships with several pre-configured File Sets you can use.

To configure a file set, do the following.

-
- Step 1** From the menu bar **Configuration** drop-down list, move the mouse over **Variables**. A cascading menu with further selections appears.
 - Step 2** Select **File Sets [UNIX or Windows]** from the cascading menu. Any existing file set configurations are shown.
 - Step 3** Click the **New** button to create a new file set. This takes you to the file set configuration view (see [Figure 9-4](#)).
 - Step 4** In the available edit fields, enter the following information (See [Using the Correct Syntax, page 2-32](#). You can also click the Quick Help question mark beside each field for syntax information.):
 - **Name**—This is a unique name for this file set. Generally, it's a good idea to adopt a naming convention that lets you quickly enter file set names in a corresponding rule configuration field. When using configuration variables in file access rules, network access rules and application classes, you must enter the variable name preceded by a dollar sign:

For example, if you have a file set variable named `cgi_files`, you must enter `$cgi_files` into any edit field where you are using this variable. The dollar sign tells CSA MC that this is a variable value.
 - **Description**—This is a line of text that is displayed in the list view helping you to identify this particular file set configuration.
 - **Display only in Show All mode**—If your lists of variables are growing too long in rule or application configuration pages, clicking the Display only in Show All mode checkbox on a variable page causes that variable to no longer appear in selection lists for that variable type. This feature works in

conjunction with Admin Preference settings. You must go the Admin Preferences page to make the item reappear. See [Configuring Role-Based Administration, page 2-5](#).

Step 5 Operating System—When you create a file set, you must select to either create a UNIX or a Windows file set. Your file set is then designated for all UNIX or all Windows platforms. Optionally, you select to target an operating system more narrowly by selecting a specific UNIX or Windows operating system from the **Target** pulldown menu.

Step 6 Directories matching—Enter the directories and files here (one per line) to which you want to impose restrictions.

By default, this field has an <all> entry indicating all directories. When you click inside this field, the <all> disappears so that you can enter your own directory restrictions. When entering directory restrictions, use the following syntax (See [Using the Correct Syntax, page 2-32](#)):

Windows example:

```
c:\Program Files\**\*SQL*\bin\**
\Program Files\**\*SQL*\bin
```

UNIX example:

```
/apache/webroot/**
/usr/admn/sg
```



Note See [Using the Correct Syntax, page 2-32](#) for details for information on protecting directory paths and files.

Step 7 but not—Make exceptions to the files and directories you’ve entered in the directories matching field. For example:

Windows example:

```
c:\Program Files\**\*SQL*\bin\temp
```



Caution

The exclusion entry above means that any temp files in the bin folder are ignored by the restrictions you apply using this file set. This also means that the path you’re protecting in the Directories matching field is NOT protected when the excluded directory “temp” is being accessed.

UNIX example:

/etc/passwd

By default, this field has a <none> entry indicating no exceptions. When you click inside this field, the <none> disappears so that you can enter your own exceptions.

Step 8 Files Matching—Enter the names of the files to which you are controlling access.

You can use wildcards here to indicate all of a specific file type. For example, *.exe to specify all executables.

By default, this field has an <all> entry indicating all files. When you click inside this field, the <all> disappears so that you can enter your own file restrictions.

Step 9 but not—Make exceptions to the file names you enter in the Files Matching field. For example, all executables, but not regedit.exe.

By default, this field has a <none> entry indicating no exceptions. When you click inside this field, the <none> disappears so that you can enter your own exceptions.



Note

Use **@dynamic** in the File set text field to indicate all files that have been quarantined by CSA MC. This list updates automatically (dynamically) as logged quarantined files are received.

To view the files that are added to the dynamically quarantined files list and to manually add files to be quarantined, click the **Manage dynamically quarantined files** link on the Global Event Correlation page. See [Manage Dynamically Quarantined Files and IP Addresses, page 7-8](#) for more information.

Step 10 (UNIX only instruction) File Sets created for UNIX have an additional configuration field. In the **Attributes Matching** edit fields, click the **Insert attribute** link and optionally select one or more file types to match against. Available file types are as follows:

- block device—A special file used for buffered or block I/O. For example, a disk device.
- character device—A special file used for unbuffered or character I/O. For example, a tty file.
- executable file—A file identified in /etc/magic as being executable.

- interpreter file—A file which contains a script (shell, Perl, etc.) where the first line starts with “#! interpreter [arg]”.
- java class file—A file identified in /etc/magic as being executable Java byte code.
- setgid file—A file with the “set group ID on execution” property set in the file mode.
- setuid file—A file with the “set user ID on execution” property set in the file mode.

