Blue Coat® Systems SG™ Appliance

Volume 6: The Visual Policy Manager and Advanced Policy Tasks

SGOS Version 5.2.2

Blue Coat

Contact Information

Blue Coat Systems Inc.
420 North Mary Ave
Sunnyvale, CA 94085-4121

http://www.bluecoat.com/support/contact.html

bcs.info@bluecoat.com
http://www.bluecoat.com

For concerns or feedback about the documentation: documentation@bluecoat.com

Document Number: 231-02843
Document Revision: SGOS 5.2.2—09/2007

Contents

Chapter 1: Introduction

Creating policy is the core task of implementing Blue Coat SG appliances into the enterprise. After the basic SG appliance configurations are complete, defined policy is what controls user activities and implements company authentication and network resource allocation goals.

The Visual Policy Manager is a user interface that creates underlying Blue Coat Content Policy Language (CPL). In the VPM, you create policy layers by selecting and customizing policy *objects*. This volume discusses the facets of the VPM, including layer interactions and summary object descriptions. When approrpriate, cross references are provided to other Blue Coat volumes that describe the conceptual information of the feature. This volume also contains a chapter that discusses some common tasks that are only achieved through policy, not the Management Console.

This document contains the following chapters:

- Chapter 2: "Managing Policy Files" on page 15
- Chapter 3: "The Visual Policy Manager" on page 27
- Chapter 4: "Advanced Policy Tasks" on page 169

Document Conventions

The following section lists the typographical and Command Line Interface (CLI) syntax conventions used in this manual.

Table 1-1. Document Conventions

Conventions	Definition
Italics	The first use of a new or Blue Coat-proprietary term.
`Courier font`	Command line text that appears on your administrator workstation.
`Courier Italics`	A command line variable that is to be substituted with a literal name or value pertaining to the appropriate facet of your network system.
`Courier Boldface`	A Blue Coat literal to be entered as shown.
{ }	One of the parameters enclosed within the braces must be supplied
[ ]	An optional parameter or parameters.
|	Either the parameter before or after the pipe character can or must be selected, but not both.

Chapter 2: Managing Policy Files

Policy files contain the policies (triggers and actions) that manage every aspect of the SG appliance, from controlling user authentication and privileges to disabling access logging or determining the version of SOCKS.

The policy for a given system can contain several files with many layers and rules in each. Policies can be defined through the Visual Policy Manager (VPM) or composed in Content Policy Language (CPL). (Some advanced policy features are not available in VPM and can only be configured through CPL.)

Policies are managed through four files:

- Central policy file—Contains global settings to improve performance and behavior and filters for important and emerging viruses (such as Code Red and Nimda). This file is usually managed by Blue Coat, although you can point the SG appliance to a custom Central policy file instead.
- Forward policy file—Usually used to supplement any policy created in the other three policy files. The Forward policy file contains Forwarding rules when the system is upgraded from a previous version of SGOS (2.x) or CacheOS (4.x).
- Local policy file—A file you create yourself. When the VPM is not the primary tool used to define policy, the Local file contains the majority of the policy rules for a system. If the VPM is the primary tool, this file is either empty or includes rules for advanced policy features that are not available in VPM.
- Visual Policy Manager—The policy created by the VPM can either supplement or override the policies created in the other policy files.

This chapter contains the following sections:

To learn about writing policies, refer to *Volume 10: Blue Coat SG Appliance Content Policy Language Guide*.

Creating and Editing Policy Files

You can create and edit policy files two ways:

- Through the Management console (recommended).
- Through the CLI inline policy command (not recommended because the policies can grow large and using `inline policy` overwrites any existing policy on the SG appliance).

Using the Management Console

You can install the policy files with the following methods:

