



Available Rule Types

Overview

Rule modules contain various types of rules. Each rule type is intended to control a different set of system resources. For example, there is a rule type that controls access to files and directories and another rule type that controls network accesses. It is through the combining of the many available rule types that a rule module provides overall security to the entire system. This chapter provides information on each rule type.

This section contains the following topics.

- [Rules Common to Windows and UNIX, page 6-3](#)
- [Agent Service Control, page 6-3](#)
- [Agent UI Control, page 6-7](#)
- [Application Control, page 6-11](#)
- [Connection Rate Limit, page 6-15](#)
- [Data Access Control, page 6-19](#)
- [File Access Control, page 6-23](#)
- [Network Access Control, page 6-28](#)
- [Network Shield Rule, page 6-32](#)
- [System API Control Rule, page 6-40](#)
- [Windows Only Rules, page 6-44](#)

- [Clipboard Access Control](#), page 6-44
- [COM Component Access Control](#), page 6-46
- [File Version Control](#), page 6-49
- [Kernel Protection](#), page 6-54
- [NT Event Log](#), page 6-58
- [Registry Access Control](#), page 6-61
- [Service Restart](#), page 6-64
- [Sniffer and Protocol Detection](#), page 6-67
- [UNIX Only Rules](#), page 6-70
- [Network Interface Control](#), page 6-75
- [Resource Access Control](#), page 6-78
- [Rootkit / kernel Protection](#), page 6-80
- [Syslog Control](#), page 6-84

Rules Common to Windows and UNIX

The following rule types are available for both Windows and UNIX policies.

Agent Service Control

Use the Agent service control rule to control whether administrators are allowed to stop agent security (This is via a net stop command on Windows or via `/etc/init.d/ciscosec stop` on UNIX. See [Chapter 12, “Using Management Center for Cisco Security Agents Utilities,”](#) for details) and whether end users can disable security via the agent UI security slide bar. Stopping agent security disables all rules until security is manually resumed or the system is rebooted.

If you use this rule to deny agent service stops, the agent service cannot be stopped on the system in question and therefore agents cannot be uninstalled.

**Note**

Although agents cannot be uninstalled by administrative users if this rule denies the stopping of the agent service, this rule does not prevent agent software updates from occurring.

You can also use this rule to monitor, terminate, or tag a process that attempts to modify the agent configuration.

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

-
- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Agent service control** rule. This takes you to the configuration view for this rule type (see [Figure 6-1](#)).

- Step 3** In the Agent service control rule configuration view, enter the following information:
- **Description**—Enter a description of this rule. This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.) By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.
- Step 4** **Take the following action**—(Note that not all action types are available for this rule on Windows platforms.) Select an action type from the pulldown list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#).
- Step 5** and
- **Log**—Enable this checkbox to turn logging on for this rule. Generally, you will want to turn logging on for all deny rules. This means that the denied system action in question is logged and sent to the server at regular time intervals.
 - **Take precedence over other <action type> rules**—Enable this checkbox to manipulate rule precedence so that this rule is evaluated before other similar rules. You should generally NOT require this checkbox. Do not use it without understanding how it works. See [Rules: Manipulating Precedence, page 5-41](#) for details.
- Step 6** **when**
- Applications in any of the following selected classes**
- Select *one or more* preconfigured application classes. Note that the entry, <All Applications>, is selected by default. You can use this default or you can unselect it and create your own application classes. For application class configuration details, see [Chapter 8, “Using Application Classes”](#).



Note On UNIX systems, anyone with root access can stop the agent service. To prevent this, while still allowing administrators to stop the agent service, you would configure an Agent service control rule to Deny <All Applications> from stopping the service. Then configure another Agent service control rule which Allows only a UNIX Secured Management application class to stop the service.

But not in the following class—Optionally, select application classes here to exclude from the application class(es) you've selected in the included applications field. Note that the entry <None> is selected by default.

- **attempt to disable agent security**

This checkbox controls whether users with administrator privileges can stop the agent service from the Service Control Manager or by running `net stop "Cisco Security Agent"` from a command prompt on Windows or via `/etc/init.d/cisecsec stop` on UNIX.



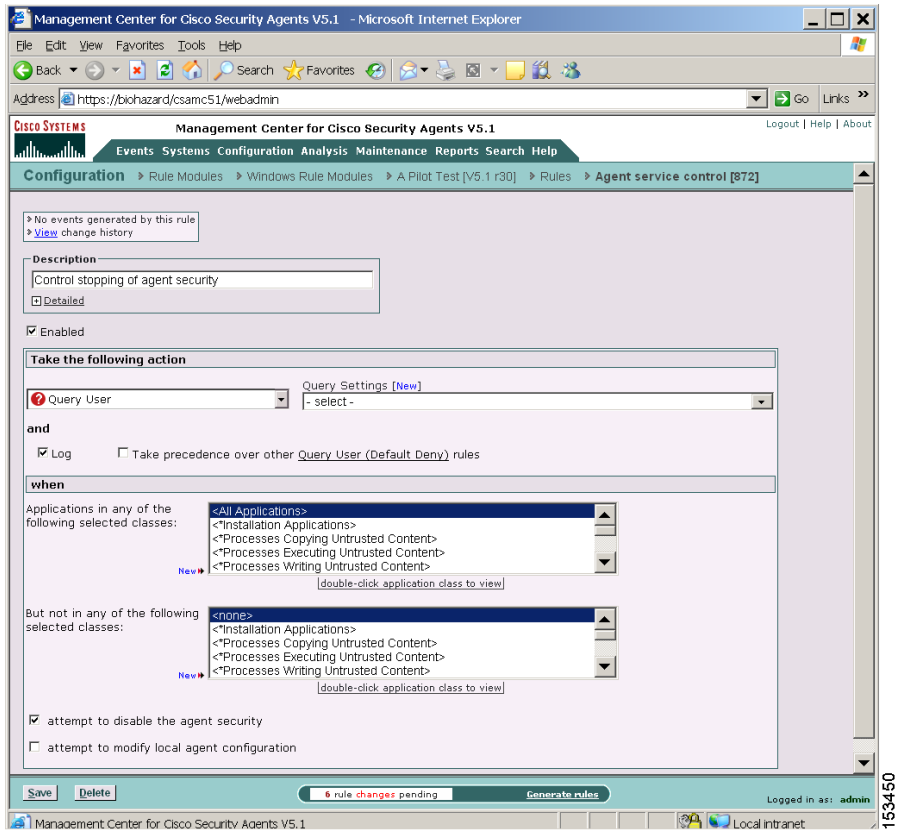
Note This also controls whether the "Off" setting on the agent security level sidebar allows the end user to turn agent security off. If you do not allow the stopping of the agent service, the Off level, if available, is ineffective. See [Agent UI Control, page 6-7](#) for more information.

- **attempt to modify local agent configuration**

The Cisco Security Agent has built-in global security policies which protect agent binaries and data. (Note that this protection is only offered when the agent service is running and is not stopped or in Test Mode.) While you cannot turn these non-logged, built-in rules off while the agent is active, you can use this rule to monitor, terminate, or tag a process that attempts to modify the agent configuration.

Step 7 Click the **Save** button.

Figure 6-1 Agent Service Control Rule (Windows)



153450

Agent UI Control

**Note**

This rule only applies to Windows and Linux platforms. The agent UI is not supported on Solaris systems.

Also note that Test Mode does not apply to this rule type.

Use the Agent UI rule to control how the agent user interface is displayed to end users. See [Figure 6-2](#). In the absence of this rule, end users have no visible agent UI. If this rule is present in a module, you can select to display the agent UI and one or more controls to the end user. These controls give the user the ability to change certain aspects of their agent security. Optional controls are as follows:

- **Allow user to reset agent UI default settings**—On Windows, this is available from the **Start>Programs>Cisco Systems** menu. On Linux, this is available from **System Menu>Cisco Security Agent**. By selecting this option, users can reset agent UI functionality to the original factory default settings. All user set controls are lost and all persistent query responses are removed. This is useful on Windows platforms where different users with varying user agent permission settings may log into the same machine.
- **Allow user interaction**—Selecting this checkbox causes the end user to have a visible and accessible agent UI, including a red flag in the system tray. With no other subsequent checkboxes selected, the agent UI contains a status view, a messages page to view agent events, and the ability to clear persistent and temporary query user responses. (If this rule is present in a module, but this checkbox is not selected, the end user will have no visible agent UI. In the presence of two or more Agent UI control rules, these rules are combined and selected checkboxes take precedence over unselected checkboxes.)

Add one or more additional controls as follows:

- **Allow user access to agent configuration and contact information:** Selecting this checkbox allows the end user to enter Contact information into the agent UI. They also have access to a Poll button which allows them to force a manual polling of the MC.
- **Allow user to modify agent security settings:** Selecting this checkbox provides the end user with the ability to alter their security level by moving a sidebar between Off, Low, Medium, and High (in accordance with policies) and to manage the classification of untrusted content.

This checkbox provides an Off control on the agent UI. This Off control works in combination with the Agent service control rule (see [Agent Service Control, page 6-3](#)). You must both provide this sidebar to the end user and have an Agent service control rule in place which allows the agent security to be disabled in order for the Off setting to actually turn security off.

Allowing this action (moving the sidebar to Off) permits all users (including non-administrative users) to disable all rules on the agent until they are re-enabled by the user. (Note that if there is no agent UI present, agent security cannot be turned off.)

- **Allow user to modify agent personal firewall settings**—Selecting this checkbox provides the end user with the ability to dictate which applications are allowed network access. They also gain a file protection capability by which they can enter the names of local files that network applications are not allowed to access on their system. Note that if a user is allowed to configure personal firewall settings, resource access attempts on the system must pass both policy rules and firewall settings. (If you select this checkbox, you are providing the end user with controls that you have limited access to. Firewall queries and other information will not log to the CSA MC event log.)

Hiding the agent UI

Not enabling the **Allow user interaction** checkbox in this rule has the following effects.

Software updates

There are no effects. Hiding the agent UI and Software updates are independent features. You can provide a software update prompt when an agent UI is not present.

Queries

When there is no agent UI present, there are no query user pop-up boxes displayed. The default is immediately taken on all query user rules and heuristics that are present in the assigned policies. (Note that this does not apply to cases where the end user manually exits the agent UI. Only the administrator controlled agent UI rule can affect query pop-up displays on the end user system.)

Unavailable end user features

- No messages to inform user that actions have been denied and why.
- No ability to clear cache or re-enable logging.
- No fast polling ability.
- No end user contact information can be sent to CSA MC.

Hidden agent UI feature notes

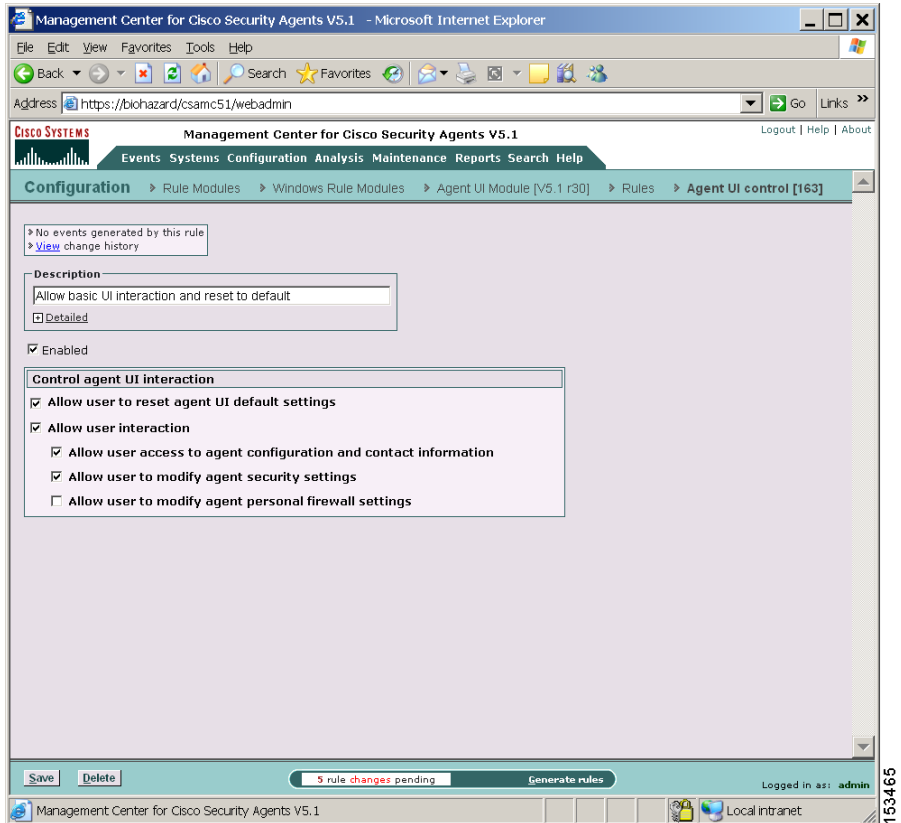
If a host belongs to multiple groups with multiple policies, having a visible agent UI setting, if present in any group for which the host is a member, takes precedence over a no user interaction agent UI setting.

Whether or not an end user system is going to have a visible agent UI or a hidden one, the end user (or administrator) must download and install the agent kit on the system. The initial installation of an agent kit cannot be done automatically (unless you have written your own script to do so, see [Scripted Agent Installs and Uninstalls, page 3-21](#)).

When there is no agent UI present, there are no query user pop-up boxes displayed. The default is immediately taken on all query user rules and heuristics that are present in the assigned policies.

If an end user system already has an agent UI installed, when you unselect the **Allow user interaction** checkbox and generate rules, the agent UI disappears when the new rules are downloaded.

Figure 6-2 Agent UI Control Rule



Application Control

Use Application control rules to control what applications can run on designated agent systems. This rule type does not control what application can access what resources as do other access control rules. This rule type can stop selected applications from running on systems. If you deny an application class (in total) in this rule, users cannot use any application in that class.

With this rule, you can also prevent an application from running only if that application was invoked by another application you specify. This way, you could prevent a command prompt from running on a system if it is invoked by an application that has downloaded content from the network.

**Note**

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

-
- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Application Control** rule. This takes you to the configuration view for this rule type (see [Figure 6-3](#)).
- Step 3** In the Application control rule configuration view, enter the following information:
- **Description**—Enter a description of this rule. This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.) By not selecting this checkbox, you can save this rule, but it will not be active in the policy and it will not be distributed to groups.
- Step 4** **Take the following action**—Select an action type from the pulldown list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#).



Note Creating dynamic application classes from the Application control rule is a bit different than creating them from other rule types. Because this rule has two application class fields, you can choose to add the current application to the dynamic class or choose to add the new application that is invoked by the first application to the dynamic class.

Step 5 and

- **Log**—Enable this checkbox to turn logging on for this rule. Generally, you will want to turn logging on for all deny rules. This means that the denied system action in question is logged and sent to the server at regular time intervals.
- **Take precedence over other <action type> rules**—Enable this checkbox to manipulate rule precedence so that this rule is evaluated before other similar rules. You should generally NOT require this checkbox. Do not use it without understanding how it works. See [Rules: Manipulating Precedence, page 5-41](#) for details.

Step 6 when

- **Current applications in any of the following selected classes**—If you want to control an application (allow or deny) running on a system no matter how it is invoked, allow "All Applications" to remain selected by default. (Then you will select the application you want to control from the second Application class list.)

If you want to control which application(s) can invoke other applications, select one or more preconfigured application classes here to indicate the application that is doing the invoking (such as Network Applications).

(When your rule is configured, currently selected application classes appear at the top of the list. See [Configuring Static Application Classes, page 8-8](#) for configuration details.)