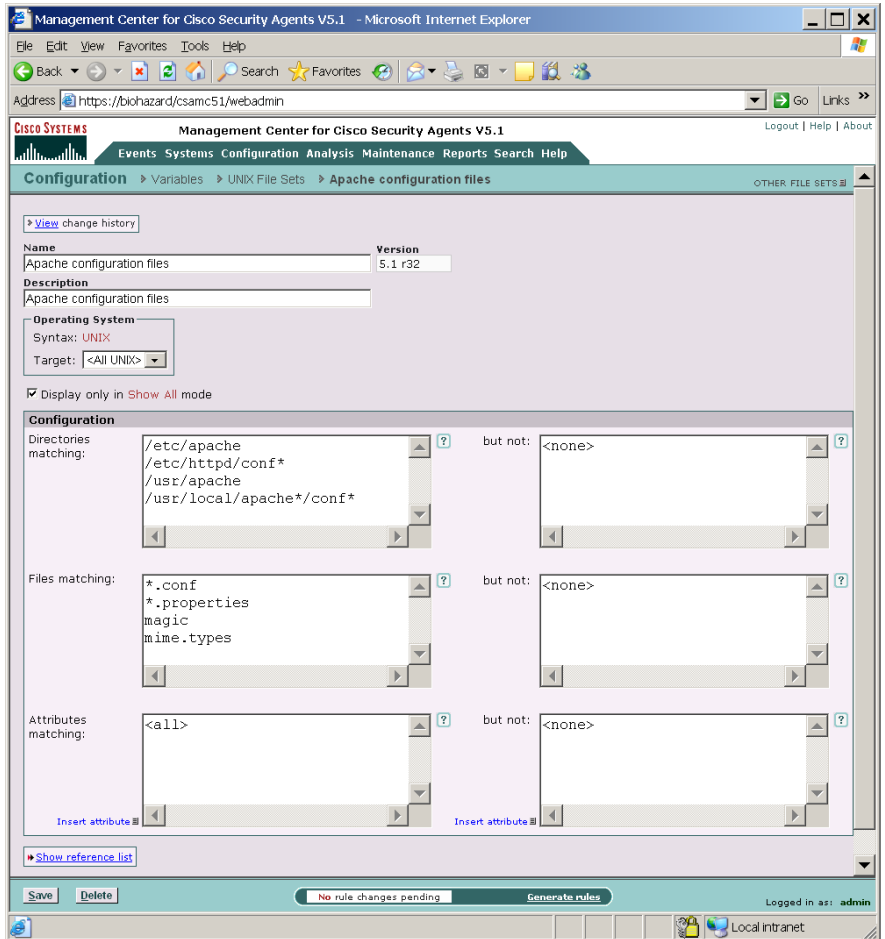
Step 11 When all required information is entered, click the **Save** button to save your file set in the CSA MC database.

You can now enter this file set name by clicking the **Insert File Set** link in the application class files field and in the file access control rule files field.



Note At the top of each variable page, there is a **View change history** link. Click this link to go to a page which lists all the changes that have been made to the item in question. This View change history link is also available for application classes, policies, and rules.

Figure 9-4 File Set Configuration View



153540

Network Address Sets

Configure network address sets for use in network access control rules to impose restrictions on specified IP addresses or a range of addresses. Once configured, you can simply enter the name of the address set in any network access control rules you create.

To configure network address sets, do the following.

-
- Step 1** From the menu bar **Configuration** drop-down list, move the mouse over **Variables**. A cascading menu with further selections appears.
 - Step 2** Select **Network Address Sets** from the cascading menu. Any existing address set configurations are shown.
 - Step 3** Click the **New** button to create a new network address set. This takes you to the configuration view (see [Figure 9-5](#)).
 - Step 4** In the available edit fields, enter the following information (See [Using the Correct Syntax, page 2-32](#). You can also click the Quick Help question mark beside each field for syntax information.):
 - **Name**—This is a unique name for this address set. When using configuration variables in file access rules, network access rules and application classes, you must enter the variable name preceded by a dollar sign:

For example, if you have a network address set variable named Finance systems, you must enter `$Finance systems` into any edit field where you are using this variable. The dollar sign tells CSA MC that this is a variable value.
 - **Description**—This is a useful line of text that is displayed in the list view and helps you to identify this particular set of addresses.
 - **Display only in Show All mode**—If your lists of variables are growing too long in rule or application configuration pages, clicking the Display only in Show All mode checkbox on a variable page causes that variable to no longer appear in selection lists for that variable type. This feature works in conjunction with Admin Preference settings. You must go the Admin Preferences page to make the item reappear. See [Configuring Role-Based Administration, page 2-5](#).
 - Step 5** **Address ranges matching**—In the available edit field, enter a single address, range of addresses, or network address class.