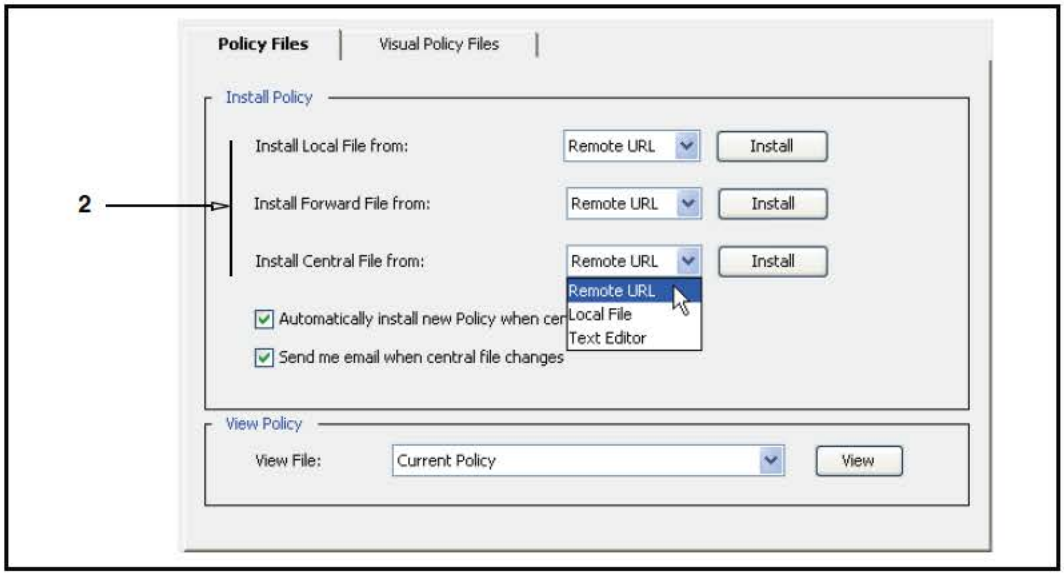
- Using the SG appliance Text Editor, which allows you to enter directives (or copy and paste the contents of an already-created file) directly onto the SG appliance.
- Creating a file on your local system; the SG appliance can browse to the file and install it.
- Using a remote URL, where you place an already-created file on an FTP or HTTP server to be downloaded to the SG appliance.

The SG appliance compiles the new policy from all source files and installs the policy, if the compilation is successful.

Important: If errors or warnings are produced when you load the policy file, a summary of the errors and/or warnings is displayed automatically. If errors are present, the policy file is not installed. If warnings are present, the policy file is installed, but the warnings should be examined.

To define and install policy files directly:

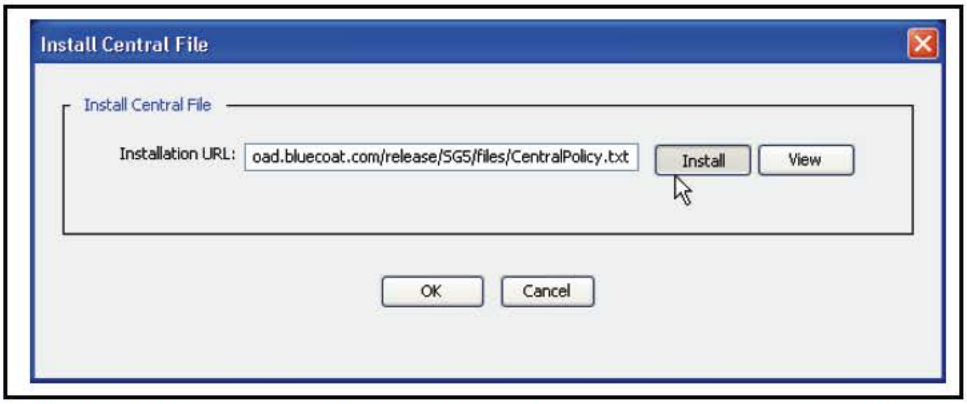
1. Select **Configuration > Policy > Policy Files > Policy Files.**



2. From the **Install Local/Forward/Central File from** drop-down list, select the method used to install the local, forward, or central policy configuration; click **Install** and complete one of the three procedures below:

Note: A message is written to the event log when you install a list through the SG appliance.

- Installing a policy file using a Remote URL:



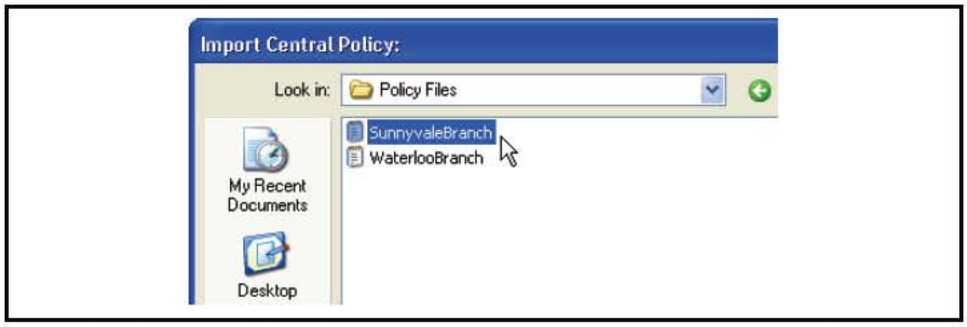
In the Install Local/Forward/Central File dialog that appears, enter the fully-qualified URL, including the filename, where the policy configuration is located. To view the file before installing it, click **View**. Click **Install**. The **Installation Status** field summarizes the results; click **Results** to open the policy installation results window. Close the window when you are finished viewing the results; click **OK** in the Install Local/Forward/Central File dialog.