- **But not in the following class**—Optionally, select application classes here to exclude from the application class(es) you've selected in the included applications field. Note that the entry <None> is selected by default.

attempt to run

- **New applications in any of the following selected classes**—If you are controlling which applications can invoke other applications, this second field indicates the application class that you do not want to run when invoked by the application you chose in the top field.

If you selected "All Applications" in the top application field, you cannot select All Applications in this second field. If you did so, all applications would be completely prevented from running on systems if this is a deny rule.

- **But not in the following class**—Optionally, select application classes here to exclude from the application class(es) you've selected in the included applications field. Note that the entry <None> is selected by default.

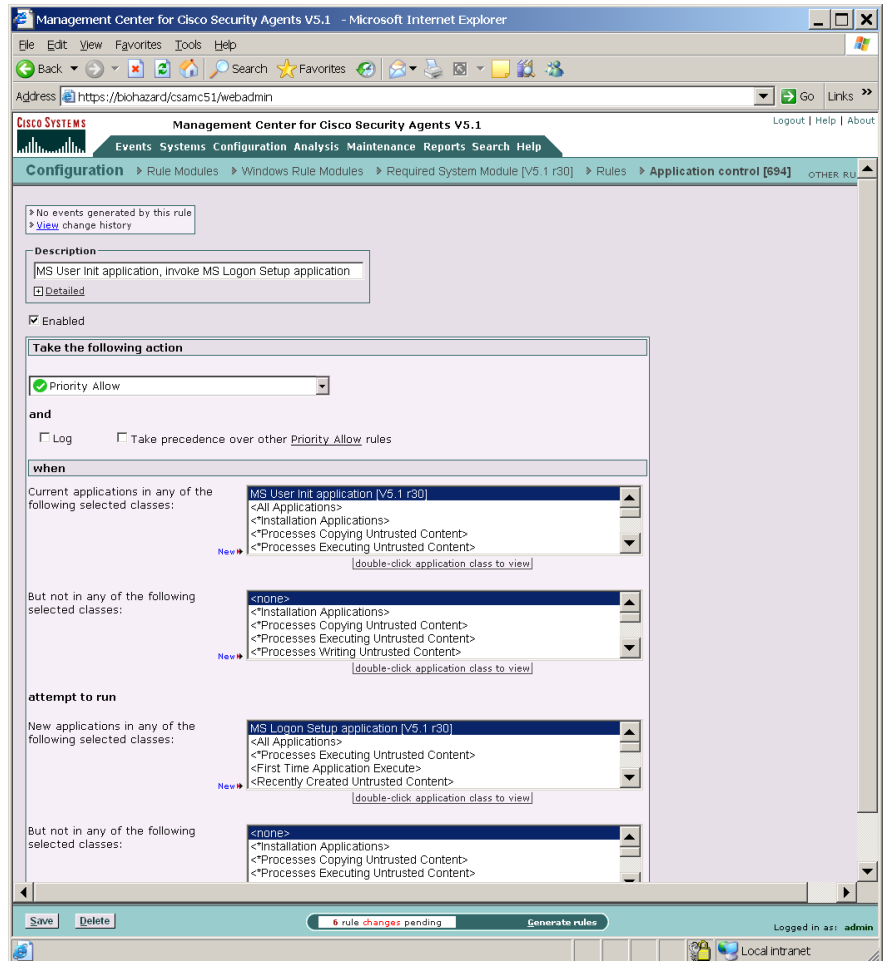
**Note**

Most dynamic application classes are not available in this second application class inclusion field.

Step 7

When you are finished configuring your Application control rule, click the **Save** button.

Figure 6-3 Application Control Rule



Connection Rate Limit

Use the connection rate limit rule to control the number of network connections that can be sent or received by applications within a specified time frame. This is useful in preventing attacks aimed at bringing down system services, e.g. denial of service attacks (server connection rating limiting). This is also useful in preventing the propagation of denial of service attacks (client connection rate limiting).

**Note**


Multiple instances of the same application are counted together with respect to this rule. E.g. If a machine has several instances of Apache web server running, all Apache connections are counted together when applying this rule.

**Note**

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

Click the **Modify rules** link at the top of the Rule Module page to go to the Rules page.

-
- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Connection rate limit** rule. This takes you to the configuration view for this rule type.
- Step 3** Enter the following information for the rule:
- **Description**—Enter a description of this rule.
This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.)
By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.
 - **Log**—Use this checkbox to enable logging within the module.

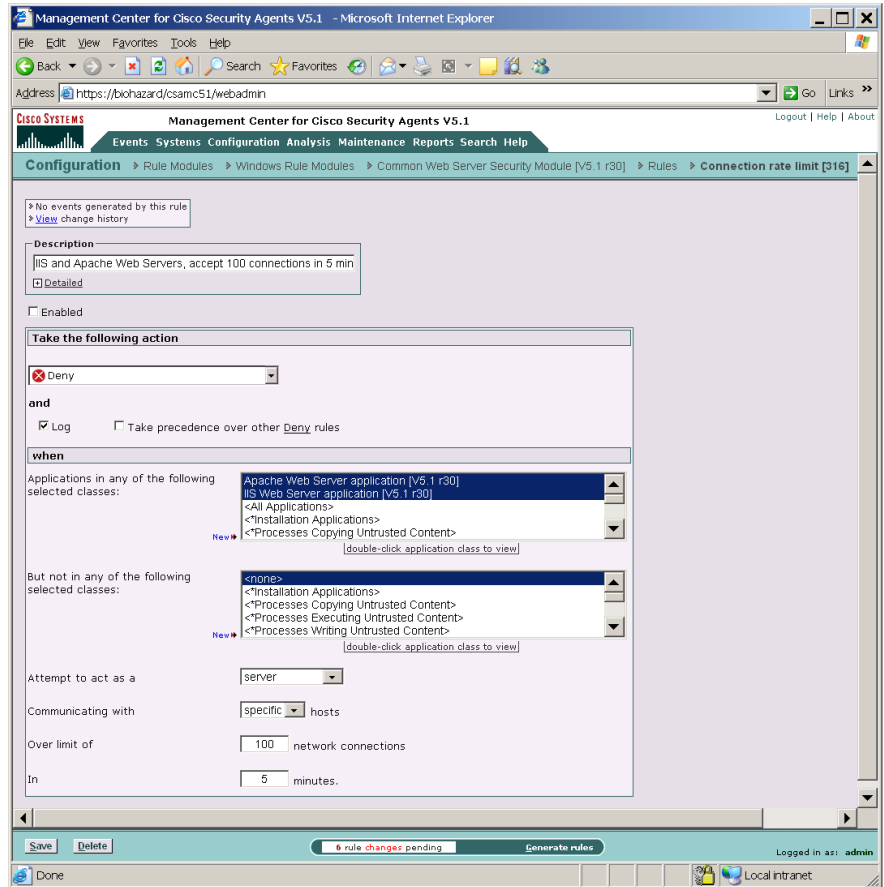
- Step 4 Take the following action**—Select an action type from the pulldown list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#). (Note that you cannot configure Query User Connection rate limit rules. Also note that if you select the Set action for marking a host as untrusted, you can only configure the rule for servers and for specific hosts.)
- Step 5 When Applications in any of the following selected classes**
- Select *one or more* preconfigured application classes here to indicate the application(s) whose connection rate access you want to exercise control over.
- Note that the entry, <All Applications>, is selected by default. You can use this default or you can unselect it and create your own application classes. For application class configuration details, see [Chapter 8, “Using Application Classes”](#).
- But not in the following class**—Optionally, selection application classes here to exclude from the application class(es) you’ve selected in the included applications field. Note that the entry <None> is selected by default.
-  **Note** When your rule is configured, currently selected application classes appear at the top of the list.
- Step 6 Attempt to act as a**—Select server, client, or “client or server”
- From the pulldown menu, select **server**, **client**, or **client or server** depending on the *direction* of the connection you are controlling.
- If you are limiting a server’s connection limit, select server here. If you are limiting a client connection, select client here.
- Step 7 Communicating with**—Select specific hosts or all hosts
- When the rate limit set here is reached, you can determine whether all subsequent service requests are dropped or only those received or sent by a specific host. If you select a “specific” host, this indicates that the host in question exceeded the rate limit. If you select all hosts, this indicates that the sum total of to and from all hosts exceeds the limit and all hosts are blocked.
- Step 8 Over/Under a limit <100> network connections in <5> minutes**

Reasonable values are entered into these fields by default. They define the number of connections that can normally be expected during a time frame from either specific hosts or all hosts.

- If you select an action type of “deny” or “terminate”, and the limit is exceeded (*Over*) in this time frame (abnormal amount of connections that could represent an attack of the system), subsequent connection requests are dropped. (The dropped connections can be those received to/from individual “specific” hosts or to/from all hosts. This setting is configured at the bottom of the page.)
- If you configure this as an “allow” rule, you are setting a limit *Under* which the number of connections must remain for the subsequent connections to be explicitly allowed.

Step 9 When you are finished configuring your connection rate limit rule, click the **Save** button.

Figure 6-4 Connection Rate Limit Rule



153471

Data Access Control

Use data access control rules on Web servers to detect clients making malformed web server requests where such requests could crash or hang the server. A malformed request could also be an attempt by an outside client to retrieve configuration information from the web server or to run exploited code on the server. This rule detects and stops such web server attacks by examining the URI portion of the HTTP request.

An HTTP request consists of:

- the request method (a “get” or a “post”)
- the request URI (Uniform Resource Identifier—This includes the URL and related request parameters and arguments)
- the HTTP version (for example, HTTP/1.0)
- the HTTP header

The Data access control rule examines patterns in the URI portion of the HTTP request. The pre-configured Data sets (see [Data Sets, page 9-7](#)) group patterns to match based upon

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

Use the data access control rule to allow or deny specified underlying network data requests for the following web servers and platforms:

- Microsoft IIS (Windows platforms, version 4.0 or higher)
- Apache (Windows and UNIX platforms, versions 1.3, 2.0)
- IPlanet (UNIX platforms, version 6.0)



Caution

On Windows platforms, if you install Web server software (IIS or Apache) after you install the Cisco Security Agent on the server system, or if you have installed the Web server in a directory other than the default, you must manually install the Cisco Security Agent data filter in order to use Data access control rules on the system in question.

If your Web server software is already installed (in its default directory) when you install the agent on Windows, the server software is detected by the agent and the

data filter capability is automatically installed with the agent.

On Solaris (Apache or IPlanet servers) and Linux (Apache servers), in order to use Data access control rules you must install the data filter manually after you install the Cisco Security Agent. Unlike Windows, the Solaris and Linux installations do not detect Web server software and do not install the data filter with the agent. You must always manually install it.

See [Manual Agent Data Filter Installation, page 12-10](#) for instructions.



Note

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

Click the **Modify rules** link at the top of the Rule Module page to go to the Rules page.

-
- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Data access control** rule. This takes you to the configuration view for this rule type.
- Step 3** Enter the following information for the rule:
- **Description**—Enter a description of this rule.
This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.)
By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.
- Step 4** **Take the following action**—Select an action type from the pulldown list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#).
- Step 5** and
- **Log**—Enable this checkbox to turn logging on for this rule. Generally, you will want to turn logging on for all deny rules. This means that the denied system action in question is logged and sent to the server at regular time intervals.

- **Take precedence over other <action type> rules**—Enable this checkbox to manipulate policy rule precedence so that this rule is evaluated before other similar rules. You should generally NOT require this checkbox. Do not use it without understanding how it works. See [Rules: Manipulating Precedence, page 5-41](#) for details.

Step 6 When—Applications in any of the following selected classes

Select one or more preconfigured application classes here to indicate the application(s) whose access to the listed data sets you want to exercise control over.

Note that the entry, <All Applications>, is selected by default. You can use this default or you can unselect it and create your own application classes. For application class configuration details, see [Chapter 8, “Using Application Classes”](#).

- **But not in the following class**—Optionally, select application classes here to exclude from the application class(es) you’ve selected in the included applications field. Note that the entry <None> is selected by default.



Note When your rule is configured, currently selected application classes appear at the top of the list.

Step 7 Attempt to access these data sets

Click the **Insert Data Set** link to enter a pre-configured data set here. When you click this link, a list of the Data Sets you've configured appears here, allowing you to select one or more. Instead of data sets, you can list the literal data strings you want to protect. You can use a wildcard designation.

For information on configuring Data Sets, see [page 9-7](#).

Step 8 When you are finished configuring your Data access control rule, click the **Save** button.



Note If you specify an application here other than IIS, Apache, or iPlanet, this rule is ignored.

Figure 6-5 Data Access Control Rule

The screenshot displays the Management Center for Cisco Security Agents V5.1 interface in Microsoft Internet Explorer. The browser address bar shows the URL `https://bohazard/csamc51/webadmin`. The interface includes a navigation menu with options like **Events**, **Systems**, **Configuration**, **Analysis**, **Maintenance**, **Reports**, and **Search Help**. The current view is **Configuration** > **Rule Modules** > **Windows Rule Modules** > **Common Web Server Security Module [V5.1 r30]** > **Rules** > **Data access control [314]**.

The rule configuration details are as follows:

- Description:** IIS and Apache Web Servers, Common configuration file exploit. Includes a **Detailed** link.
- Enabled:** Enabled
- Take the following action:**
 - Action: **Deny** (selected from a dropdown menu)
 - and
 - Log Take precedence over other Deny rules
- when:**
 - Applications in any of the following selected classes:
 - Apache Web Server application [V5.1 r30]
 - IIS Web Server application [V5.1 r30]
 - <All Applications>
 - <Installation Applications>
 - <Processes Copying Untrusted Content>
 - But not in any of the following selected classes:
 - <none>
 - <Installation Applications>
 - <Processes Copying Untrusted Content>
 - <Processes Executing Untrusted Content>
 - <Processes Writing Untrusted Content>
 - Attempt to access these data sets:
 - \$Common configuration file names [V5.1 r30]

At the bottom of the configuration window, there are **Save** and **Delete** buttons, a status bar indicating **6 rule changes pending**, and a **Generate rules** button. The user is logged in as **admin**. The system tray shows the **Local intranet** icon.

153472

File Access Control

Use file access control rules to allow or deny what operations (read, write) selected applications can perform on files. You should understand that file protection encompasses read/write access. Directory protection encompasses directory deletes, renames, and new directory creation.

**Note**

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

Click the **Modify rules** link at the top of the Policy page to go to the Rules page.

-
- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **File access control** rule. This takes you to the configuration view for this rule type.
- Step 3** Enter the following information for the rule:
- **Description**—Enter a description of this rule. This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.)
By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.
- Step 4** **Take the following action**—Select an action type from the pulldown list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#).
- Step 5** and
- **Log**—Enable this checkbox to turn logging on for this rule. Generally, you will want to turn logging on for all deny rules. This means that the denied system action in question is logged and sent to the server at regular time intervals.

- **Take precedence over other <action type> rules**—Enable this checkbox to manipulate rule precedence so that this rule is evaluated before other similar rules. You should generally NOT require this checkbox. Do not use it without understanding how it works. See [Rules: Manipulating Precedence, page 5-41](#) for details.

Step 6 When—Applications in any of the following selected classes

Select one or more preconfigured application classes here to indicate the application(s) whose access to the listed files you want to exercise control over.

Note that the entry, <All Applications>, is selected by default. You can use this default or you can unselect it and create your own application classes. For application class configuration details, see [Chapter 8, “Using Application Classes”](#).

- **But not in the following class**—Optionally, select application classes here to exclude from the application class(es) you’ve selected in the included applications field. Note that the entry <None> is selected by default.



Note When your rule is configured, currently selected application classes appear at the top of the list.

Step 7 Attempt the following operations

Select the operations **Read** and/or **Write** you are allowing/denying on the files named in the Files field. For directory protection, the actions you are allowing/denying are **Create**, **Delete**, and **Rename**. Refer to File and Directory protection in [Using the Correct Syntax, page 2-32](#).