By default, this field has a <all> entry indicating no addresses. When you click inside this field, the <all> disappears so that you can enter your own addresses. When entering directory restrictions, use the following syntax:

Put each entry on its own line. For address ranges, use a hyphen to indicate the range. Address ranges are inclusive.

For example: 128.66.24.130

128.67.2.10-20 or 10.0.0.0/24

Use **@local** to indicate all local addresses on the agent system. You would want to use this if you are allowing different applications on a single system to talk to each other without opening these applications up to network access.



Note

Use **@dynamic** in the Addresses set field to indicate untrusted hosts that have been quarantined by CSA MC. Addresses are added to this list when they are seen as an untrusted host. (The built-in “Processes Communicating with Untrusted Hosts” is triggered in a rule.) This list updates automatically (dynamically) as logged quarantined addresses are received.

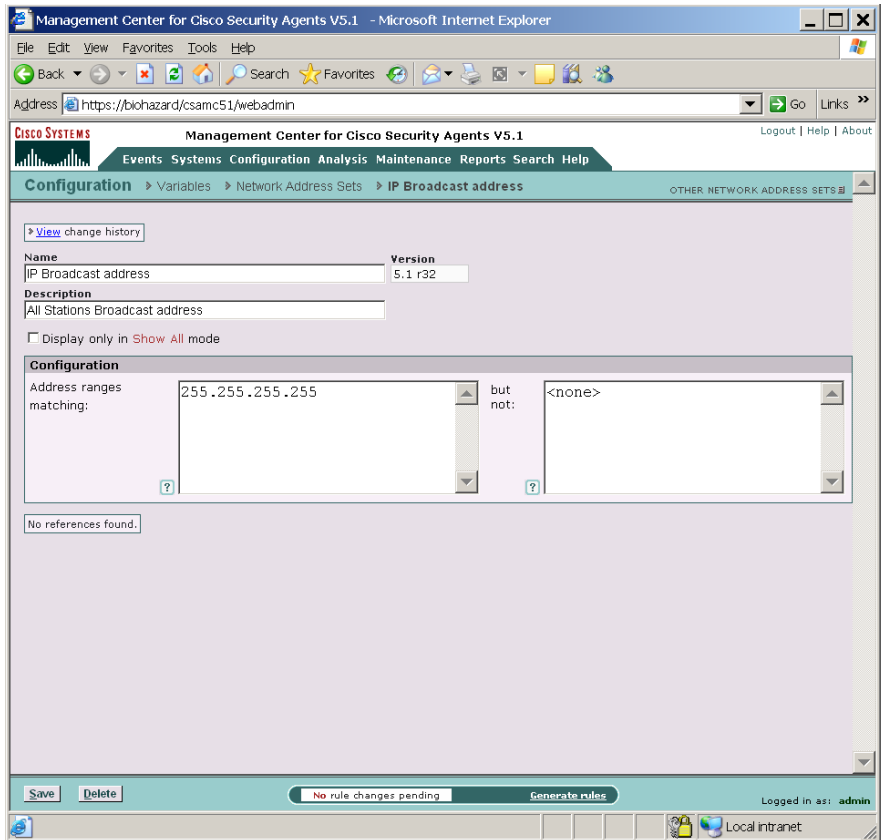


Caution

On UNIX platforms, IPV6 addresses are not officially supported; however, an IPV6 connection will work as the applied rules dictate if the address in question is covered by the "all" addresses range (0.0.0.0-255.255.255.255 includes IPV6 addresses) or by @local. Local addresses on the agent system (indicated by @local) also include IPV6 addresses.

but not—Use this field to make exclusions to addresses within the address ranges entered in the address matching field.

Figure 9-5 Network Address Set Configuration View



- Step 6** When all required information is entered, click the **Save** button to save your address set in the CSA MC database.



Note You can now enter this network address set name by clicking the **Insert Network Address Set** in the network access control rule host addresses field.

Network Services

Configure network services for use in network access control rules to add preconfigured protocol and port number restrictions. You can restrict by initial connection ports, and when applicable, by subsequent client/server connection.

CSA MC ships with several pre-configured network services you can use.

To configure network services, do the following.