Note: If you use the default Blue Coat Central policy file, load it from: `https://download.bluecoat.com/release/SG5/files/CentralPolicy.txt`.

If you install a Central policy file, the default is already entered; change this field only if you want to create a custom Central policy file.

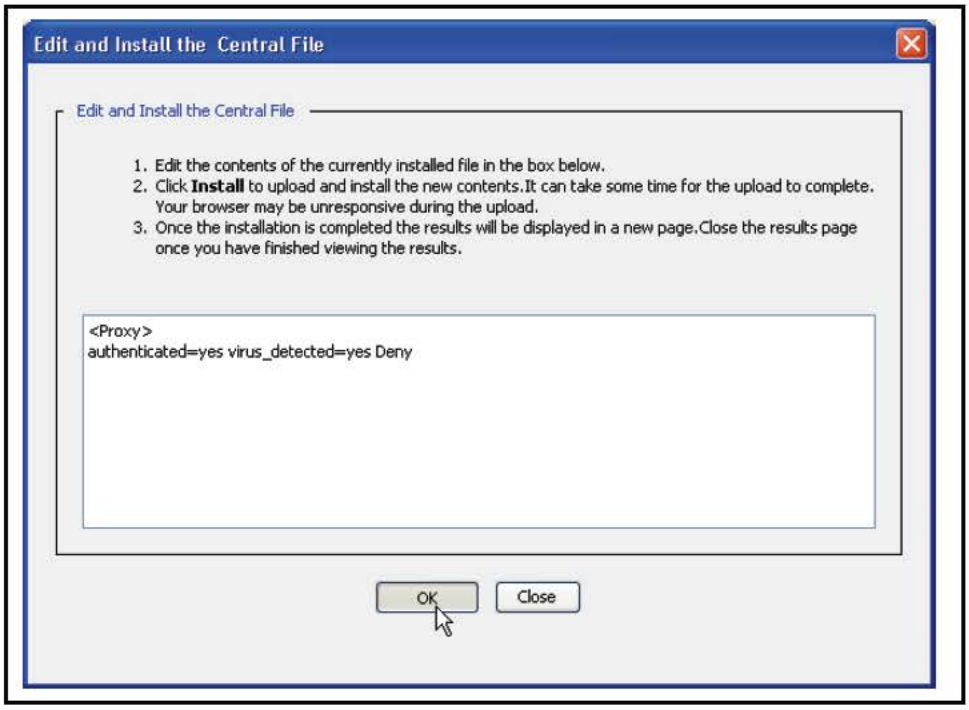
To load a Forward, Local, or a custom Central policy file, move it to an HTTP or FTP server, and then use that URL to download the file to the SG appliance.

- Installing a policy file using a Local File:



In the Upload and Install File window that opens, either enter the path to the file into the **File to upload** field, or click **Browse** to display the Choose file dialog, locate the file on the local system, and open it. Click **Install**. When the installation is complete, the installation results display. View the results and close the window.

- Installing a policy file using the SG appliance Text Editor:



The current configuration is displayed in installable list format. Define the policy rules using CPL in the Edit and Install File window that opens (refer to *Volume 10: Blue Coat SG Appliance Content Policy Language Guide*); click **Install**. When the installation is complete, a results window opens. View the results, close the results window and click **OK** in the Edit and Install File window.

3. Click **Apply**.

Note: There are other management-related tasks regarding the Blue Coat Central Policy File. See "Managing the Central Policy File" on page 22.

Using the CLI Inline Command

To create policies using the CLI, you can use the SG appliance `inline policy` command. This command either creates a new policy file or, if the specified file already exists, overwrites an existing policy file. You cannot edit an existing policy file using this command.

Note: If you are not sure whether a policy file is already defined, check before using the `inline policy` command. For more information, see "Viewing Policy Source Files" on page 24.

To create policy files:

1. At the `(config)` command prompt, enter the following command:

   ```
   SGOS#(config) inline policy file end-of-input-marker
   ```

   where *`file`* specifies the type of policy you want to define: `Central` (Central policy file), `Forward` (Forward policy file), or `local` (local policy file).

Note: Do not use the `inline policy` command with files created using the VPM module.

`end-of-file-marker`—Specifies the string that marks the end of the current `inline` command input; `eof` usually works as a string. The CLI buffers all input until you enter the marker string.