Caution

Directory protection ignores the file portion of the specified path and only matches the directory portion of the path. If the directory portion is not well specified, the protection will be overly broad. For example, if you select to protect a directory in a deny rule and enter the directory path as follows: `**\Program Files**Outlook.exe`, then no directories can be modified under Program Files. That is an overly broad protection to specify and would likely result in system instability. If you choose to protect directories, be sure to get very specific in your path string and understand the resulting behavior.

Step 8 On any of these files

Click the **Insert File Set** link to enter a pre-configured file set here. When you click this link, a list of the File Sets you've configured appears here, allowing you to select one or more. Instead of file sets, you can list the literal files you want to protect, using the file paths (including wildcards).

For information on entering file path literals here rather than using pre-configured File Sets, see [Using the Correct Syntax, page 2-32](#).

For local system paths, you must specify the disk drive. You can use a wildcard designation. When protecting directory creates, in particular, you should note that directory creation applies to an exact directory path match, but directory write and rename protection applies to all directories explicitly named in a path. If a directory name is completely wildcarded `**\`, no protections exist for that particular component of the directory. For example:

Windows:

```
*:\Program Files\winnt\*  
or @system\** (this indicates all files below the system directory)
```

UNIX:

```
/etc/passwd
```

For network machines (Windows only), enter

```
\\<machine name>\<share>\<path>\<filename>
```

For example: `\\Backup_Server\finance\records\database.db`

You can enter more than one file path, but each entry must appear on its own line. For File Set configuration details, see [page 9-10](#).

**Caution**

Symbolic Links and Hard Links: For UNIX, if you create a File access control rule to protect a symbolic link, **ONLY** that symbolic link is protected. The underlying resource, unless also specified, is **NOT** protected. For example, a File access control rule written for `/etc/hosts` does not protect `/etc/inet/hosts`. Similarly, a File access control rule written for `/etc/inet/hosts` does not protect `/etc/hosts`. If you want to protect a symbolic link and its underlying resource, both must be specified in the rule.

Also note that on UNIX systems, if you attempt to create a hard link to an agent-protected file, that action is seen as a write attempt on the file.

**Note**

Use **@dynamic** in the File set text field to indicate all files that have been quarantined by CSA MC as a result of correlated email worm events, correlated virus scanner log messages, or files that were added manually by the administrator. 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 9

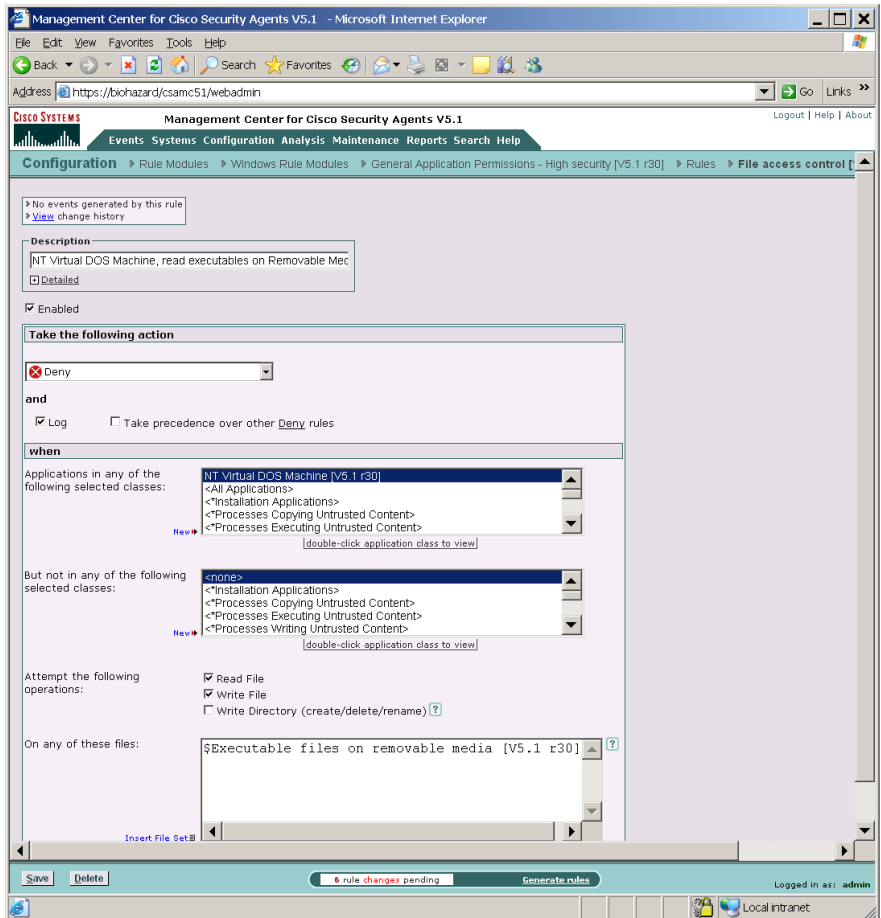
When you are finished configuring your File access control rule, click the **Save** button.

This rule is now part of your new policy. It takes effect when the policy is attached to a group and then downloaded by an agent on the network.

**Caution**

In order to distribute rules to the correct hosts, you must associate policies with groups and then Generate rules. See [page 5-34](#) for instructions.

Figure 6-6 File Access Control Rule



153473

Network Access Control

Use network access control rules to control access to specified network services and network addresses. You can also use this rule type to listen for applications attempting to offer unknown or not sanctioned services.



Note

The following instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

-
- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Network Access Control** rule. This takes you to the configuration view for this rule type.
- Step 3** Enter the following information:
- **Description**—Enter a description of this rule. This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.) By not selecting this checkbox, you can save this rule, but it will not be active in the policy and it will not be distributed to groups.
- Step 4** **Take the following action**—Select an action type from the pulldown list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#).
- Step 5** **and**
- **Log**—Enable this checkbox to turn logging on for this rule. Generally, you will want to turn logging on for all deny rules. This means that the denied system action in question is logged and sent to the server at regular time intervals.
 - **Take precedence over other <action type> rules**—Enable this checkbox to manipulate rule precedence so that this rule is evaluated before other similar rules. You should generally NOT require this checkbox. Do not use it without understanding how it works. See [Rules: Manipulating Precedence, page 5-41](#) for details.
- Step 6** **When**—Applications in any of the following selected classes

Select one or more preconfigured application classes here to indicate the application(s) whose access to the listed services and addresses you want to exercise control over.

Note that the entry, <All Applications>, is selected by default. You can use this default or you can unselect it and create your own application classes. For application class configuration details, see [Chapter 8, “Using Application Classes”](#).

- **But not in the following class**—Optionally, select application classes here to exclude from the application class(es) you’ve selected in the included applications field. Note that the entry <None> is selected by default.

Step 7 Attempt to act as a—Select server or client or both, or select listener

From the pulldown menu, select **server**, **client**, **client or server**, or **listener** (see [page 6-31](#) for more information on the listener option) depending on the direction or type of connection you are controlling or listening for.

If you are limiting a server’s contact with clients, select server here and enter the client(s) address in the host addresses field. If you are limiting a client’s contact with a server, select client here and enter the server(s) address in the host addresses field.

Step 8 for network services

Enter the literal protocol/port number combination for the service you want to control access to or click the **Insert Network Service** link to enter a pre-configured network service variable here. When you click this link, a list of the Network Service Variables you’ve configured appears here, allowing you to select one or more.

This field refers to either a server providing this service or a client accessing this service. For Network Service configuration details, see [page 9-18](#).

Step 9 Communicating with host addresses

Enter the literal network address(es) for the client/servers you want to control access to or click the **Insert Network Address Set** link to enter a pre-configured network address set variable here.

If you select server in the previous pulldown list, you enter client addresses here. If you select client in the previous pulldown list, you enter server addresses here. Note that you can use Network Address Set variables.

- You also can use the following "short hand" entry in Network Address Sets and in Network Access Control rules to indicate all local addresses on the agent system in question. The @symbol must appear at the start of the short hand name.

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.

Refer to [Using the Correct Syntax, page 2-32](#) for more valid @ entries that can be used in this rule type.

Step 10 Using these local addresses

Enter the literal network address(es) for the local system addresses you want to control (i.e. control clients making connections from or control servers making connections to). You can also click the **Insert Network Address Set** link to enter a pre-configured network address set variable here.

The addresses or address ranges you enter here can be used to control the host initiating the network connection. For example, you could write a Network access control rule which would only allow laptop users to connect to an internal network database if their connection is coming through a VPN (i.e. machine using an allowed/disallowed address to make a connection, incoming or outgoing). If the connection attempt comes in through an ISP-assigned address that is not part of this rule, it would not be allowed.

You could also use this field to impose a restriction that only trusted addresses can read an internal server. If the connection is received from an internal system or via a VPN from a fixed, trusted address, it is allowed.

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.

Step 11 When you are finished configuring your Network access control rule, click the **Save** button.

This rule is now part of your rule module. It takes effect when the policy is attached to a group and then downloaded by an agent on the network. You should note that new rules only apply to new connections. See [Preserving Application Process Classes, page 8-8](#) for details.



Caution

In order to distribute rules to the correct hosts, you must associate policies with groups and then Generate rules. See [page 5-34](#) for instructions.

**Note**

No network access control rule denial events are logged for any UDP port resulting from multicast packet signals. (If a collection of hosts have the same network access control rule and a broadcast such as UDP/138 were denied, then event messages would inundate CSA MC.)

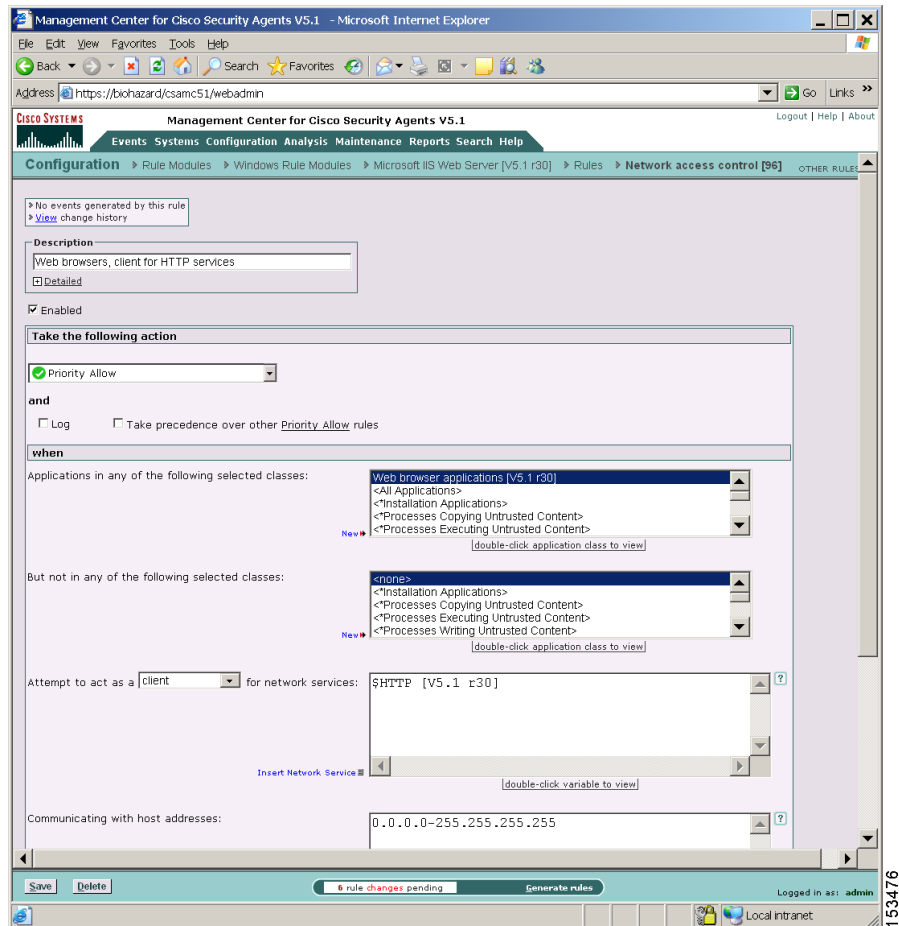
**Note**

When the system accepts a network connection on behalf on an application, the system requires an immediate answer to allow or deny the connection. Therefore, resource requests which trigger network access control server queries will immediately choose the query default. The user is still queried and the response will be cached for future connections.

What is the “listener” option for?

You can use the listener option in a Network access control rule to indicate what applications have the ability to be a server before they are allowed to accept a server connection. This is in contrast to the “server” option which offers real-time per connection control. The listener option can be used in a monitoring capacity to reveal any applications that are attempting to offer a network service. For example, if a system is already infected with a Trojan, that Trojan may be listening on a high numbered port for a network server connection. A NACL listener rule would detect this occurring before a server connection is achieved. You could then craft a subsequent NACL rule to deny the server connection.

Figure 6-7 Network Access Control Rule



153476

Network Shield Rule

The Network shield rule provides network protocol stack hardening capabilities.



Note The information provided in this manual, in this section especially, assumes a basic knowledge of TCP/IP. A good source for further reading on the topic is the book *Internetworking with TCP/IP*, Douglas R. Comer and David L. Stevens, Prentice Hall, Inc.

-
- Step 1** In the Network shield rule configuration view, enter the following information:
- **Description**—Enter a description of this rule. This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.) By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.
- Step 2** **Take the following action**—(Note that only Priority Deny, Allow, Deny, Monitor, and Add process action types are available for this rule. You can choose to add the system process to the Processes communicating with Untrusted Hosts application class which causes the remote host IP address to be sent to the MC for global correlation. This may result in the address being added to the @dynamic address list for quarantining. Also see [Correlation, page 7-6](#) for details on quarantining IP addresses.) Select an action type from the pull-down list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#).



Note Because IP addresses can be spoofed, we don't recommend using this capability for this rule type. It is more applicable for NACL-based rules where you are sure you are communicating with the address. (i.e. an established TCP connection.)

- Step 3** and
- **Log**—Enable this checkbox to turn logging on for this rule. Generally, you will want to turn logging on for all deny rules. This means that the denied system action in question is logged and sent to the server at regular time intervals.
 - **Take precedence over other <action type> rules**—Enable this checkbox to manipulate rule precedence so that this rule is evaluated before other similar rules. You should generally NOT require this checkbox. Do not use it without understanding how it works. See [Rules: Manipulating Precedence, page 5-41](#) for details.

- Step 4** when detecting (Select one or more of the checkboxes described here. Please note any address and/or state condition restrictions that are called out beside each check.)

**Caution**

You cannot use network shield rules in rule modules that have user state conditions set. If you attempt to attach a user state to a rule module that contains a network shield rule, you will be notified of a configuration error.

IP Security checks

- Invalid IP header

Enabling this feature causes the Cisco Security Agent to perform an integrity check on the IP packet header. This includes performing a consistency check on the IP header, on the length of the IP header, and on the number of bytes in the packet. If you configure this as a Deny rule, the following occurs: if any of these checks fail, the packet is dropped, if an IP checksum fails, the packet is dropped, IP options and IP fragments are validated as well and dropped if they are found to be invalid. (This defeats attacks such as Teardrop, Boink, and Ping of Death.)
- Invalid IP address

IP addresses are determined to be invalid for several reasons: if the source address is a multicast address, if the TCP connection is to a broadcast address. You can select this checkbox as part of a Deny rule to protect against these types of attacks.
- Source routed packet

This detects IP options which control explicit routing instructions for packets. With IP source routing (an IP header option) the originator of a packet can try to partially or completely control the path through the network to the destination.
- Trace route

This detects the mapping of network topology via trace route.