-
- Step 1** From the menu bar **Configuration** drop-down list, move the mouse over **Variables**. A cascading menu with further selections appears.
 - Step 2** Select **Network Services** from the cascading menu. Any existing configurations are shown.
 - Step 3** Click the **New** button to create a new network service variable. This takes you to the configuration view ([Figure 9-6](#)).
 - Step 4** In the available edit fields, enter the following information (See [Using the Correct Syntax, page 2-32](#). You can also click the Quick Help question mark beside each field for syntax information.):

- **Name**—This is a unique name for this network service configuration. This name is case insensitive. Generally, it's a good idea to adopt a naming convention that lets you quickly enter network service variables in network access control rule configuration fields. When using configuration Variables in file access rules, network access rules and application classes, you must enter the variable name preceded by a dollar sign.

For example, if you have a network service variable named FTP Service, you must enter `$FTP Service` into any edit field where you are using this variable. The dollar sign tells CSA MC that this is a variable value.

- **Description**—This is a useful line of text that is displayed in the list view and helps you to identify this particular configuration.
- **Display only in Show All mode**—If your lists of variables are growing too long in rule or application configuration pages, clicking the Display only in Show All mode checkbox on a variable page causes that variable to no longer appear in selection lists for that variable type. This feature works in conjunction with Admin Preference settings. You must go the Admin Preferences page to make the item reappear. See [Configuring Role-Based Administration, page 2-5](#).

Step 5 Protocol ports-Destination—Enter a tcp or udp protocol and corresponding port or port range to indicate a restriction.

By default, this field has a <all> entry indicating no ports. When you click inside the edit field, the <all> disappears so that you can enter your own port restrictions.

Use the following syntax:

```
TCP/21
UDP/1025-65535
```

Protocol ports-Source—(**CAUTION:** Using a specific source port, rather than the default of <all> , in a Network access control rule may degrade performance.) You can enter a tcp or udp protocol and corresponding port or port range to indicate a restriction if it is necessary. Generally, you will not want specify a specific port here and simply leave <all> as the entry for the source port. You only want to enumerate a specific source port for a data connection that has an ephemeral destination port and a well-known source port.

Since most network connections are keyed off of well-known destination ports, applications that only have well-known source ports, such a multimedia applications or Active FTP data connections, must be controlled off the source port. Therefore, if you are specifying a differentiated service marking for a multimedia connection, you would key off the source port.



Note

Some protocols, such as ftp, create additional connections as part of the same session started by the initial connection. The port numbers used for these additional connections must be defined as another Network Service and used appropriately in a rule module to consider callback connections. When a network service is used in an allow rule, once an initial connection is established, the subsequent connections will also be allowed, but only to the process that participated in the initial connection.

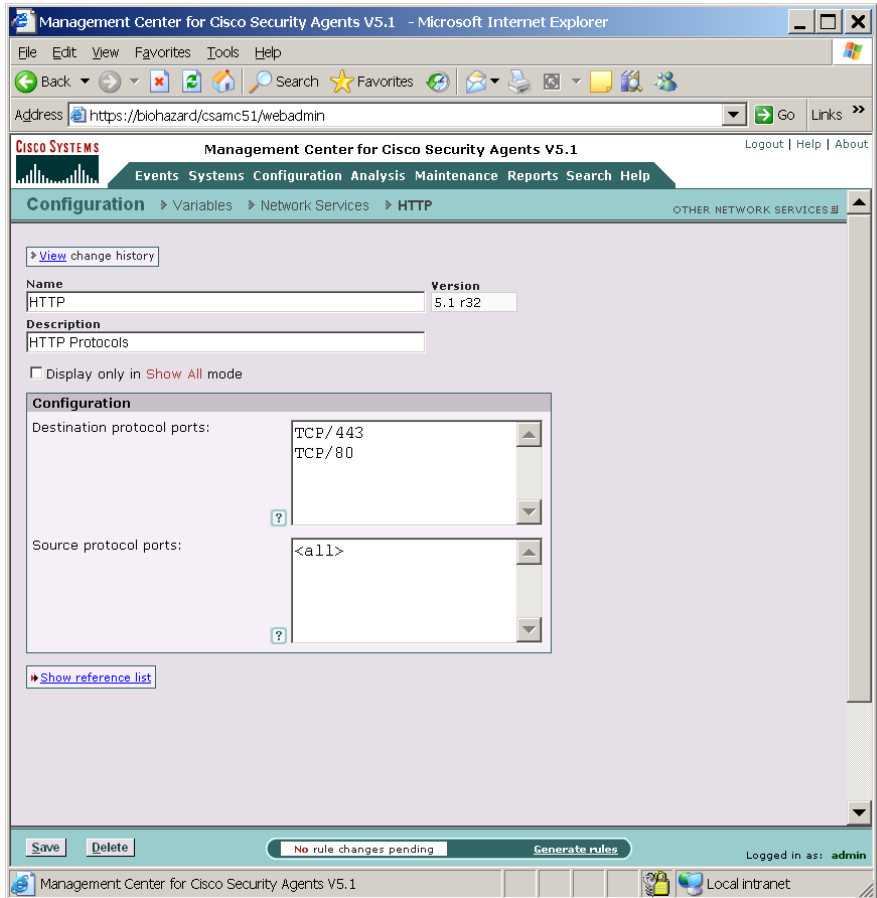
In some cases, an application may want to offer a temporary service port for callback data connections. An ephemeral port is a temporary system-assigned port for this purpose. You can specify an ephemeral port range for a Network service as follows (See [Using the Correct Syntax, page 2-32](#) for more details):

```
TCP/ephemeral
UDP/ephemeral
```