2. Define the policy rules using CPL (refer to *Volume 10: Blue Coat SG Appliance Content Policy Language Guide*).

   Enter each line and press <Enter>. To correct mistakes on the current line, use <Backspace>. If a mistake has been made in a line that has already been terminated by <Enter>, exit the `inline policy` command by typing <Ctrl>c to prevent the file from being saved.

3. Enter the *`eof`* marker to save the policies and exit the `inline` mode.

For more information on the `inline` command, refer to *Volume 11: Blue Coat SG Appliance Command Line Reference*.

To load policy files:

At the `(config)` command prompt, enter the following commands:

```
SGOS#(config) policy {forward-path | local-path | central-path} url
SGOS#(config) load policy {forward | local | central}
```

The SG appliance compiles and installs the new policy. The SG appliance might display a warning if the new policy causes conflicts. If a syntax error is found, the appliance displays an error message. For information about these messages, refer to *Volume 10: Blue Coat SG Appliance Content Policy Language Guide*. Correct the error, then reload the file.

Unloading Policy Files

To disable policies, perform the following procedure to unload the compiled policy file from the SG appliance memory. These steps describe how to replace a current policy file with an empty policy file.

To keep a current policy file, either make a backup copy or rename the file before unloading it. By renaming the file, you can later reload the original policy file. If you use multiple policy files, back up or rename files as necessary. Alternatively, rather than use an empty policy file, you can delete the entire contents of the file, then reload it.

To unload policies:

1. Select **Configuration > Policy > Policy Files > Policy Files**.
2. Select **Text Editor** in the **Install Local/Forward/Central File from** drop-down list and click the appropriate **Install** button. The Edit and Install the Local/Forward/Central Policy File appears.
3. Delete the text and click **Install**.
4. View the results in the results page that opens; close the page.
5. Click **Close**.

Configuring Policy Options

This section disrobes the Policy Options screen, which allow you re-order policy evaluation, change the default transaction setting, and enable policy tracing.

Policy File Evaluation

The order in which the SG appliance evaluates policy rules is important. Changes to the evaluation order can result in different effective policy, as the order of policy evaluation defines general rules and exceptions. While this order is configurable, the default and recommended order is:

```
VPM File—Local Policy File—Central Policy File-Forward File
```

This prevents policies in the Central file that block virus signatures from being inadvertently overridden by allow (access-granting) policy rules in the VPM and Local files.

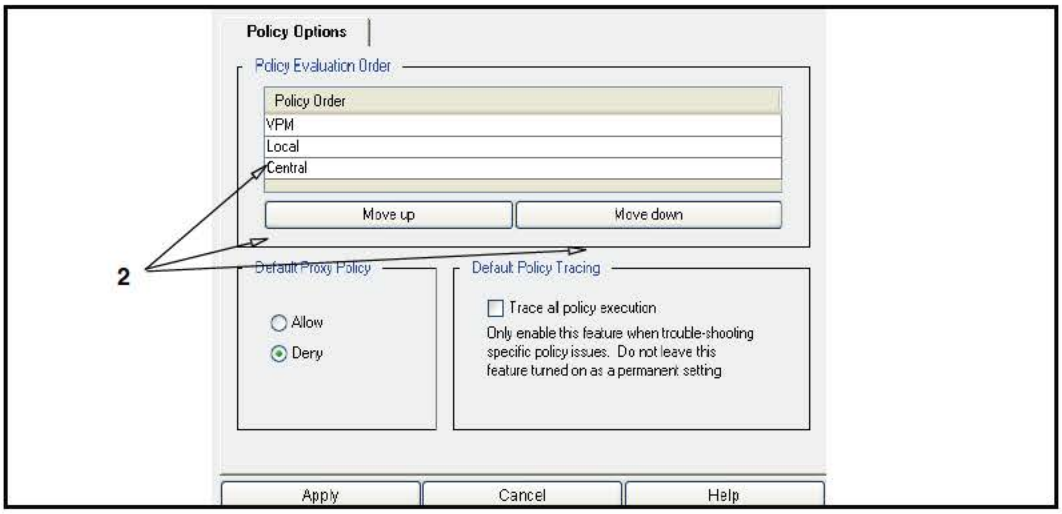
When changing the policy file evaluation order, remember that final decisions can differ because decisions from files later in the order can override decisions from earlier files (the Forward policy file order cannot be changed).

For a new SG appliance, the default evaluation order is: VPM, Local, Central, and Forward.

For an upgraded SG appliance, the policy evaluation order is the order already existing on the appliance before the upgrade.

To change policy order

1. Select **Configuration > Policy > Policy Options**.



2. To change the order, select the file to move and click **Move Up** or **Move Down**. Remember that the last file in the list overwrites decisions in files evaluated earlier.