Transport Security checks

- Invalid TCP/UDP/ICMP header

This check ensures that transport headers are the proper length and that they are consistent (have enough data in the packet for them to fit). This includes verifying that certain fields have valid values and that certain combinations of TCP flags are legal. This defeats attacks such as a Christmas Tree scan.

- TCP SYN floods

SYN flooding is a type of denial of service attack. It occurs when a TCP/IP connection request is received from a return address that is not in use (i.e. a non-existent host for a spoofed address) resulting in a half open connection. An abundance of half open states on a server can prevent legitimate connections from being established. Detecting and preventing SYN floods stops this attack from succeeding.

(This rule type is not available for UNIX policies as the UNIX OS already provides this protection.)

Servers that are external to your network and not protected by a firewall should be protected against SYN floods. Firewalls generally provide this protection.

**Note**

If you enable the “TCP SYN floods” and the “TCP blind session spoofing attempts” checkboxes, you cannot enter address restrictions into the address field for this rule. You must use all addresses.

- TCP blind session spoofing attempts

If you configure this as a Deny rule, this check causes agents to make TCP sequence numbers unpredictable.

A server accepting connections using predictable TCP sequence numbers may be tricked into accepting a connection from a malicious source that is spoofing a trusted host. This prevents that vulnerability.

(This rule type is not available for UNIX policies as the UNIX OS already provides this protection.)

**Note**

If you make any changes to the “TCP blind session spoofing attempts” feature, these changes are not enforced until after the agent system(s) is rebooted.

- TCP/UDP port scan

Port scanning is a common method for finding weaknesses at a site by determining what network services are being run. An attacker attempts to connect to port after port on a target system, mapping ports to identify network services and machine type vulnerabilities. Configure this rule to log an event when an attempt is made to scan the system for an open port. Information is also gathered on the number of different source IP addresses perpetrating the scan and it reveals the source. In most cases, you should apply port scan detection to servers and end-user systems in your enterprise.

Configure this rule as a Deny rule to prevent unauthorized port scans, effectively cloaking a system on the network. Denying port scans causes a system to not respond to connectivity tests and to not respond to service requests with connectivity error messages.

A system generally sends out error messages when a remote machine sends a request for a service which is not running on the system. Often, this is how remote machines locate other systems and obtain network information about the system in an attempt to target it for an attack. By not responding, this prevents both UDP and TCP-based port scans of the system and basically hides it on the network.

If you are running an allowed service on a system and you are denying port scans, connection requests to this service are honored and your machine is viewable for the service you're offering.

**Note**

If you select the network scans correlation checkbox in the Global Event Correlation page (see [Correlation, page 7-6](#)), when scans are detected and denied across several machines, CSA MC correlates these events and generates an additional event to warn of this correlation. Note that this correlation only occurs when Deny rules are triggered.

- ICMP ping message

This check works similar to the TCP/UDP port scan feature, but for ping scans. See the port scan description for information.

- ICMP configuration message

If you configure this rule as a Deny, this feature restricts messages which can change the configuration of a machine. For example, a redirect can be used to cause routing tables to be updated.

- ICMP information message

Some ICMP messages may be used to gather information about a machine in an attempt to attack it. This data, when obtained, can be used to gather system information which can be used to exploit the system. If you configure this rule as a Deny, this feature restricts messages which report back on system or network configuration.

- ICMP covert channel

Configuring this rule as a Deny, causes agents to drop unsolicited echo responses.

The Cisco Security Agent validates that the echo response data matches the echo request data. This way, ping cannot be used as a transport for communications.

- Malicious packet

Configuring this rule as a Deny causes agents to block packets which are technically legal but are known exploits against protocol stacks (e.g. UDP packet storm or RF poison).

System Startup Security checks

- Unrestricted network connectivity during boot

Configuring this feature as a Deny, prevents non-essential network connections during system startup. This check is automatically disabled when the agent service starts and policies (including those which govern allowed network connections) are enforced. This protects the system from network-based attacks at boot-time before the agent service has started.

(This rule type is not available for UNIX policies.)



Note

If you enable the Unrestricted network connectivity during boot checkbox, you cannot enter address restrictions into the address field for this rule. You must use all addresses.



Note

You cannot use a rule that has the Unrestricted network connectivity during boot checkbox selected in policies with rule modules that have system and/or user state conditions set.

Step 5 and Communicating with host addresses

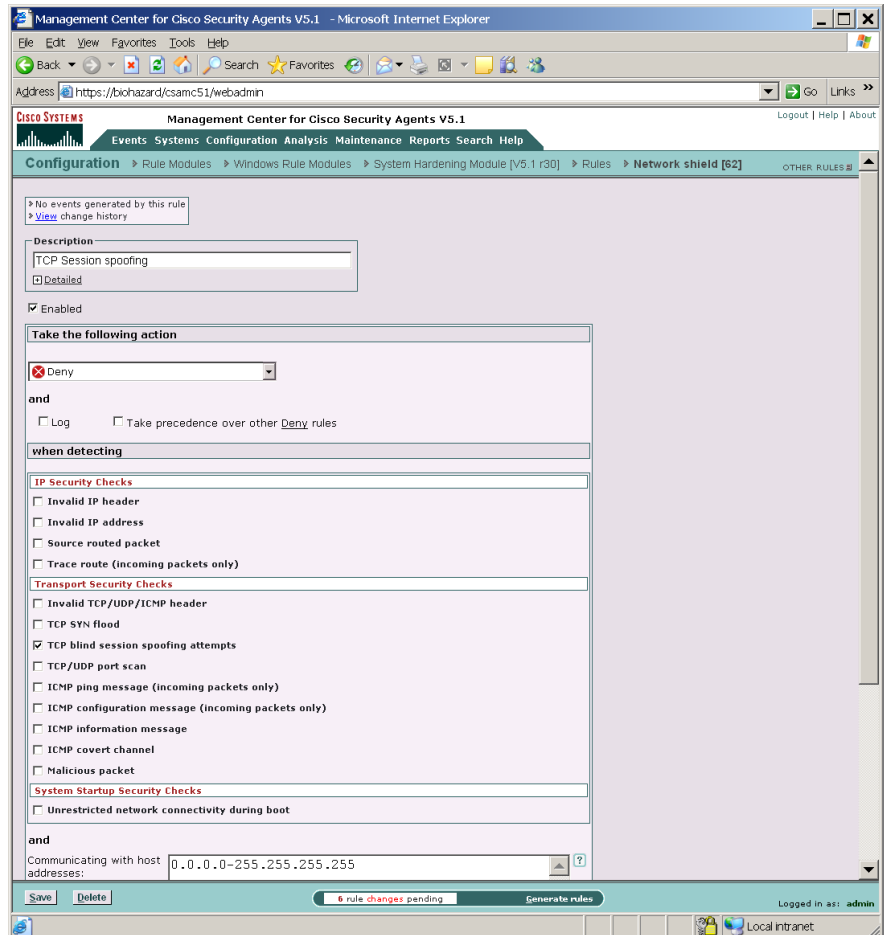
Optionally, you can enter specific addresses here for those checkbox features in this rule that support addressing parameters.

Step 6 Using these local addresses

By default, this field indicates all local addresses on the agent system. You would want to use this to identify specific network interfaces, if necessary.

Step 7 Click **Save** when finished.

Figure 6-8 Network Shield Rule

**Note**

Refer to [Replicate Feature, page 6-44](#) for details on easily propagating the changes you make to one Network shield rule to other Network shield rules in other policies.

System API Control Rule

The System API control rule detects several forms of malicious programming code that is installed on a system by an unsuspecting user either thinking that he or she is running some other type of program, or as a result of some other activity such as reading an attachment to an email message. Once installed, these malicious programs (for example, Trojans) may allow others to access and virtually take over a system across the network. Other errant programs may be set up to automatically send mail messages or other types of network traffic (including system passwords) while the system owner is unaware of what is occurring.

**Note**

This rule type is not available for UNIX policies. Refer to the Buffer overflow rule information on [page 6-70](#) for similar UNIX functionality.

It could be useful, especially in the case of server systems, to use a service restart rule in conjunction with a System API control rule. This way, if you are forced to press the Terminate button if queried by a triggered rule and you subsequently terminate the application in question, a service restart rule will cause the application to automatically restart.

Use the System API control rule in a policy to detect and prevent errant programs from performing malicious acts on individual systems and networks. The included System API control rule lets you enable several different types of system detection. See [Figure 6-9](#).

System Information Checks

- Access local configuration information
Detect applications that attempt to read system registry settings.
- Access Security Account Manager
Detect applications that attempt to steal local system passwords.

System Monitoring Checks

- Trap keystrokes
Detect applications that attempt to capture system keystrokes.

- Monitor media devices

This checkbox lets you control which applications can monitor media devices on the system. Media device “inputs” can be exploited by Trojans which can, for example, turn on the microphone on a system and covertly listen to a conversation.

Patterns to be included: Use the Wizard from the Event log message in question to include particular devices in a System API “allow” rule. You must specify media devices as “device\port”. For example, `plantronics\microphone`.



Note Monitor media devices is not supported on Windows NT systems. It is also not supported for parallel port media devices on any operating system.

System Modification Checks

- Access physical memory
Detect applications that attempt to directly access physical memory while bypassing virtual memory restrictions.
- Download and invoke ActiveX controls
Detect applications that download ActiveX controls and immediately attempt to execute them.
This functionality limits applications from downloading ActiveX controls (signed and unsigned). This type of behavior is generally typical of a web browser and sites that require the downloading of ActiveX can trigger this rule. Note that this rule may be unnecessary if system web browser settings are configured with a "High" security level that would restrict the downloading of ActiveX controls.
- Inject code into other applications
Detect applications that are attempting to write code to space owned by other applications. e.g. injecting a malicious .dll into a privileged process.
- Write memory owned by other applications
Detect applications that attempt to interfere with the memory space of other applications or detect Trojans attempting to hide in another executable to escape detection and gain permissions to access other resources.

Atypical System Behavior Checks

- Access system functions from code executing in data or stack space

Although this behavior is sometimes exhibited by downloaded/executable content (e.g. license checking software), this may be symptomatic of a buffer overflow attack.

 - Patterns to be included: Use the Wizard from the Event log message in question to include a particular pattern in a System API “allow” rule when you are seeing buffer overflow events you believe are harmless.
- Handle exceptions

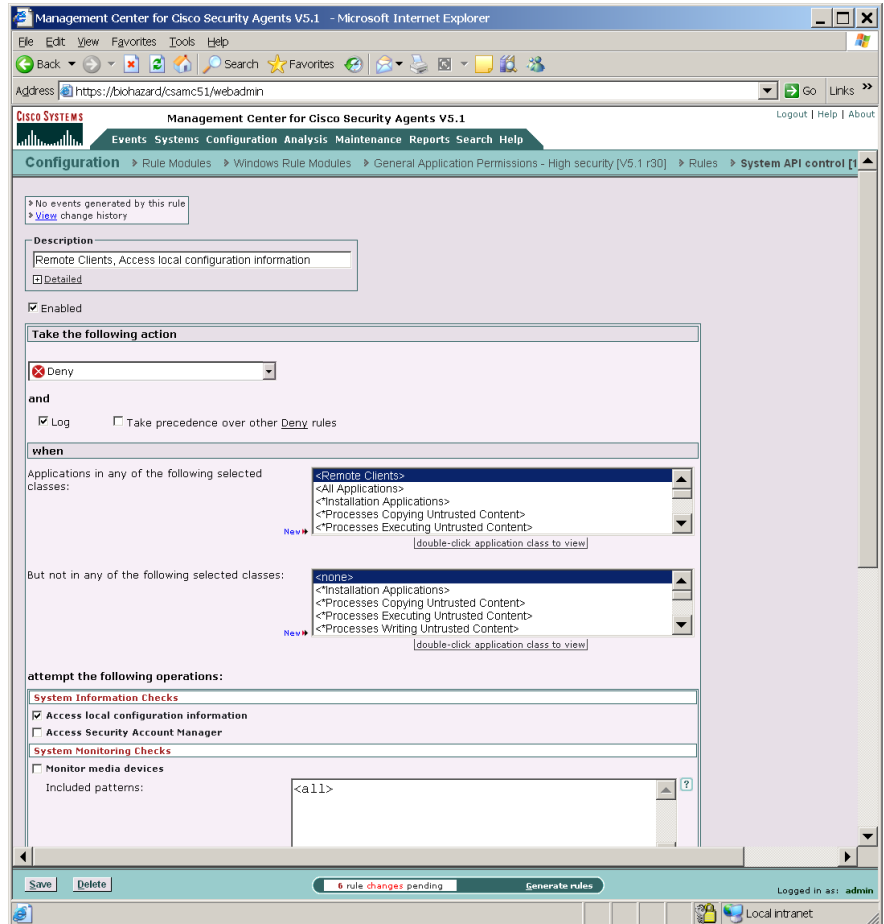
Detect processes running exception handling routines. This typically occurs due to bugs in the application software. But this may be a sign of an attack if this occurs with an application that does not generally exhibit this behavior.
- Invoke unusual system calls

Use this checkbox to detect processes invoking system calls that are rarely used. In normal system operation, many system calls are either never used or may only be used infrequently by a specific system application performing a service. Attempting to exploit undetected flaws in these unusual system calls is common attack vector for malware.

 - Patterns to be included: Use the Wizard from the Event log message in question to include a particular module in a System API “allow” rule when you are seeing events you believe are harmless.

You also have the ability to select specific **application classes to exclude** from the various System API control rules you designate. For example, in some cases, debuggers may perform actions that can be misconstrued as malicious behavior. Therefore, you would want to create an application class, and select it as an exclusion to one or more System API control rule features.

Figure 6-9 System API Control Rule



153494

Replicate Feature

When you make rule changes and click the Save button for rule types that contain multiple checkboxes, such as System API control rule (Network shield and Buffer overflow rules also provide this feature) a "replicate" link appears beside the "Saved changes" message at the top of the rule page. Click on **replicate** to access a pop-up box. From this box, you select other policies that contain System API control rules and choose to propagate the same change(s) you made on the current page to System API control rule pages in other policies. If the change you make to one System API control rule page is a change you need to make to all System API control rules in all your policies, this is a quick way to propagate those changes on a wide or even global scale.

Windows Only Rules

The following rules are only available for Windows Rule Modules.

Clipboard Access Control

Use the clipboard access control rule to dictate which applications can access information that is written to the clipboard. When writing security policies, you may want to protect information from being accessed by other applications or network processes. To fully protect this information, you must consider preventing other applications from accessing protected information that may have been written to the clipboard.

This rule works in the following manner. When a process belonging to an application class specified in a clipboard rule writes to the clipboard, the data is marked as protected. Only processes which also match the specified application classes are allowed to read the data from the clipboard. When a process that does not belong to the application class specified writes to the clipboard, the data is marked as unprotected. Any process is allowed to read from the clipboard then.

For example, if Microsoft Excel (which is part of the Microsoft Office application class) saves data to the clipboard, other Microsoft Office applications will be able to read the clipboard data to allow cut and paste functionality between Office applications. Non-Office applications would be prevented from reading this clipboard data.