Step 6 When all required information is entered, click the **Save** button to save your event set in the CSA MC database.

You can now enter this network service name by clicking the **Insert Network Service** link in the network access control rule network services field.

Figure 9-6 Network Services Configuration View



153542

Registry Sets

A variety of viruses invoke themselves using registry settings. Use the preconfigured registry sets in registry access control rules to prevent viruses from writing to registry values popular with viruses.

This variable is not available for UNIX configurations.



Caution

If you attempt to create your own registry sets to include in a rule, you should note that the ability to restrict registry access is an extremely powerful tool. Critical applications may not function as a result of a misconfigured registry restriction. Therefore, registry values should be as specific as possible. All rules restricting registry access should first be run in **Test Mode** to ensure that no unintended restrictions have been configured.

Registry sets are groupings of registry keys and settings under one common name. This name is then used in rules that allow or deny registry write operations. All the registry restriction parameters that exist under that name are then applied to the rule where the name is used.

To view preconfigured registry sets or to create a new registry set, do the following.

-
- Step 1** From the menu bar **Configuration** drop-down list, move the mouse over **Variables**. A cascading menu with further selections appears.
 - Step 2** Select **Registry Sets** from the cascading menu. Any existing registry set configurations are shown.
 - Step 3** To view an existing registry set, click the link for that item. Click the **New** button if you would like to create a new registry set variable. This takes you to the configuration view (see [Figure 9-7](#)).
 - Step 4** Enter the following.
 - **Name**—This is a unique name for this registry set.
 - **Description**—This is a line of text that is displayed in the list view helping you to identify this particular registry set configuration.
 - **Display only in Show All mode**—If your lists of variables are growing too long in rule or application configuration pages, clicking the Display only in Show All mode checkbox on a variable page causes that variable to no longer

appear in selection lists for that variable type. This feature works in conjunction with Admin Preference settings. You must go the Admin Preferences page to make the item reappear. See [Configuring Role-Based Administration, page 2-5](#).

Step 5 Registry keys matching—You *must* enter a value in this field if you are creating a registry set.

It is recommended that there be at least one non-wildcarded component in a registry key other than the hive itself. Otherwise, the specified key might be overly generalized.

Hives are one of the following strings:

HKLM—refers to the HKEY_LOCAL_MACHINE

HKCR—refers to HKEY_CLASSES_ROOT

HKCC—refers to HKEY_CURRENT_CONFIG

HKU—refers to HKEY_USERS (HKU* refers to all users)

Table 9-1 Example valid and invalid registry key entries

| | |
|-------------------------|---|
| **\MSSQLSERVER** | This is a valid entry. |
| **\M*** | This is not a recommended entry because there is no non-wildcard component. (M* is wildcarded.) |
| HKLM\SOFTWARE\CSCOpX** | This is a valid entry. |
| FOO\SOFTWARE\CSCOpX** | This is invalid (FOO is not a hive). |



Note

Note that the wildcard syntax explained in [Table 2-2 on page 2-34](#) also applies to registry sets. However, the asterisk is a valid single character in a registry key and should be represented with a single “?” wildcard.

Step 6 but not—Make exceptions to registry keys.