Transaction Settings: Deny and Allow

The default proxy transaction policy is to either *deny proxy transactions* or to *allow proxy transactions*. A default proxy transaction policy of Deny prohibits proxy-type access to the SG appliance: you must then create policies to explicitly grant access on a case-by-case basis.

A default proxy transaction policy of Allow permits most proxy transactions however, if protocol detection is enabled (the default), HTTP CONNECT transactions are only allowed if they are tunneling SSL. If protocol detection is disabled, HTTP CONNECT is only allowed on port 443. If your policy is set to Allow, you must create policies to explicitly deny access on a case-by-case basis.

Note: The default proxy policy does not apply to admin transactions. By default, admin transactions are denied unless you log in using console account credentials or if explicit policy is written to grant read-only or read-write privileges.

Defaults:

- Proxy Edition: The default depends on how you installed SGOS and if it was a new installation or an upgrade:
  - If you installed the SGOS through a browser using the Initial Configuration Web site, you chose whether to allow or deny proxied transactions during initial configuration.
  - If you installed the SGOS using the front panel or a serial console port, the default setting is Deny.
  - If you upgraded the SGOS from a previous version, the default remains whatever it was for the previous policy.
- MACH5 Edition: The default setting is Allow.

You can always change the setting—see the procedures below for instructions.

Also keep in mind that:

- Changing the default proxy transaction policy affects the basic environment in which the overall policy is evaluated. It is likely that you must revise policies to retain expected behavior after such a change.
- Changes to the evaluation order might result in different effective policy, because the order of policy evaluation defines general rules and exceptions.
- Changing the default proxy transaction policy does not affect the evaluation of cache and admin transactions.

To configure Deny or Allow default proxy policy:

1. Select **Configuration > Policy > Policy Options**.
2. Under Default Proxy Policy, select either **Deny** or **Allow**.
3. Click **Apply** to commit the changes to the SG appliance.

Policy Tracing

Tracing enabled with the Management Console or CLI is global; that is, it records every policy-related event in every layer. It should be used only while troubleshooting. For information on troubleshooting policy, refer to *Volume 10: Blue Coat SG Appliance Content Policy Language Guide*. Turning on policy tracing of any kind is expensive in terms of system resource usage and slows down the SG appliance's ability to handle traffic.

To enable policy tracing:

1. Select **Configuration > Policy > Policy Options**.

2. Select **Trace all policy execution.**
3. Click **Apply**.

Managing the Central Policy File

The Central policy file is updated when needed by Blue Coat. The file can be updated automatically or you can request e-mail notification. You can also configure the path to point to your own custom Central policy file.

Configuring Automatic Installation

You can specify whether the SG appliance checks for a new version of the Central policy file. If a new version exists, the appliance can install it automatically.

Perform the following procedure to configure the SG appliance to check for and install a new version of the Central policy file.

To configure automatic installation:

1. Select **Configuration > Policy > Policy Files > Policy Files**.
2. Select **Automatically install new Policy when central file changes**.
3. Click **Apply**.

Configuring a Custom Central Policy File for Automatic Installation

If you define your own Central policy file, you can configure the SG appliance to automatically install any subsequent updated version of the file. To use this capability, you must change the Central policy file's first line with each version update. With automatic installation, the SG appliance checks for a change to the first line of the file. In defining a custom Central policy file, add an item, such as a comment, to the first line of the Central policy file that changes with each update. The following is a sample first line, containing date information that is routinely updated with each version:

```
; Central policy file MonthDate, Year version
```

When you update and save the file in the original location, the SG appliance automatically loads the updated version.

Configuring E-mail Notification

You can specify whether the SG appliance sends e-mail when the Central policy file changes. The e-mail address used is the same as that used in diagnostic reporting: the event recipient for the custom heartbeat e-mail. For information about diagnostic reporting, see "Diagnostic Reporting (Heartbeats)" on page 424.

To configure e-mail notification:

1. Select **Configuration > Policy > Policy Files > Policy Files**.
2. Select **Send me email when central file changes**.
3. Click **Apply**.

Configuring the Update Interval

You can specify how frequently the SG appliance checks for a new version of the Central policy file. By default, the appliance checks for an updated Central policy file once every 24 hours (1440 minutes). You must use the CLI to configure the update interval. You cannot configure the update interval through the Management Console.

To configure the update interval:

At the `(config)` command prompt, enter the following command:

```
SGOS#(config) policy poll-interval minutes
```

Checking for an Updated Central Policy File

You can manually check whether the Central policy file has changed. You must use the CLI. You cannot check for updates through the Management Console.