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

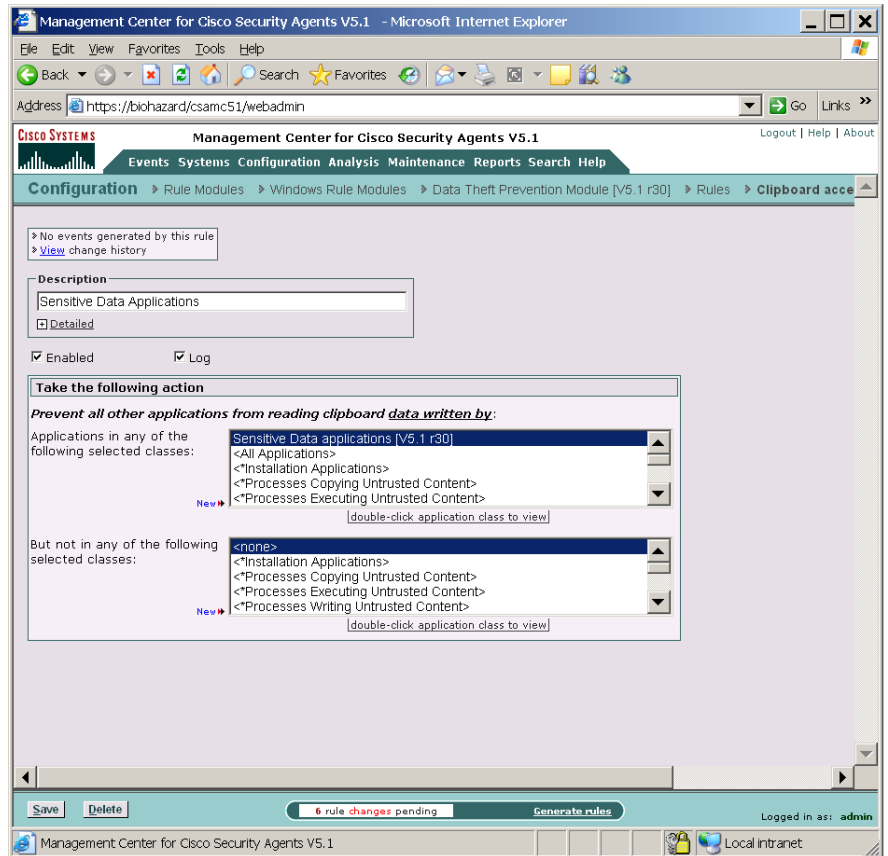
-
- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Clipboard access control** rule. This takes you to the configuration view for this rule type. See [Figure 6-10](#).
- Step 3** Enter the following information for the rule:
- **Description**—Enter a description of this rule. This description appears in the list view for the module.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.) By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups
- Step 4** Take the following action: **Prevent all other applications from reading clipboard data written by**:
- **Applications in any of the following selected classes**—Select one or more preconfigured application classes here to indicate the application(s) whose data you want to exercise control over. Note that the entry <All Applications> is selected by default. You can use this default or you can unselect it and create your own application classes.

When your rule is configured, currently selected application classes appear at the top of the list.
 - **But not in the following class**—Optionally, select application classes here to exclude from the application class(es) you've selected in the included applications field. Note that the entry <None> is selected by default.
- Step 5** When you are finished configuring your Clipboard access control rule, click the **Save** button.

**Note**

If you are using the Clipboard rule to restrict applications from accessing data on the clipboard, the system Print Screen functionality is also automatically disabled.

Figure 6-10 Clipboard Access Control Rule



COM Component Access Control

Use COM component access control rules to allow or deny applications from accessing specified COM components. COM is the Microsoft Component Object Model, the technology that allows objects to interact across process and machine boundaries as easily as within a single process. Each of the Microsoft Office applications (Word, Excel, Powerpoint, etc.)

exposes an "Application" COM component which can be used to create macros or utility scripts. While this is useful functionality, it can be used maliciously by an inadvertently downloaded Visual Basic script.

An example would be the Mydoom virus, which propagated by using the "Outlook.Application" COM component to send itself to each entry in the local address book. Using the COM component access control rule, you can protect specific COM components. For example, you could create a rule which limits access to Office components (Word.*, Outlook.*, Excel.*, etc.) only to the Office applications themselves. Non-Office applications (such as the Visual Basic scripting engine) would therefore be denied access to these components.

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

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **COM component access control** rule. This takes you to the configuration view for this rule type (see [Figure 6-15](#)).
- Step 3** Enter the following information
- **Description**—Enter a description of this rule.
This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.)
By not selecting this checkbox, you can save this rule, but it will not be active in the policy and it will not be distributed to groups.
- Step 4** **Take the following action**—Select an action type from the pulldown list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#).
- Step 5** **and**

- **Log**—Enable this checkbox to turn logging on for this rule. Generally, you will want to turn logging on for all deny rules. This means that the denied system action in question is logged and sent to the server at regular time intervals.
- **Take precedence over other <action type> rules**—Enable this checkbox to manipulate rule precedence so that this rule is evaluated before other similar rules. You should generally NOT require this checkbox. Do not use it without understanding how it works. See [Rules: Manipulating Precedence, page 5-41](#) for details.

Step 6 When—Applications in any of the following selected classes

Select one or more preconfigured application classes here to indicate the application(s) whose access to the selected COM components you want to exercise control over.

Note that the entry, <All Applications>, is selected by default. You can use this default or you can unselect it and create your own application classes. For application class configuration details, see [Chapter 8, “Using Application Classes”](#).

- **But not in the following class**—Optionally, select application classes here to exclude from the application class(es) you’ve selected in the included applications field. Note that the entry <None> is selected by default.

Step 7 Attempt to access a COM component....matching any of the following component sets

Click the **Insert COM Component** link to select one or more pre-configured COM component sets for this rule. If you do not want to use a COM component set variable, using the correct syntax, enter a literal PROGID or CLSID (one per line) here. CSA MC provides a utility for extracting PROGID and CLSID information from systems running agent software. See [Using the COM Extract Utility, page 12-9](#) for instructions.

PROGID’s, use the following syntax:

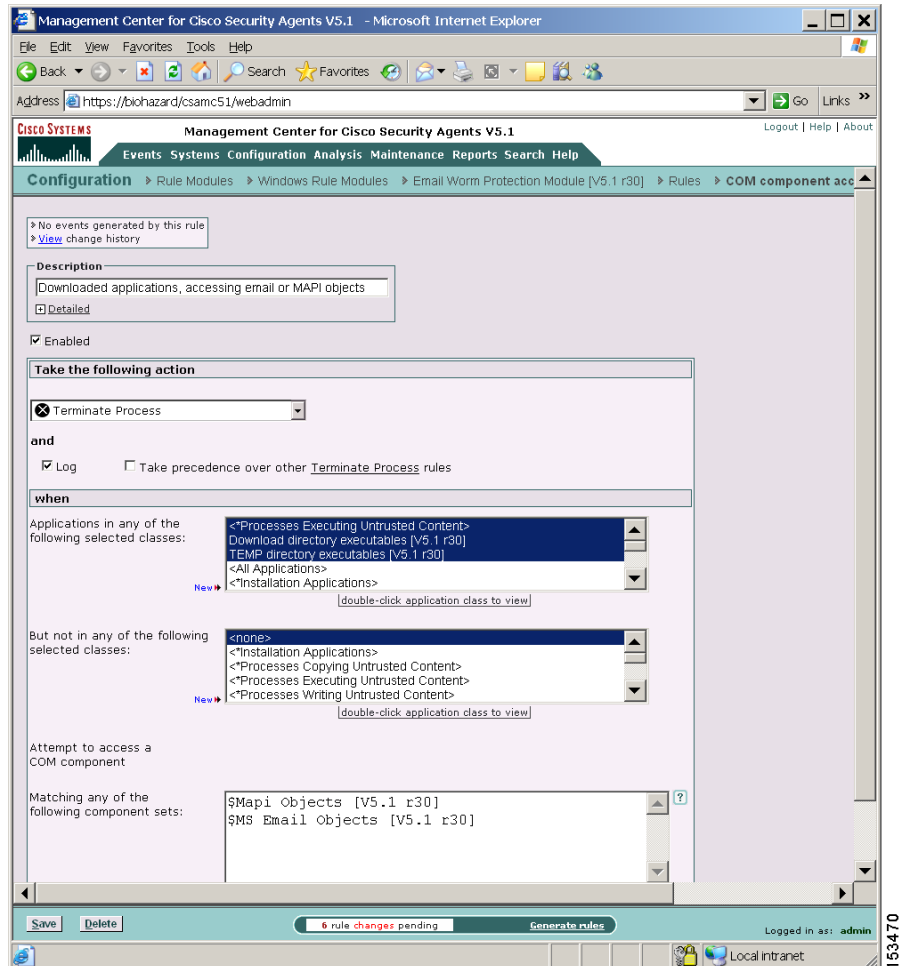
```
Outlook.Application
```

When entering CLSID’s (uppercase hexadecimals) using the following syntax, you must include the brackets shown here:

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

- #### Step 8
- When you are finished configuring your COM component access control rule, click the **Save** button.

Figure 6-11 COM Component Access Control Rule



File Version Control

Use the File version control rule to control the software versions of applications users can run on their systems. For example, if there is a known security hole in one or more versions of a particular application, this rule would prevent those specific versions from running, but would allow any versions not included in this

rule to run unimpeded.

One particular example where this type of rule would be beneficial is in the case of Microsoft Security Bulletin (MS01-020). This bulletin states the following: "Because HTML e-mail messages are Web pages, Internet Explorer can render them and open binary attachments in a way that is appropriate to their MIME type. However, there is a flaw in the type of processing that is specified for certain unusual MIME types. If a malicious user creates an HTML e-mail message that contains an attachment that can be run and then modifies the MIME header information to specify that the attachment is one of the unusual MIME types that Internet Explorer handles incorrectly, Internet Explorer may run the attachment automatically when it renders the e-mail message."

Microsoft has a patch to correct this security problem, but the patch is only available for Internet Explorer 5.01 Service Pack 1 and IE 5.5. If users are running an earlier version of IE, they must upgrade to 5.01 or 5.5 and install the correct service packs and patches to correct the problem. Therefore, earlier versions of IE contain an unfixable security problem and you will want to prevent users from running these versions. The following configuration information uses the IE security bulletin as an example.



Note

Note that users can get around a File version control rule by copying the file in question to a different file name. Therefore you must assume that users are working in cooperation with you for these rule types to be successful. You could also create a File access control rule to prevent users from changing the application file name in question.



Note

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **File version control** rule. This takes you to the configuration view for this rule type.
- Step 3** In the File version control rule configuration view, enter the following information:

- **Description**—Enter a description of this rule. This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
- **Enabled**—Use this checkbox to enable this rule within the policy. (It is enabled by default.) By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.

Step 4 Take the following action—Select an action type from the pulldown list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#).

Step 5 and

- **Log**—Enable this checkbox to turn logging on for this rule. Generally, you will want to turn logging on for all deny rules. This means that the denied system action in question is logged and sent to the server at regular time intervals.
- **Take precedence over other <action type> rules**—Enable this checkbox to manipulate rule precedence so that this rule is evaluated before other similar rules. You should generally NOT require this checkbox. Do not use it without understanding how it works. See [Rules: Manipulating Precedence, page 5-41](#) for details.

Step 6 When An execution of the following

Enter the **File** you are prohibiting (You will enter the exact version in the next field.) This field accepts file entries for .exe, .dll, and .ocx files. Enter just the file name here. No path is required.

For example: `iexplore.exe`

You cannot use wildcard entries in this field.

Step 7 with version within these Version ranges

Enter the version or version range (using a dash to indicate range) of the file you entered in the previous field.

For example: `0-5.00.3314.2100`

`5.00.3314.2100-5.50.4522.1800`

You can enter multiple, nonconsecutive ranges by entering versions on separate lines in this field.

To locate the version of a file (*.exe, *.dll, or *.ocx), select the file and right click. Select **Properties**. Click the **Version** tab. The File version is normally 4 values separated by dots.

**Note**

When entering version numbers for Microsoft applications, refer to the Microsoft web site. Application version numbers accessible from the application itself sometimes correspond to slightly different version numbers in Microsoft version charts. For example, Microsoft Article number Q164539 was used to determine the version numbers for this File version control rule.

Step 8 Click the **Save** button when you are finished.

Figure 6-12 File Version Control Rule

The screenshot displays the Management Center for Cisco Security Agents v5.1 web interface in Microsoft Internet Explorer. The browser address bar shows the URL `https://biohazard/csamc51/webadmin`. The page title is "Management Center for Cisco Security Agents v5.1". The navigation menu includes "Events", "Systems", "Configuration", "Analysis", "Maintenance", "Reports", and "Search Help". The current page is "Configuration > Windows Rule Modules > Desktop Module > Rules > File version control [415]".

The rule configuration details are as follows:

- Description:** Prevent unsafe versions of IE from running on systems. A checkbox for "Detailed" is present.
- Enabled:** Enabled
- Take the following action:**
 - Action: Deny
 - and
 - Log Take precedence over other Deny rules
- when**
 - An execution of the following
 - File: `ieexplore.exe`
 - with version within these
 - Version ranges:
 - 0.0.0.0-5.0.3313.65535
 - 5.50.0.0-5.50.4521.65535
 - is attempted.

A tooltip for the version ranges field provides the following instructions:

- Enter filename (not path)
- Allowed extensions : exe, dll, ocx
- No wildcards allowed

At the bottom of the interface, there are "Save" and "Delete" buttons, a status bar indicating "4 rule changes pending", and a "Generate rules" button. The user is logged in as "admin".

153474

Kernel Protection

Use the Kernel protection rule to prevent unauthorized access to the operating system. In effect, this rule prevents drivers from dynamically loading after system startup. You can specify exceptions to this rule for authorized drivers that you are allowing to load any time after the system is finished booting.

You can also use this rule to only detect unauthorized access to or modification of the operating system at any time. This rule also detects if a system was booted in a non-standard, insecure manner.



Note

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

-
- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Kernel protection** rule. This takes you to the configuration view for this rule type.
- Step 3** Enter the following:
- **Description**—Enter a description of this rule. This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.) By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.
- Step 4** **Take the following action**—Select an action type from the pulldown list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#). Note that Priority Deny, Allow, Deny, Add Process to Application Class, and Monitor are the only action types available. If you select Add Process to Application Class, the two classes you are adding a given process to are either Authorized rootkit or Unauthorized rootkit.
- Step 5** and
- **Log**—Enable this checkbox to turn logging on for this rule. Generally, you will want to turn logging on for all deny rules. This means that the denied system action in question is logged and sent to the server at regular time intervals.

- **Take precedence over other <action type> rules**—Enable this checkbox to manipulate rule precedence so that this rule is evaluated before other similar rules. You should generally NOT require this checkbox. Do not use it without understanding how it works. See [Rules: Manipulating Precedence, page 5-41](#) for details.

Step 6 when

- **Modules load after system startup**

The default here is <none>. You can use this rule type to prevent drivers from dynamically loading after system startup. You can also specify exceptions to this rule for authorized drivers that you are allowing to load any time after the system is finished booting.



Caution

This part of the rule only detects unauthorized access to the operating system and does not prevent it. Upon this detection, you can impose stringent network restrictions by selecting the Restrict network connectivity... checkbox.

- **Modules modify kernel functionality**

When modules are detected modifying the system, selecting this checkbox causes the system in question to log this event. You can use this detection to create dynamic rootkit application classes and to change the system state to a state that enforces a more restrictive policy.

You should only create Allow exceptions for actions you believe are safe. For example, virus-scanners and kernel debuggers might legitimately trigger this rule. Enters module data in the following edit fields:

– **Module hashes to be excluded**

By default, this field contains <all>. Enter hashes and/or drivers that identify kernel modules (e.g. drivers) into this field in the following format: 20 character hash\file system path\driver name. You can use wildcards for entries, as well. You can also use the wizard from the event in question to enter the module hash and driver information here. Some examples of valid entries are as follows:

```
*\**\system32\Drivers\uphcleanhlp.sys
ae45e23b45093dfafa899\**
ae45e23b45093dfafa899\**\uphcleanhlp.sys
```

– **Code patterns to be excluded**

By default, this field contains <all> . The wizard enters code patterns (not inside any module) into this field.

- **The previous detected boot was insecure**

When a system has previously booted in a non-standard manner, selecting this checkbox causes a message to be sent to the event log. A boot is considered standard if the system was booted from the primary hard disk. Any other boot type, for example, booting from a peripheral device (CD ROM) or a hard disk that is not the primary, is considered non-standard. A non-standard boot may be considered suspicious. (e.g., This is one way of circumventing the Cisco Security Agent and introducing a Trojan to a system.)