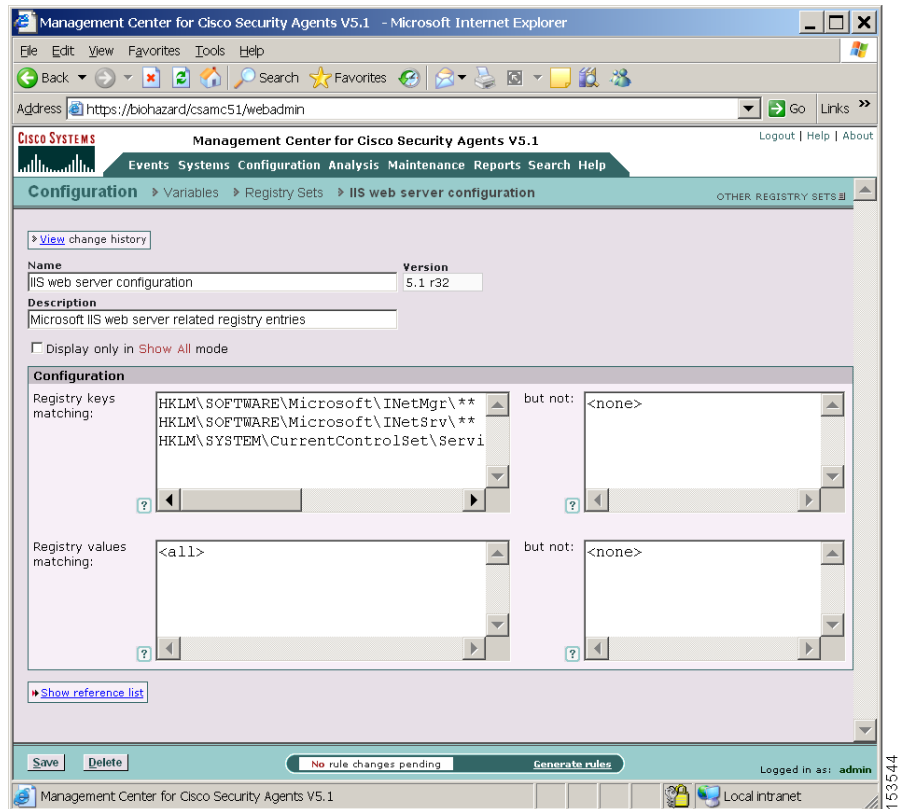
Step 7 Registry values matching—Enter the registry values you are controlling access to.

Step 8 but not—Make exceptions to registry values.

Step 9 When all required information is entered, click the **Save** button to save your registry set in the CSA MC database.

You can enter this registry set name by clicking the **Insert Registry Set** link in the registry access control rule registry entries field.

Figure 9-7 Registry Set Configuration View



Included Registry Sets

CSA MC ships with several pre-configured registry sets you can use in your registry access rules. Some are application specific, others are operating system specific. This section describes a sample of the included operating system specific registry keys.

- Run Keys are used to register programs so that the system will invoke them as a service. Viruses can make use of this key to become persistent.

Protecting this registry value by creating a rule to prevent writing to run keys can prevent the type of virus described above from invoking and propagating itself.

**Note**

It is important to note that if users have administrator privileges on their systems and are installing software, this type of rule may trigger and prevent that installation. In such cases, using a Query User rule would be most effective. This way, if users are installing software, they themselves can prevent the agent from stopping the installation by answering "Yes" to the query to allow the install. However, if users are not installing software, this type of Query User rule triggering on a system could be treated as a serious issue and the user should answer "No to all" to disallow the action.

- Shell commands are used to tell your system how to open a file based on the file format. This is how the system knows which application to use when opening a particular file.

Viruses can exploit this by having the registry setting invoke the virus along with the application being opened. In this case, the application would open correctly and the virus could silently begin doing harm.

BootExecute tells the system which executables should be run at system startup time.

- Reboot operations tell the system which operations should begin at system startup time. If programs have been uninstalled, the reboot operation also tells the system which files and services should be deleted on the next reboot and startup.

Viruses can exploit this registry setting by marking particular files for copying, overwriting, or deleting on startup. For example, a virus may attempt to delete a system service that could possibly detect the virus itself. By deleting this service at startup, the virus can go undetected.

**Note**

It is important to note that if users have administrator privileges on their systems and are uninstalling software, this type of rule may trigger and prevent the uninstall. In such cases, using a Query User rule would be most effective. This way, if users are uninstalling software, they themselves can prevent the agent from stopping the uninstall by answering "Yes" to the query to allow the action.

However, if users are not uninstalling software, this type of Query User rule triggering on a system could be treated as a serious issue and the user should answer "No to all" to disallow the action.

Query Settings

From the Query Settings page, accessible from the **Configuration>Variables>Query Settings** menu, you configure the query text and the query buttons that appear in the pop-up box the end user will see when query rules are triggered.