To check for an updated central file:

At the `(config)` command prompt, enter the following command:

```
SGOS#(config) policy poll-now
```

The SG appliance displays a message indicating whether the Central file has changed.

Resetting the Policy Files

You can clear all the policy files automatically through the CLI.

To clear all policy files:

1. At the `(config)` command prompt, enter the following command:

   ```
   SGOS#(config) policy reset
   WARNING: This will clear local, central, forward and VPM policy. Are
   you sure you want to reset ALL policy files? (y or n)
   ```

   The SG appliance displays a warning that you are resetting all of your policy files.

2. Enter **`y`** to continue or **`n`** to cancel.

Note: This command does not change the default proxy policy settings.

Moving VPM Policy Files from One SG Appliance to Another

VPM policy files are specific to the SG appliance where they were created. But just as you can use the same Central, Local, and Forward policy files on multiple SG appliance, you can use VPM policies created on one appliance on other appliances.

For detailed information on moving VPM policy files, see "Installing Policies" on page 143.

Viewing Policy Files

You can view either the compiled policy or the source policy files. Use these procedures to view policies defined in a single policy file (for example, using VPM) or in multiple policy files (for example, using the Blue Coat Central policy file and VPM).

Viewing the Installed Policy

Use the Management Console or a browser to display installed Central, Local, or Forward policy files.

Note: You can view VPM policy files through the **Visual Policy Files** tab.

To view Installed policy:

1. Select **Configuration > Policy > Policy Files > Policy Files**.
2. In the **View File** drop-down list, select **Current Policy** to view the installed and running policy, as assembled from all policy source files. You can also select **Results of Policy Load** to view any warnings or errors resulting from the last attempt (successful or not) to install policy.
3. Click **View**. The SG appliance opens a separate browser window and displays the installed policy file.

To view the currently installed policy through a browser:

1. Enter a URL in one of the following formats:
   - If an HTTPS-Console is configured, use `https://`*`SG_ip_address`*`:`*`HTTPS-Console_port`*`/Policy/current` (the default port is 8082).
   - If an HTTP-Console is configured, use `http://`*`SG_ip_address`*`:`*`HTTP-Console_port`*`/Policy/current` (the default port is 8081).

   The SG appliance opens a separate browser window and displays the policy.
2. Review the policy, then close the browser.

Viewing Policy Source Files

You can display source (uncompiled) policy files on the SG appliance.

To view policy source files:

1. Select **Configuration > Policy > Policy Files > Policy Files**.
2. To view a policy source file, select the file you want to view (**Local**, **Forward**, or **Central**) from the **View File** drop-down list and click **View**.

   The SG appliance opens a separate browser window and displays the appropriate source policy file.

Viewing Policy Statistics

You can view policy statistics on all requests processed by the SG appliance. Use the Management Console or a browser. You cannot view policy statistics through the CLI.

To review policy statistics:

1. Select **Statistics > Advanced**.
2. Click the **Policy** link.
3. Click the **Show policy statistics** link.

   A separate browser window opens and displays the statistics.

4. Examine the statistics, then close the browser.

To review policy statistics through a browser:

1. Enter a URL in one of the following formats:
   - If an HTTPS-Console is configured, use `https://SG_ip_address:HTTPS-Console_port/Policy/statistics` (the default port is 8082).
   - If an HTTP-Console is configured, use `http://SG_ip_address:HTTP-Console_port/Policy/statistics` (the default port is 8081).

   The SG appliance opens a separate browser window and displays the statistics.
2. Examine the statistics, then close the browser.

Related CLI Syntax to Manage Policy Files

```
SGOS#(config) policy order v l c
SGOS#(config) policy proxy-default {allow | deny}
SGOS# policy trace {all | none}
SGOS#(config) inline policy file end-of-input-marker
SGOS#(config) policy subscribe
SGOS#(config) policy notify:
SGOS#(config) show policy
SGOS#(config) show configuration
-or-
SGOS#(config) show sources policy {central | local | forward | vpm-cpl | vpm-xml}
```

Chapter 3: The Visual Policy Manager

The Visual Policy Manager (VPM) is a graphical policy editor included with the SG appliance. The VPM allows you to define Web access and resource control policies without having an in-depth knowledge of Blue Coat Content Policy Language (CPL) and without the need to manually edit policy files.

This chapter serves as a VPM object reference, and assumes that you are familiar with basic concepts of SG appliance policy functionality as described in Chapter 2, Managing Policy Files.