The insecure boot detection this checkbox enables works in conjunction with a particular type of compatible BIOS on compliant systems. The compatible BIOS detects a non-standard boot and on the next normal boot, if you have this checkbox selected, a message is sent to the MC which logs this insecure boot detection. (If a Host is running a BIOS that supports this insecure boot detection feature, the individual Host details page will indicate this. From the CSA MC menu bar, click Systems>Hosts. Then click on the link for an individual host. The Host Status section includes a category named "BIOS supported boot detection".)



Note

A Safe Mode boot also falls into this insecure boot category since the Cisco Security Agent provides no security in Safe Mode. Compatible BIOS is *not* required for a Safe Mode boot detection.

- **Included boot patterns**

By default, this field contains <all> . You can use provided tokens here to only detect certain boot types as insecure or to exclude a particular boot type from the detection. Available tokens are:

@fixed - This indicates a fixed hard disk that is not the primary disk. (Compatible BIOS is required to detect this.)

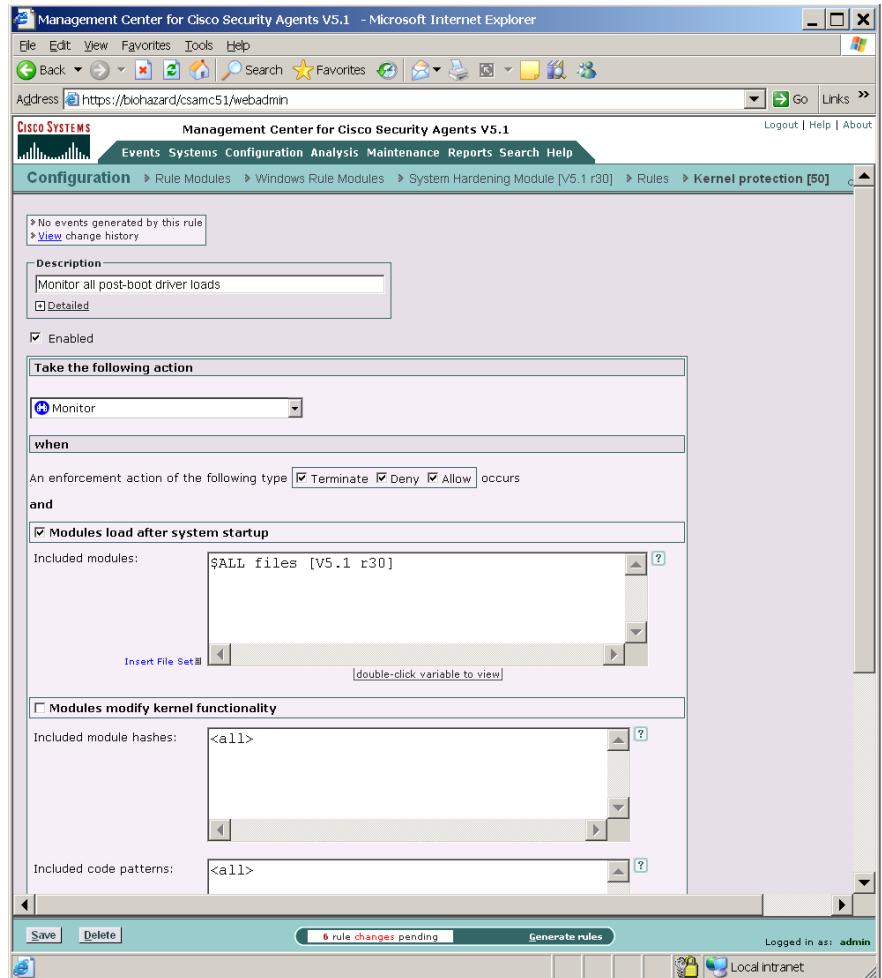
@network - This indicates all network shares. (Compatible BIOS is required to detect this.)

@removable - This indicates all removable media. That includes, floppies, CDs, zip drives, etc. (Compatible BIOS is required to detect this.)

@safemode - This indicates the detection of a system having booted in any debug mode in which the agent does not load. (Detecting this is not BIOS dependent.)

Step 7 Click **Save** when finished.

Figure 6-13 Kernel Protection Rule



NT Event Log

Use the NT Event log rule to have specified NT Event Log items appear in the CSA MC Event Log for selected groups.



Note

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **NT Event log** rule. This takes you to the configuration view for this rule type (see [Figure 6-14](#)).
- Step 3** Enter the following information:
- **Description**—Enter a description of this rule.
This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.)
By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.
- Step 4** **Log events from the event log**
- **Include events matching the following**—Select this radio button to specify the criteria for NT Event Log entries which you want to appear in the CSA MC Event Log.
 - **Include all events except those matching the following**—Select this radio button to specify the criteria for NT Event Log entries which you do not want to appear in the CSA MC Event Log. (All criteria not specified here will appear in the CSA MC Event Log.)



Note

You can configure CSA MC to correlate NT event types logged across multiple systems. You can also correlate NT events received from virus scanners running on agent systems and quarantine contaminated files. See [Installation Applications Policy, page 7-4](#).

- Step 5** **Criteria** (You should select at least one for the rule to have any effect.)

- **Event Log Type**—Select one or more checkboxes here to indicate which NT Event Log entries you want to appear (first radio button above) or which entries you want to not appear (second radio button above) in CSA MC Event Logs.

The choices are—**System, Application, Security**

- **Event Source**—In the text field, enter (one per line) event source parameters you want to filter by.

The event source is the software that logged the event, which can be either an application name, such as *SQL Server*, or a component of the system or of a large application, such as a driver name. For example, *Elnkii* indicates the EtherLink II driver.

- **Event Severity (Type)**—Select one or more checkboxes to filter the viewing of events according to severity. If you select no checkboxes, all severity levels are included in the rule.

The choices are—**Information, Warning, Error, Audit Success, Audit Failure**

- **Event Code (Event ID)**—In the text field, enter (one per line) event code parameters you want to filter by.

The event code is the number identifying the particular event type. For example, 6005 is the ID of the event that occurs when the Event log service is started. You can find the event IDs for Windows security events by searching for the following articles on the Microsoft web site: Q174074, Q299475, and Q301677.

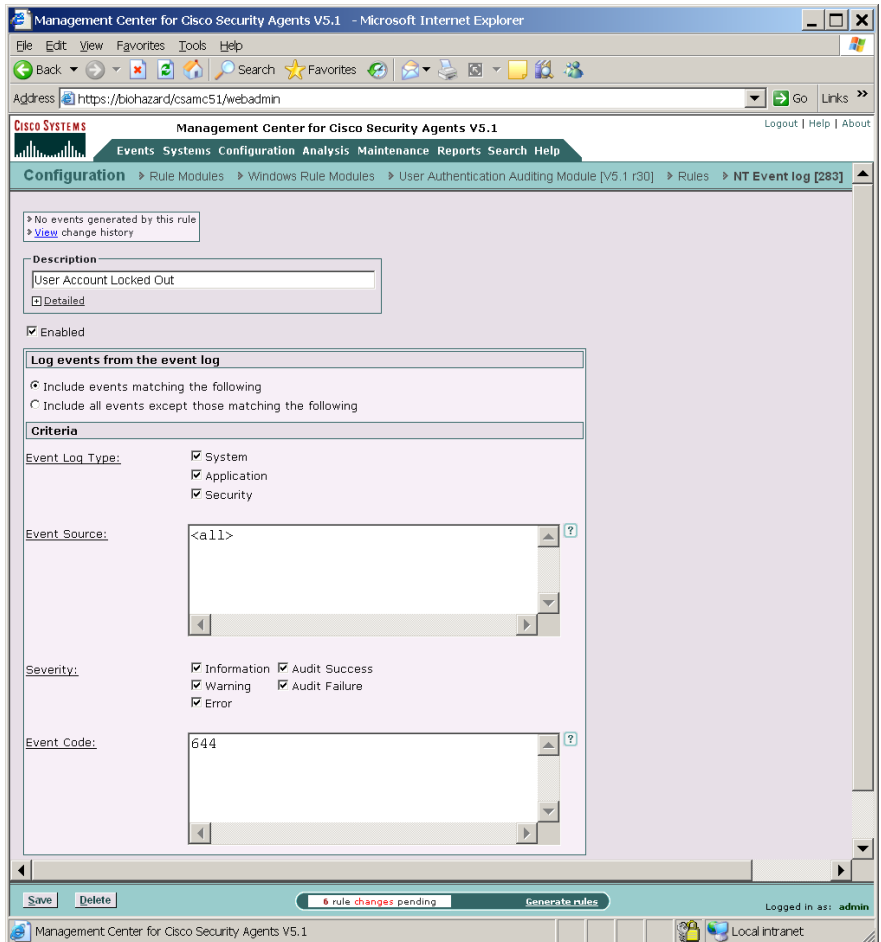
Step 6 Click the **Save** button.



Note

To receive messages logged by Norton AntiVirus and for global correlation, select the **Application** checkbox and enter *Norton AntiVirus* in the Event Source edit box. See [Installation Applications Policy, page 7-4](#) for more information.

Figure 6-14 NT Event Log Rule



153479

Registry Access Control

Use registry access control rules to allow or deny applications from writing to specified registry keys.

**Note**

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

-
- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Registry access control** rule. This takes you to the configuration view for this rule type.
- Step 3** Enter the following information:
- **Description**—Enter a description of this rule.
This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.)
By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.
- Step 4** **Take the following action**—Select an action type from the pulldown list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#).
- Step 5** and
- **Log**—Enable this checkbox to turn logging on for this rule. Generally, you will want to turn logging on for all deny rules. This means that the denied system action in question is logged and sent to the server at regular time intervals.
 - **Take precedence over other <action type> rules**—Enable this checkbox to manipulate rule precedence so that this rule is evaluated before other similar rules. You should generally NOT require this checkbox. Do not use it without understanding how it works. See [Rules: Manipulating Precedence, page 5-41](#) for details.
- Step 6** **When**—Applications in any of the following selected classes

Select one or more preconfigured application classes here to indicate the application(s) whose access to the selected registry keys you want to exercise control over.

Note that the entry, <All Applications>, is selected by default. You can use this default or you can unselect it and create your own application classes. For application class configuration details, see [Chapter 8, “Using Application Classes”](#).

- **But not in the following class**—Optionally, select application classes here to exclude from the application class(es) you’ve selected in the included applications field. Note that the entry <None> is selected by default.

Step 7 Attempt to write to any of these registry entries

Click the **Insert Registry Set** link to select one or more pre-configured registry sets for this rule. See [Included Registry Sets, page 9-23](#) for details on included operating system registry values.



Note

You cannot enter registry literals here. You must create a registry set variable if you are not using pre-configured registry sets.

Step 8 When you are finished configuring your Registry access control rule, click the **Save** button.

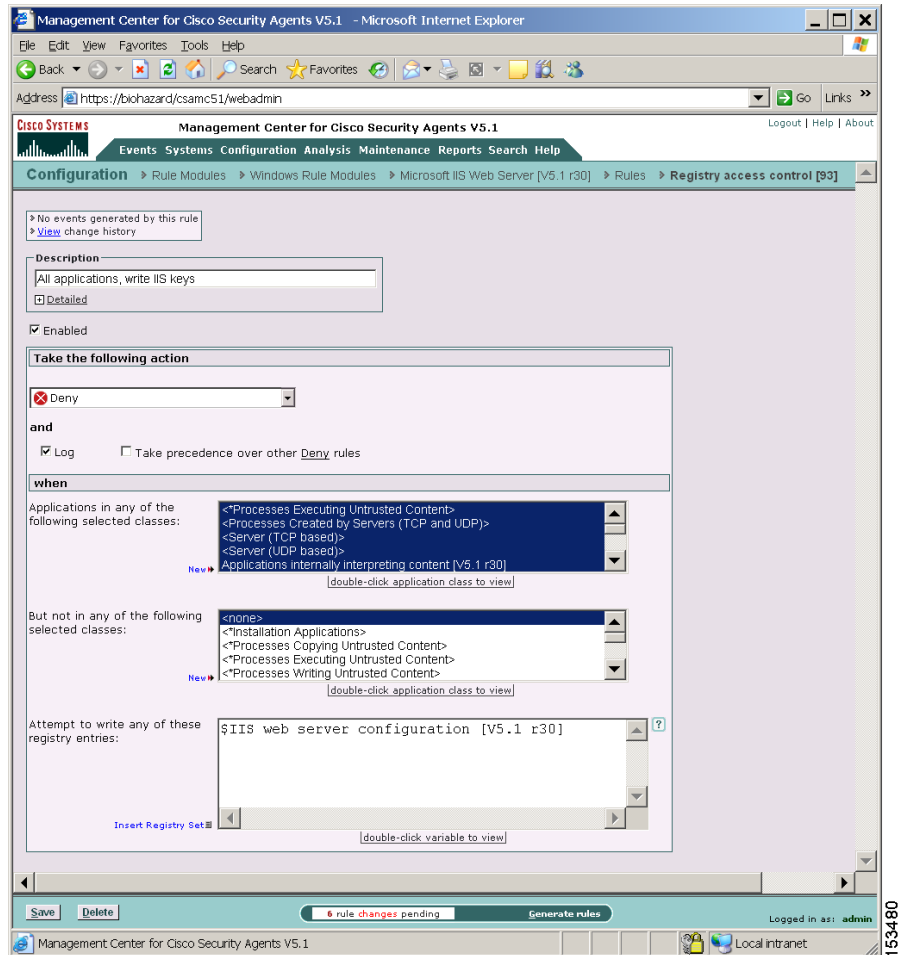
This rule is now part of your rule module. It takes effect when the policy to which it is associated is attached to a group and then downloaded by an agent on the network.



Caution

In order to distribute rules to the correct hosts, you must associate policies with groups and then Generate rules. See [page 5-34](#) for instructions.

Figure 6-15 Registry Access Control Rule



153480

Service Restart

Use the Service restart rule to have the agent restart Windows NT services that have gone down on a system or are simply not responding to service requests.

**Note**

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Service restart** rule. This takes you to the configuration view for this rule type (see [Figure 6-16](#)).
- Step 3** Enter the following information:
- **Description**—Enter a description of this rule.
This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.)
By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.
 - **Log**—Enable this checkbox to turn logging on for this rule.
- Step 4** **Restart the following service**
- Enter a service here you want the agent to automatically restart should it go down for any reason. When entering services here, use the syntax found in the following locations:
- On Windows XP and Windows 2003 and 2000: Start>Settings>Control Panel>Administrative Tools>Services "Name" field
 - On Windows NT: Start>Settings>Control Panel>Services "Service" field

Step 5 When

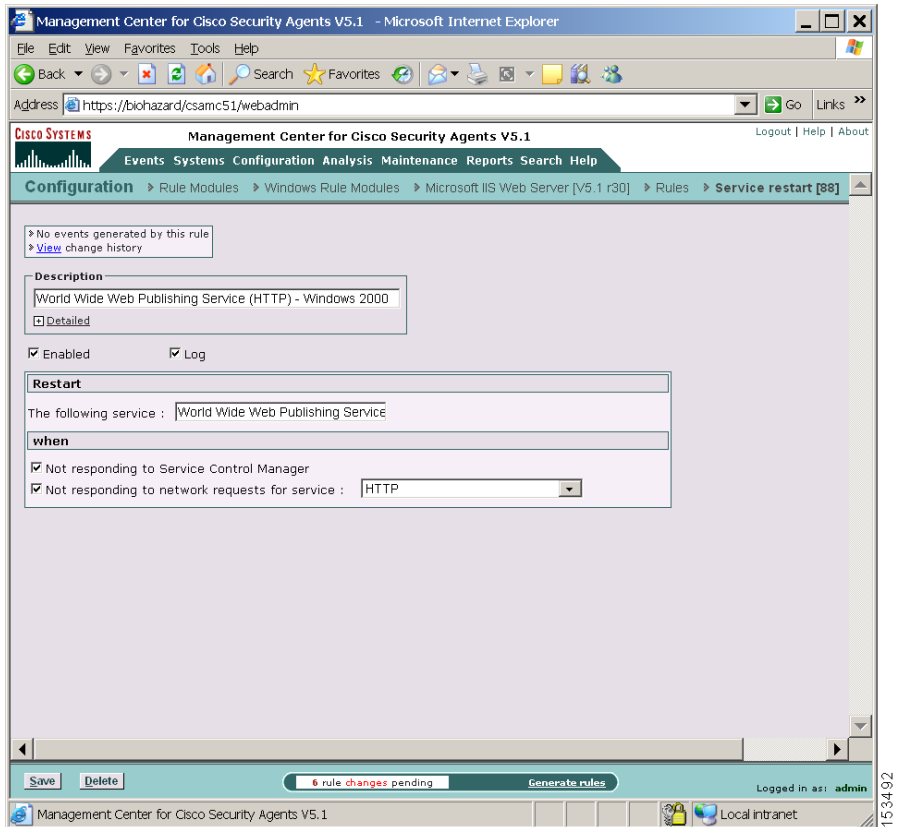
Select one or both of the following checkboxes.