Note

For a Query setting, the response to the query is relevant to the question, not to the resource. For example, if a File access control rule queries the user for a response and that identical query is also configured for a Network access control rule, the user is not queried again when the Network access control rule triggers. The query response from the previous File access control rule is automatically taken.

To configure a query pop-up box for use with a query rule, do the following:

- Step 1** On the Query Settings list page, click the **New** button to create a new query. See [Figure 9-8](#).



Note

CSA MC ships with several preconfigured queries. You can use an existing one or create a new one.

- Step 2** Enter a unique **Name** for your query. Names are case insensitive, must start with an alphabetic character, can be up to 64 characters long and can include hyphens and underscores `_`. Spaces are also allowed in names. Use a descriptive name that you can easily recognize in the rule selection box when you are selecting a specific query setting for a rule.
- Step 3** Enter a **Description** of your query.
- Step 4** **Display only in Show All mode**—If your lists of variables are growing too long in rule or application configuration pages, clicking the Display only in Show All mode checkbox on a variable page causes that variable to no longer appear in

selection lists for that variable type. This feature works in conjunction with Admin Preference settings. You must go the Admin Preferences page to make the item reappear. See [Configuring Role-Based Administration, page 2-5](#).

- Step 5** In the **Text used to query user** edit field, enter a description of the issue that likely triggered the query. This text field allows you to provide localized query text for agents using the corresponding language on their desktop. This is the same text that will appear in the query user pop-up box explaining what is occurring on the system to the user. Therefore, making this information descriptive of the system action that triggered the pop-up is important.

You can use specially designated tokens to represent the corresponding values presented to the end user who is responding to the query. See [Using Query Tokens, page 9-28](#).



Note

All Cisco Security Agent kits contain localized support for Spanish, French, German, Italian, Japanese, Korean, and Simplified Chinese language desktops. If you do not select a specific language, the default for query text is English. Click the **More languages** link to enter text to be displayed in a language other than English. This allows you to provide localized query text for agents using the corresponding language on their desktop. See [Localized Language Version Support, page 9-30](#) for more details.

- Step 6** The Allowed query actions multi-select box lets you choose which radio buttons appear on the query pop-up box. For example, you may not want the user to have a “Terminate” option. Therefore, you would only select the Allow and Deny radio buttons to be displayed.

The user reads the information posted on the query and is given the choice to select one of the following possible choices and click Apply:

- **Allow** (Yes)—Allows the application access to the resource in question.
- **Deny** (No)—Denies the application access to the resource in question.
- **Terminate**—Denies the application access to the resource in question and also attempts to terminate the application process. (Some processes cannot be safely terminated, such as winlogon.)

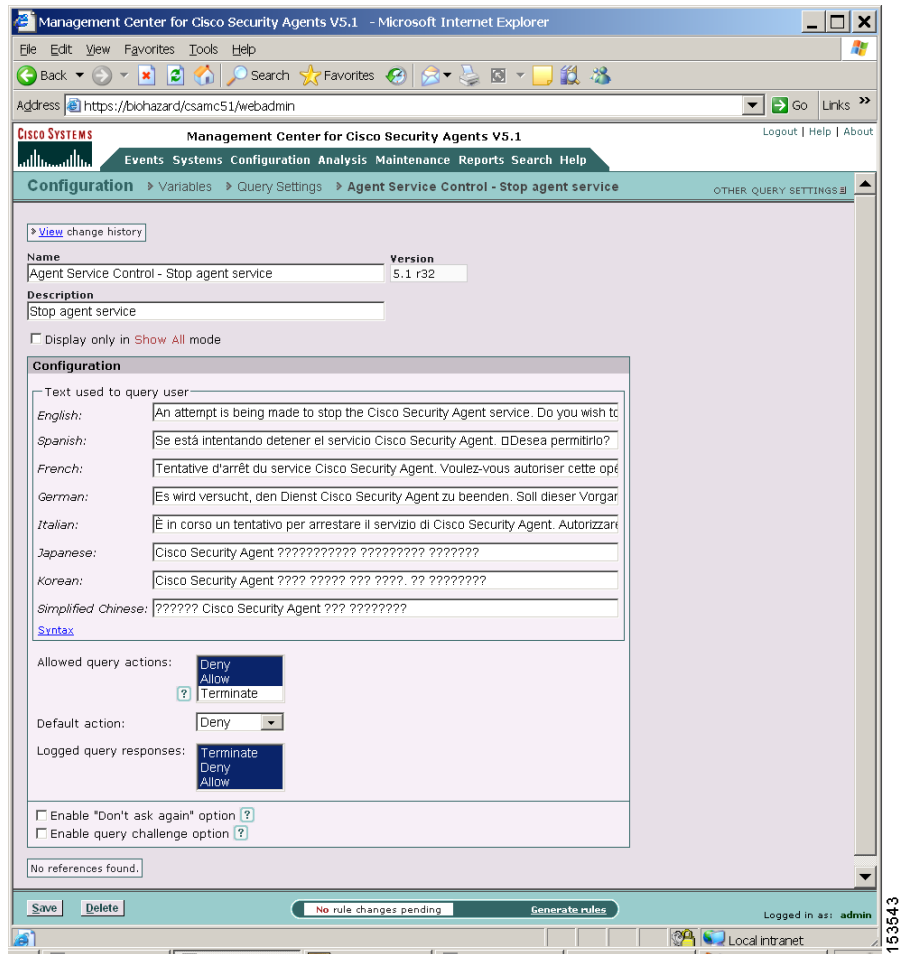
- Step 7** Of the radio buttons you decided to display, you also choose one of those buttons to be the **Default action**. If the query is not answered by the user within 5 minutes or if the user is not logged in to the system, the default action is taken immediately.