While VPM creates only a subset of everything you can achieve by writing policies directly in CPL, it is sufficient for most purposes. If your needs require more advanced policies, consult *Volume 10: Blue Coat SG Appliance Content Policy Language Guide.*

This chapter contains the following sections:

Related topics:

- Chapter 2: "Managing Policy Files"
- *Volume 7: Managing Content*
- *Volume 10: Blue Coat SG Appliance Content Policy Language Guide*

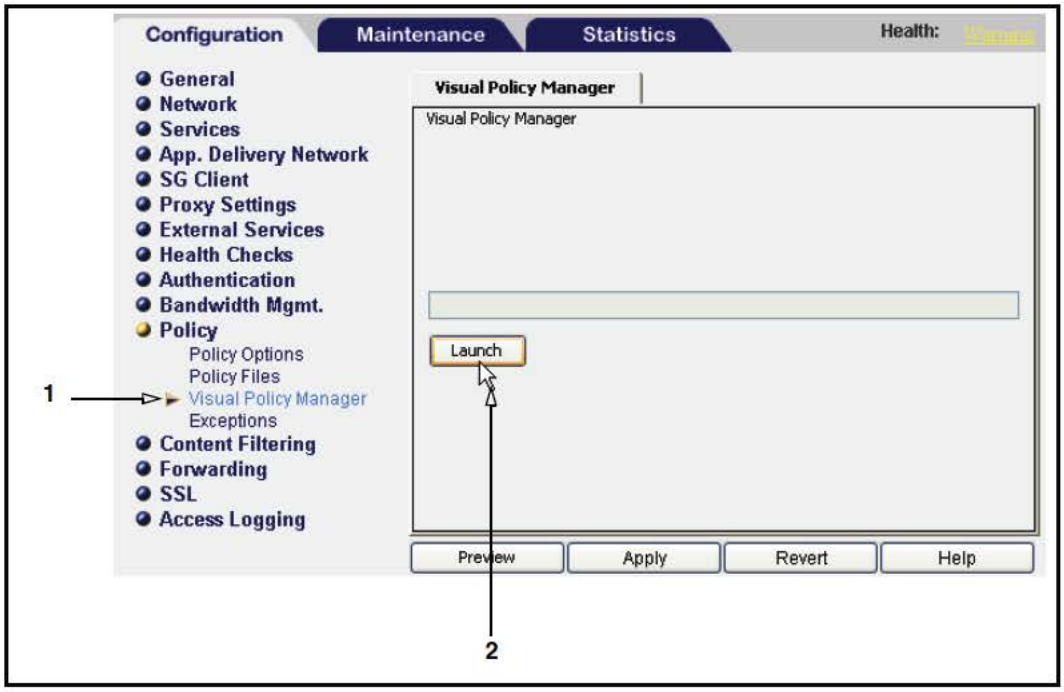
Section A: About the Visual Policy Manager

This section contains the following topics:

- "Launching the Visual Policy Manager" —Describes how to start VPM from the Management Console.
- "About the Visual Policy Manager User Interface" —Describes VPM menu items, tool bars, and work areas.
- "About VPM Components" —Provides definitions of the policy layers and describes how rule objects comprise the layers.
- "The Set Object Dialog" —Describes the dialog used to select objects to be added or edited.
- "The Add/Edit Object Dialog" —Describes the dialog used to add and edit rule objects.

Launching the Visual Policy Manager

To launch the VPM:



1. Select **Configuration > Policy > Visual Policy Manager**.
2. Click **Launch**.

   The VPM launches in a separate window.

About the Visual Policy Manager User Interface

The following figure labels VPM components.

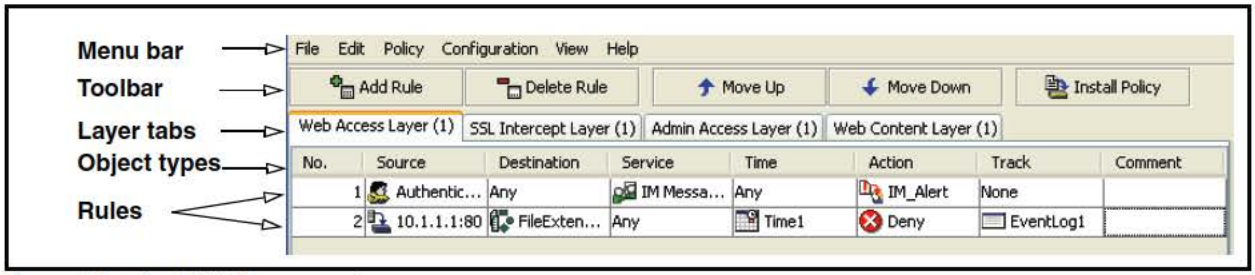


Figure 3-1. The VPM Components

Menu Bar

The following table describes VPM Menu Bar items.