- Not responding to Service Control Manager The Windows Service Control Manager checks the status of system services and recognizes when a service is not responding. Selecting this checkbox causes the Cisco Security Agent to restart the specified service when it does not respond to the Windows Service Control Manager.
- Not responding to network requests for service: Select this checkbox and then choose a network service (such as HTTP) from the available pulldown list. The Cisco Security Agent will monitor whether the system is responding to network requests for the protocols in the network service. If not, it will restart the Windows NT service specified in this rule.

Step 6 Click **Save** when finished.**Note**

The Service Restart rule is different from the Windows NT configurable restart service. Windows NT only restarts processes that have gone away. The agent restarts a process that experiences a failure of any kind.

Figure 6-16 Service Restart Rule



Sniffer and Protocol Detection

Use the Sniffer and protocol detection rule to cause an event to be logged when non-IP protocols and packet sniffer programs are detected running on systems.

Non-IP protocols, such as IPX, AppleTalk, and NetBEUI, are used to provide distributed computing workgroup functions between server and clients and/or sharing between peer clients.

A packet sniffer (also controlled by this rule type) is a program that monitors and analyzes network traffic. Using this information, a network manager can troubleshoot network problems. A sniffer can also be used illegitimately to capture data being transmitted on a network. Sensitive information such as login names and passwords can be extracted from this data and used to break into systems.

The Sniffer and protocol detection rule is a monitoring tool. By adding this rule to a policy, you are causing an event to be logged when any non-IP protocols and packet sniffer programs are detected running on systems which receive this rule.

**Note**

You can use the Sniffer and protocol detection rule page to configure exceptions to this monitoring rule. If you select any non-IP protocols or enter any packet sniffer programs here, you are allowing them to run on systems without generating events. Only non-IP protocols and packet sniffer programs which you explicitly exclude as part of the rule will not cause events to be logged. Otherwise, all are monitored when you add this rule to a policy.

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

-
- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Sniffer and protocol detection** rule. This takes you to the configuration view for this rule type (see [Figure 6-15](#)).

Step 3 Enter the following information:

- **Description**—Enter a description of this rule.
This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
- **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.)
By not selecting this checkbox, you can save this rule, but it will not be active in the policy and it will not be distributed to groups.

Step 4 Select one or more preconfigured **Standard protocols** here to be excluded as part of this rule. The protocols you select here are the only non-IP protocols that will not generate events when they are detected.

If the non-IP protocol(s) you want to exclude are not included in the Standard Protocols list, enter your own in the **Non-standard protocols and packet sniffers** text field. By default, TCP/IP Protocol is already excluded.

This is also where you should enter any packet sniffer programs you want to exclude from this rule. (Find the names for these programs in Cisco Security Agent log files or in system registries.) For example, enter:

PacketDriver

In this example, Windump is the application. The libcap packet capture driver registers using the name PacketDriver.

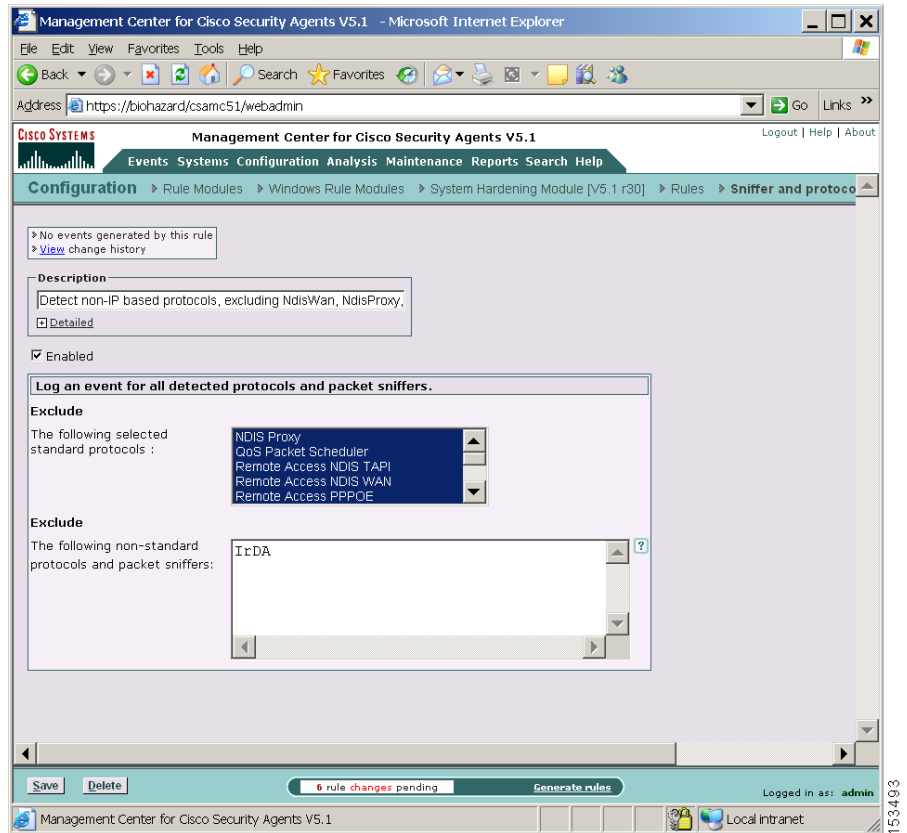
Step 5 Click the **Save** button.



Note

If you have multiple sniffer and protocol detection rules, the exceptions are combined.

Figure 6-17 Sniffer and Protocol Detection Rule



153433

UNIX Only Rules

The following rules are only available for UNIX Rule Modules.

Buffer Overflow Rule

A buffer overflow is what happens when two conditions are met: Firstly, an application is coded in a manner such that it trusts that all users of that application will provide the application with reasonable and expected data. Secondly, the application is provided larger quantities of data than it is capable of correctly handling. When these events come together, an application can behave in unexpected and unintentional ways.

For applications with special privileges, this can result in external users gaining access to machine resources and privileges which they normally would not be able to acquire. In other words, a hostile, network-based attack on a privileged, trusted application via buffer overflows can result in undesirable parties gaining access to your system.

In the case of UNIX operating systems, there are three distinct types of buffer overruns which can occur, based upon the type of memory space involved: stack, data, and heap.

- Stack space is used to store data and information which is local to the piece of code currently being executed in an application, and contains stored away control flow information for the application.
- Data space is used to store data with fixed sizes which needs to be shared among different parts of an application. Often, content in data space has been given initial values.
- Heap space is dynamically given out to applications, with the intent that it is relatively short-lived, of varying size based upon the input datasets, and is frequently visible to numerous sub-components of an application.

**Note**

This rule is UNIX specific. Some corresponding Windows functionality is available from the System API control rule page.

Configure the Buffer overflow rule as follows.

- Step 1** To add rules to your policy, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Buffer overflow** rule. This takes you to the configuration view for this rule type (Figure 6-18).
- Step 3** In the Buffer overflow rule configuration view, enter the following information:
- **Description**—Enter a description of this rule. This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.) By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.
- Step 4** **Take the following action**—Select an action type from the pulldown list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#). (Note that Priority Deny and Deny actions are not available for this rule.)
- Step 5** and
- **Log**—Enable this checkbox to turn logging on for this rule. Generally, you will want to turn logging on for all deny rules. This means that the denied system action in question is logged and sent to the server at regular time intervals.
 - **Take precedence over other <action type> rules**—Enable this checkbox to manipulate rule precedence so that this rule is evaluated before other similar rules. You should generally NOT require this checkbox. Do not use it without understanding how it works. See [Rules: Manipulating Precedence, page 5-41](#) for details.
- Step 6** **when**
- Applications in any of the following selected classes**
- Select *one or more* preconfigured application classes here. Note that the entry, <All Applications>, is selected by default. You can use this default or you can unselect it and create your own application classes. For application class configuration details, see [Chapter 8, “Using Application Classes”](#).
- But not in the following class**—Optionally, select application classes here to exclude from the application class(es) you’ve selected in the included applications field. Note that the entry <None> is selected by default.

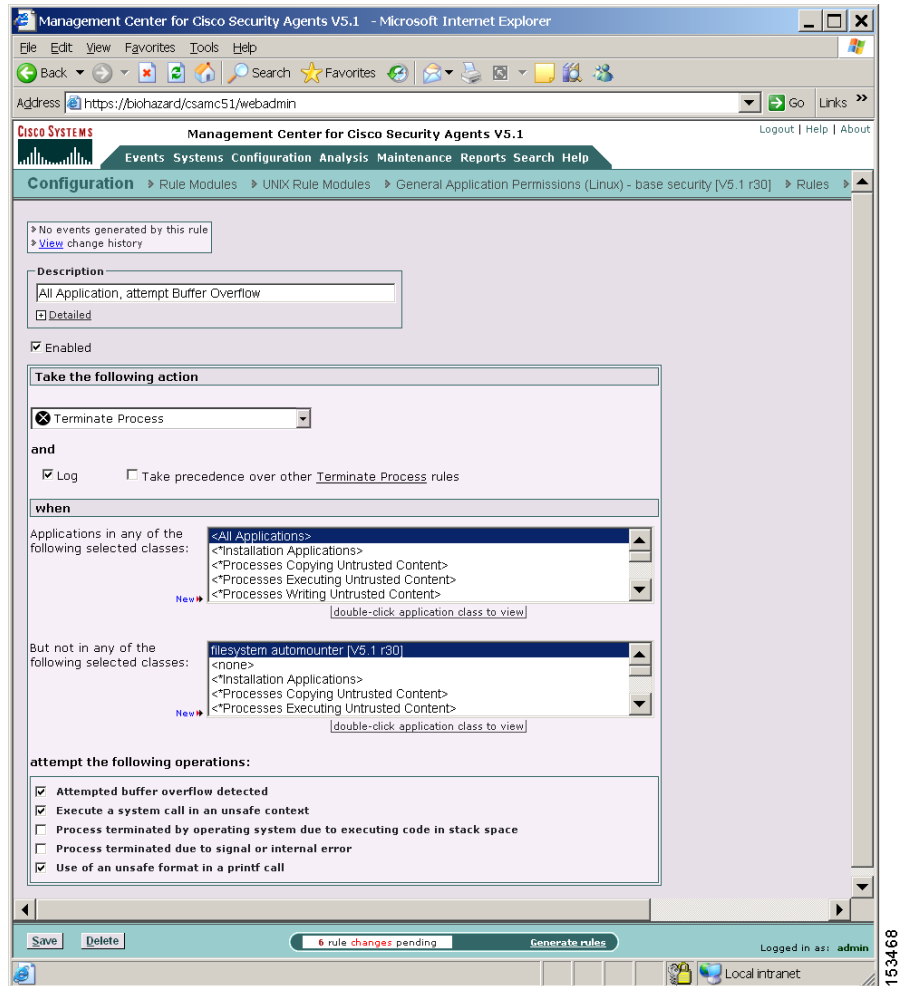
- Step 7** Select one or more of the following checkboxes to prevent the associated buffer overflow attack from occurring.
- **Attempted buffer overflow detected**
Enable this checkbox to detect buffer overflow conditions which occur in UNIX executables. This feature provides protection from stack buffer overflows to a number of commonly used libc routines. As a large number of attacks on UNIX systems are based upon buffer overflow attacks, it is recommended that you enable this feature. Processes terminated by operating system due to executing code in stack space.
 - **Executing a system call in an unsafe context**
Use this checkbox to prevent certain system calls (e.g. those which grant extra privileges or start new processes) from occurring if they are invoked in an unsafe manner, or if they appear to have come from a corrupted or invalid context.
 - **Processes terminated by operating system due to executing code in stack space**
This checkbox enables the "noexec_user_stack" system variable for all processes or for processes added to the <*Processes requiring OS Stack Execution Protection>. See [Built-in Configurable Application Classes, page 8-7](#) for details. This checkbox monitors the execution of instructions from stack memory. This only provides logging.
 - **Process terminated due to signal or internal error**
Processes can be killed on a system by either another process or by an internal error occurring on the system. This checkbox causes the agent to monitor when this occurs. The only action type available when this checkbox is enabled is Monitor.
 - **Use of an unsafe format in a printf call**
Use this checkbox to prevent the usage of the '%n' *printf() format qualifier. Numerous attacks utilize the '%n' format on *printf() routines to gain access to program control flow information.

You also have the ability to select specific **application classes to exclude** from the various Buffer overflow types you designate. If you select an application in the available list beside a checkbox rule, that rule does not apply to the selected application class. If you have multiple, similar Buffer overflow rules, the application class exceptions are combined.

**Note**

Refer to [Replicate Feature, page 6-44](#) for details on easily propagating the changes you make to one Buffer overflow rule to other Buffer overflow rules in other policies.

Figure 6-18 Buffer Overflow Rule



Network Interface Control

Use the Network interface control rule to specify whether applications can open a device and act as a sniffer (promiscuous mode). A packet sniffer is a program that monitors and analyzes network traffic. Using this information, a network manager can troubleshoot network problems. A sniffer can also be used illegitimately to capture data being transmitted on a network. Sensitive information such as login names and passwords can be extracted from this data and used to break into systems.

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

-
- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Network interface control** rule. This takes you to the configuration view for this rule type (see [Figure 6-19](#)).
- Step 3** Enter the following information:
- **Description**—Enter a description of this rule.
This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.)
By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.
- Step 4** **Take the following action**—Select an action type from the pulldown list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#).
- Step 5** and
- **Log**—Enable this checkbox to turn logging on for this rule. Generally, you will want to turn logging on for all deny rules. This means that the denied system action in question is logged and sent to the server at regular time intervals.
 - **Take precedence over other <action type> rules**—Enable this checkbox to manipulate policy rule precedence so that this rule is evaluated before other similar rules. You should generally NOT require this checkbox. Do not use it without understanding how it works. See [Rules: Manipulating Precedence, page 5-41](#) for details.

Step 6 When—Applications in any of the following selected classes

Select one or more preconfigured application classes here to indicate the application(s) whose access to the selected resource(s) want to exercise control over.

Note that the entry, <All Applications>, is selected by default. You can use this default or you can unselect it and create your own application classes. For application class configuration details, see [Chapter 8, “Using Application Classes”](#).