- Step 8** In addition to deciding which query actions (Allow, Deny, Terminate) are available to the user for the query pop-up, you can also configure the query response to log only when a particular query action is selected by the user. Using the multi-select box available from the **Logged query responses** section, you can select one or more response types to produce a log message. For example, if all query actions are being made available for the query, you can configure only a Terminate response to produce a log message. (By default, all query responses are logged.)
- Step 9** You can also decide to display a **Don't ask again** checkbox so that the user's query response is remembered. If the user selects that checkbox when he/she responds to the query, and the same action is attempted on the same resource, the remembered response is automatically taken and the user is not queried again.
- Step 10** For added security, you can issue a **query challenge** on the query pop-up box. If the default answer is not selected by the user and the selected answer is weaker than the default, a challenge will appear. This ensures that the user sitting in front of the system is answering the query rather than a malicious remote user or program attempting to respond. To pass the challenge, the user enters the information displayed in a graphic on the pop-up box itself.
- Step 11** Click the **Save** button.

**Tip**

When you phrase the question that will appear to users and select the radio button options to be displayed, make sure that the logic you use is in sync with the response the user should select. For example, you probably should not phrase a question in the following way: “Do you want to prevent this action from occurring?” In this case, if the response is “Yes”, this is counterintuitive to how queries should be used. The user is selecting Yes to indicate No. Instead, phrase the question as follows: “Select No to prevent this action from occurring.”

Figure 9-8 Query Settings Configuration View



153543

Using Query Tokens

When entering query text into the edit field, you can use the following tokens to represent the values presented to the end user who is responding to the query.

- @parent—The path of the parent process. Use in Application control rules only.

- @ActiveXname—The name of the ActiveX control being downloaded. Use in System API access control rules only.
- @appname—The path of the process triggering the action. Use in all access control rule types, except Application control rules.
- @child—The path of the process being invoked. Use in Application control rules only.
- @progid—The ProgID of the COM object. Use in COM component access control rules only.
- @clsid—The GUID of the COM object. Use in COM component access control rules only.
- @dataname—The name of data being filtered. Use in Data access control rules only.
- @filename—The full file path of the file being accessed. Use in File access control rules only.
- @fileop—The type of file operation (file/directory, read/write). Use in File access control rules only.
- @funcname—The system API function being called. Use in System API access control rules only.
- @hostaddr—The remote address of a connection. Use in Network access control rules only.
- @localaddr—The local address of a connection. Use in Network access control rules only.
- @netop—The type of network operation (client/server). Use in Network access control rules only.
- @netservice—The service/destination port used by the remote connection end. Use in Network access control rules only.
- @regname—The registry entry being accessed. Use in Registry access control rules only.
- @targetapp—The path of the application being targeted for code injection or modification. Use in System API access control rules only.

Localized Language Version Support

On systems running multiple locales, (for example, Multilingual User Interface installations or Terminal Services), queries are displayed in the supported language used for the Windows desktop on which the query is shown. Events appear in the Windows Event Log in the default systems language.

For example, on a Windows 2000 Multilingual User Interface (MUI) installation, if a user is running a Japanese language version desktop, queries will appear in Japanese. But the Windows Event Log on this system will store events formatted in US English because the system language on a Windows MUI system is English.

On a localized Japanese system, both the queries and the events appearing in the Windows Event Log appear in Japanese.