Step 7 Attempt the following operations

Select one or more of the following checkboxes:

- Open a stream connection to the NIC driver



Note Open a stream connection to the NIC driver - For Linux systems, this only applies to modification of the interface characteristics, e.g. using ifconfig to modify an interface's network mask. This does not apply to simply reading the interface characteristics.

- Put the NIC into promiscuous mode



Note If you have selected the Allow radio button, when you select to "Put the NIC into promiscuous mode", the "Open a stream connection to the NIC driver" checkbox is also automatically selected. It must be enabled for promiscuous mode to work.

Conversely, if you have selected a Deny radio button, when you select the "Open a stream connection to the NIC driver" checkbox, the "Put the NIC into promiscuous mode" checkbox is also automatically selected. If you deny one, the other is automatically denied as well.

Step 8 When you are finished configuring your rule, click the **Save** button.

**Note**

If you are using remote management tools and you are configuring a Network interface control rule to deny "all applications" from opening a stream connection to the NIC and operating in promiscuous mode, you may want to make an exception for the remote management application (if you want to run snoop).

Figure 6-19 Network Interface Control Rule

The screenshot displays the Management Center for Cisco Security Agents V5.1 web interface. The browser window title is "Management Center for Cisco Security Agents V5.1 - Microsoft Internet Explorer". The address bar shows "https://biohazard/csamc51/webadmin". The page header includes "Management Center for Cisco Security Agents V5.1" and navigation links for "Events", "Systems", "Configuration", "Analysis", "Maintenance", "Reports", "Search", and "Help". The breadcrumb trail is "Configuration > Rule Modules > UNIX Rule Modules > Required System Module (Linux) [V5.1 r30] > Rules > Network interf...".

The main content area shows the configuration for a rule. It includes a "Description" field with the text "route application, open and configure NIC" and a "Detailed" checkbox. The "Enabled" checkbox is checked. Under "Take the following action", the "Priority Allow" dropdown is selected, and the "Log" and "Take precedence over other Priority Allow rules" checkboxes are unchecked. The "when" section lists "Applications in any of the following selected classes:" with a list box containing "route application (Linux) [V5.1 r30]", "<All Applications>", "<Installation Applications>", "<*Processes Copying Untrusted Content>", and "<*Processes Executing Untrusted Content>". A "New" link and a "double-click application class to view" instruction are also present. Under "attempt the following operations", the "Open a stream connection to the NIC driver" checkbox is checked, and the "Put the NIC into promiscuous mode" checkbox is unchecked.

The bottom of the interface features a "Save" button, a "Delete" button, a status bar indicating "6 rule changes pending", a "Generate rules" button, and a "Logged in as: admin" indicator. The footer shows "Management Center for Cisco Security Agents V5.1" and "Local intranet". A vertical page number "153478" is visible on the right side.

Resource Access Control

Use the Resource access control rule to protect systems from symbolic link attacks. In this type of attack, an attacker attempts to determine the name of a temporary file prior to its creation by a known application. If the name is determined correctly, the attacker could then create a symbolic link to the target file for which the user of the application has write permissions. The application process would then overwrite the contents of the target file with its own output when it tries to write the named temporary file.

For example, a directory such as /tmp is writable by everyone. An attacker could create a symbolic link in this directory to a protected file such as /etc/shadow. A superuser process may then unwittingly write to or copy from /etc/shadow. This would then grant the attacker access to this sensitive information via a symbolic link from the /tmp directory.

By enabling the resource access control rule, you can prevent "suspicious" symbolic links from being followed. A suspicious symbolic link is one that meets all of the following criteria:

- The parent directory is a temporary directory such as /tmp and /usr/tmp
- The symbolic link's owner is different from the parent directory's owner
- The symbolic link's owner is different from the effective UID of the process

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

-
- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Resource access control** rule. This takes you to the configuration view for this rule type (see [Figure 6-21](#)).
- Step 3** Enter the following information:
- **Description**—Enter a description of this rule.
This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.)
By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.
- Step 4** Select the **Symbolic Link Protection** checkbox to turn on that functionality.

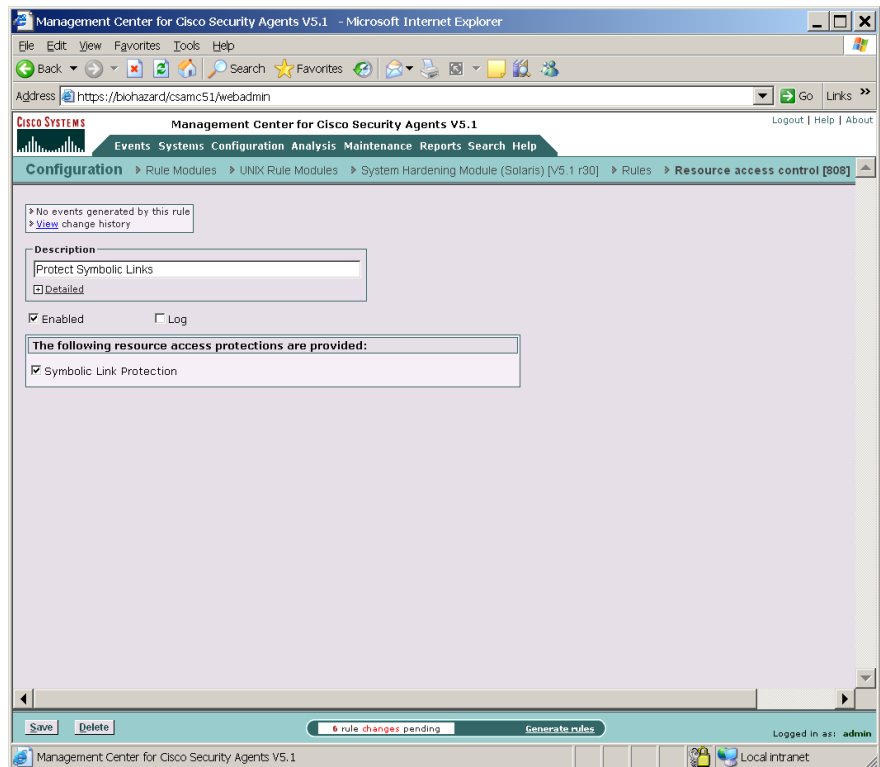
Step 5 Click the **Save** button.



Caution

Symbolic Links: If you create a File access control rule to protect a symbolic link, **ONLY** that symbolic link is protected. The underlying resource, unless also specified, is **NOT** protected. For example, a File access control rule written for `/etc/hosts` does not protect `/etc/inet/hosts`. Similarly, a File access control rule written for `/etc/inet/hosts` does not protect `/etc/hosts`. If you want to protect a symbolic link and its underlying resource, both must be specified in the rule.

Figure 6-20 Resource Access Control Rule



153490

Rootkit / kernel Protection

Use the Rootkit / kernel protection rule to control unauthorized access to the operating system. In effect, this rule controls drivers attempting to dynamically load after boot time. You can use to this rule to specify authorized drivers that you are allowing to load any time after the system is finished booting.



Note

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

-
- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Rootkit / kernel protection** rule. This takes you to the configuration view for this rule type (see [Figure 6-21](#)).
- Step 3** Enter the following information:
- **Description**—Enter a description of this rule.
This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.)
By not selecting this checkbox, you can save this rule, but it will not be active in the policy and it will not be distributed to groups.
- Step 4** **Take the following action**—Select an action type from the pulldown list. For further details on rule action types, see [Rules: Action Options and Precedence, page 5-37](#).
- Step 5** and
- **Log**—Enable this checkbox to turn logging on for this rule. Generally, you will want to turn logging on for all deny rules. This means that the denied system action in question is logged and sent to the server at regular time intervals.
 - **Take precedence over other <action type> rules**—Enable this checkbox to manipulate policy rule precedence so that this rule is evaluated before other similar rules. You should generally NOT require this checkbox. Do not use it without understanding how it works. See [Rules: Manipulating Precedence, page 5-41](#) for details.

Step 6 When—Applications in any of the following selected classes

Select one or more preconfigured application classes here to indicate the application(s) whose access to the selected resource(s) want to exercise control over.

Note that the entry, <All Applications>, is selected by default. You can use this default or you can unselect it and create your own application classes. For application class configuration details, see [Chapter 8, “Using Application Classes”](#).

- **But not in any of the selected classes**—Optionally, select application classes here to exclude from the application class(es) you’ve selected in the included applications field. Note that the entry <None> is selected by default.

Step 7 Attempt to load the following modules

By default, this field contains <none> which indicates no specified drivers. Enter the names of drivers you want to specify for this the rule and therefore allow, deny, or monitor the loading of at any time.

**Caution**

If you enter file sets which use "content-matching" constraints, via the Insert File Set link, the content-matching constraints are ignored.

- **The previous detected boot was insecure**

When a system has previously booted in a non-standard manner, selecting this checkbox causes a message to be sent to the event log. A boot is considered standard if the system was booted from the primary hard disk. Any other boot type, for example, booting from a peripheral device (CD ROM) or a hard disk that is not the primary, is considered non-standard. A non-standard boot may be considered suspicious. (e.g., This is one way of circumventing the Cisco Security Agent and introducing a Trojan to a system.)

The insecure boot detection this checkbox enables works in conjunction with a particular type of compatible BIOS on compliant systems. The compatible BIOS detects a non-standard boot and on the next normal boot, if you have this checkbox selected, a message is sent to the MC which logs this insecure boot detection. (If a Host is running a BIOS that supports this insecure boot detection feature, the individual Host details page will indicate this. From the CSA MC menu bar, click Systems>Hosts. Then click on the link for an individual host. The Host Status section includes a category named "BIOS supported boot detection".)

– **Included boot patterns**

By default, this field contains <all>. You can use provided tokens here to only detect certain boot types as insecure or to exclude a particular boot type from the detection. Available tokens are:

@fixed - This indicates a fixed hard disk that is not the primary disk. (Compatible BIOS is required to detect this.)

@network - This indicates all network shares. (Compatible BIOS is required to detect this.)

@removable - This indicates all removable media. That includes, floppies, CDs, zip drives, etc. (Compatible BIOS is required to detect this.)

Step 8 Click the **Save** button.

Figure 6-21 Rootkit / kernel Protection Rule

The screenshot displays the Management Center for Cisco Security Agents V5.1 interface. The browser address bar shows `https://bchazard/csamc51/webadmin`. The page title is "Management Center for Cisco Security Agents V5.1". The navigation menu includes "Events", "Systems", "Configuration", "Analysis", "Maintenance", "Reports", and "Search Help". The current configuration path is "Configuration > Rule Modules > UNIX Rule Modules > Required System Module (Linux) [V5.1 r3D] > Rules > Rootkit / kernel protection [73]".

The rule configuration details are as follows:

- Description:** Allow post-boot dynamic kernel module loading. Detailed
- Enabled
- Take the following action:** Priority Allow
- and:** Log Take precedence over other Priority Allow rules
- when:**
 - Modules load after system startup
 - Applications in any of the following selected classes:
 - insmod [V5.1 r3D]
 - <All Applications>
 - <Installation Applications>
 - <Processes Copying Untrusted Content>
 - <Processes Executing Untrusted Content>
 - But not in any of the following selected classes:
 - <none>
 - <Installation Applications>
 - <Processes Copying Untrusted Content>
 - <Processes Executing Untrusted Content>
 - <Processes Writing Untrusted Content>
 - Attempt to load the following modules: `/lib/modules/2.4*/kernel/**/*`
 - The previous detected boot was insecure

At the bottom of the configuration area, there are "Save" and "Delete" buttons, a status bar indicating "6 rule changes pending", and a "Generate rules" button. The user is logged in as "admin".

153491

Syslog Control

Use the Syslog control rule to have specified Solaris and Linux Syslog items appear in the CSA MC Event Log for selected groups.



Note

These instructions are a continuation of [Configuring Rule Modules, page 5-5](#).

- Step 1** To add rules to your module, click the **Add rule** link at the bottom of the rule list. A pop-up list of the available rule types appears.
- Step 2** Select the **Syslog control** rule. This takes you to the configuration view for this rule type (see [Figure 6-22](#)).
- Step 3** Enter the following information:
- **Description**—Enter a description of this rule.
This description appears in the list view for the module. Optionally, expand the **+Detailed** field to enter a longer description.
 - **Enabled**—Use this checkbox to enable this rule within the module. (It is enabled by default.)
By not selecting this checkbox, you can save this rule, but it will not be active in the module and it will not be distributed to groups.
- Step 4** **Log events from syslog**
- **Include events matching the following**—Select this radio button to specify the criteria for Syslog entries which you want to appear in the CSA MC Event Log.
 - **Include all events except those matching the following**—Select this radio button to specify the criteria for Syslog entries which you do not want to appear in the CSA MC Event Log. (All criteria not specified here will appear in the CSA MC Event Log.)



Note

You can configure CSA MC to correlate Syslog events logged across multiple systems. See [Installation Applications Policy, page 7-4](#).

- Step 5** **Criteria** (You should select at least one for the rule to have any effect.)

- **Event Source**—In the text field, enter (one per line) event source parameters you want to filter by.
The event source is the software that logged the event, which can be an application name such as `/sbin/dhcpagent`, a kernel level driver module such as `scsi`, or the `unix` kernel itself.
- **Facility**—Select one or more items from the list box you want to appear (first radio button above) or which entries you want to not appear (second radio button above) in CSA MC Event Logs.
- **Priority**—Select one more checkboxes by which to filter the viewing of events according to priority. If you select no checkboxes, all priorities are included in the rule.
- **Message Pattern**—In the text field, enter (one per line) message patterns you want to match and filter by. To match, the string you enter must literally appear somewhere within the message.

Step 6 Click the **Save** button.



Note

On Linux platforms, the default `syslogd` does not embed the facility or priority level in the syslog messages. Using a different `syslogd`, such as `syslog-ng`, with correct message formatting, it is possible to use the facility and/or priority levels to report these events. Therefore, if `syslog-ng` is used, the message template must take the following form:

```
template("$DATE $HOST $PROGRAM: [ID 0 $FACILITY.$LEVEL] $MSG\n")
```

For example, the entry for content recorded into `/var/log/messages` would appear as follows:

```
destination d_1 {
file("/var/log/messages" create_dirs(yes) template("$DATE $HOST
$PROGRAM: [ID 0 $FACILITY.$LEVEL]
$MSG\n")); };
```

General Syslog rule configuration examples

For Example:

Configure a syslog rule to log warning messages such as the one listed here:

```
Apr 29 13:46:35 myhost /sbin/dhcpagent[39]: [ID 929444
daemon.warning] configure_if: no IP broadcast specified
for eri0
```

To get every message of category "warning" from the /sbin/dhcpagent daemon, you would configure your syslog rule in the following manner (See [Figure 6-22](#)):

Select the "Include events matching the following" radio button and enter:

- Facility: daemon
- Event Source: /sbin/dhcpagent
- Priority: Warning checkbox
- Message Pattern: <all>

For Example:

Configure a syslog rule to log failed su root attempts such as the one listed here:

```
Apr 29 13:49:23 myhost su: [ID 810491 auth.crit] 'su
root' failed for haxor on /dev/pts/4
```

To get messages for failed su root attempts, you would configure your syslog rule in the following manner:

Select the "Include events matching the following" radio button and enter:

- Facility: auth
- Event Source: su
- Priority: Alert and Above checkbox
- Message Pattern: root

For Example:

Configure a syslog rule to include all events but exclude all lockstat-related messages such as the one listed here:

```
Apr 29 13:46:43 myhost genunix: [ID 936769 kern.info]
lockstat0 is /pseudo/lockstat@0
```

To log all events except for lockstat-related messages, configure your rule in the following manner:

Select the "Include events except those matching the following" radio button and enter:

- Facility: kern
- Event Source: <all>
- Priority: all checkboxes
- Message Pattern: lockstat

Figure 6-22 Syslog Control Rule