Table 3-1. VPM Menu Bar Items

File	Install Policy On....	Saves all new policy rules.
	Revert to existing Policy on ...	Ignores changes and reloads installed policy rules.
	Exit	Exits the application.
Edit	Add Rule Delete Rule	Adds a new blank rule to the visible policy layer or removes a rule from the visible policy layer.
	Cut Rule Copy Rule Paste Rule	Standard cut, copy, and paste operations.
	Move Rule Up Move Rule Down	Moves rules up or down one position in a policy layer.
	Disable/Enable Layer	Disables or enables the selected layer. You can disable a layer without removing it from the VPM (thus losing composed policy rules) and re-enable it if required.
	Reorder Layers Delete Layer	Reorders the policy layers. Deletes a specific policy layer.
Policy	Add Admin Authentication Layer Add Admin Access Layer Add DNS Access Layer Add SOCKS Authentication Layer Add SSL Intercept Layer Add SSL Access Layer Add Web Authentication Layer Add Web Access Layer Add Web Content Layer Add Forwarding Layer	The Policy menu items add policy layers to be populated with policy rules.

Table 3-1. VPM Menu Bar Items (Continued)

Configuration	Set DNS Lookup Restrictions	Restricts DNS lookups during policy evaluation.
	Set Reverse DNS Lookup Restrictions	Restricts reverse DNS lookups during policy evaluation.
	Set Group Log Order	Configures the order in which the group information is logged.
	Edit Categories	Edits content filtering categories.
View	Generated CPL	Displays the CPL generated by VPM.
	Current SG Appliance VPM Policy Files	Displays the currently stored VPM policy files.
	Object Occurrences	Lists the user-created object(s) in the selected rule; lists use in other rules as well.
	All Objects	Displays a dialog that lists current static and user-defined VPM objects. You can also create, edit, and delete objects. See "Centralized Object Viewing and Managing" on page 131.
	Tool Tips	Toggles the tool-tip display on and off.
Help	Help Topics	Displays the online help.
	About	Displays copyright and version information.

Tool Bar

The VPM Tool Bar contains the following functions:

- **Add Rule**—Adds a blank rule to visible policy layer; all values for the rule are the defaults.
- **Delete Rule**—Deletes the selected rule from the visible policy layer.
- **Move Up**—Moves a rule up one position in the visible policy layer.
- **Move Down**—Moves a rule down one position in the visible policy layer.
- **Install Policy**—Converts the policies created in VPM into Blue Coat Content Policy Language (CPL) and installs them on the SG appliance.

Policy Layer Tabs

Every policy layer you create from the **Policy > Add Layer** menu is displayed as a tab. Click a tab and the rules included in that policy layer display below in the main body of the pane. Right-clicking a tab displays the options of disable or enabling, renaming, and deleting the policy layer.

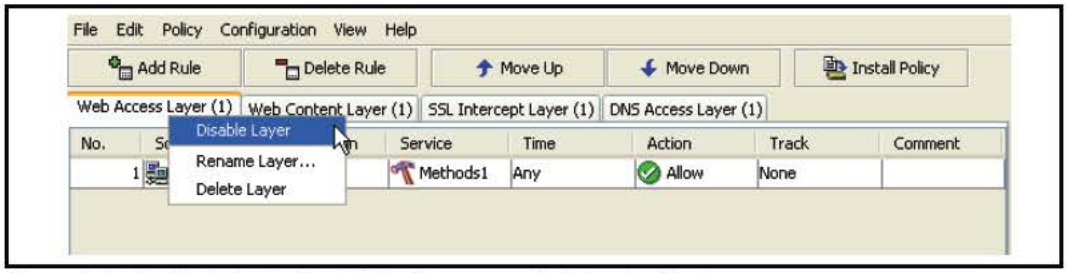


Figure 3-2. Right-click a Policy Tab to Rename or Delete a Policy Layer

Each VPM policy layer is described in later sections in this chapter.

Rules and Objects

A policy layer can contain multiple rules. Every rule is numbered and listed in a separate row. To create a new rule, click the **Add Rule** button; a new rule is added to the bottom of the list. If multiple rules exist within a policy layer, the SG appliance finds the first one that matches a given situation and ignores the remaining rules. Therefore, rule order is important. Use the **Move** buttons on the rule bar to reorder the rules in a policy.

Each rule is comprised of objects. The objects are the individual elements of a rule you specify. With the exception of **No.** (number), which indicates the order of the rule in the layer and is filled in automatically, all objects are configurable.

To specify or edit an object setting, position the mouse in the appropriate object cell within a rule and right-click to display the drop-down menu.

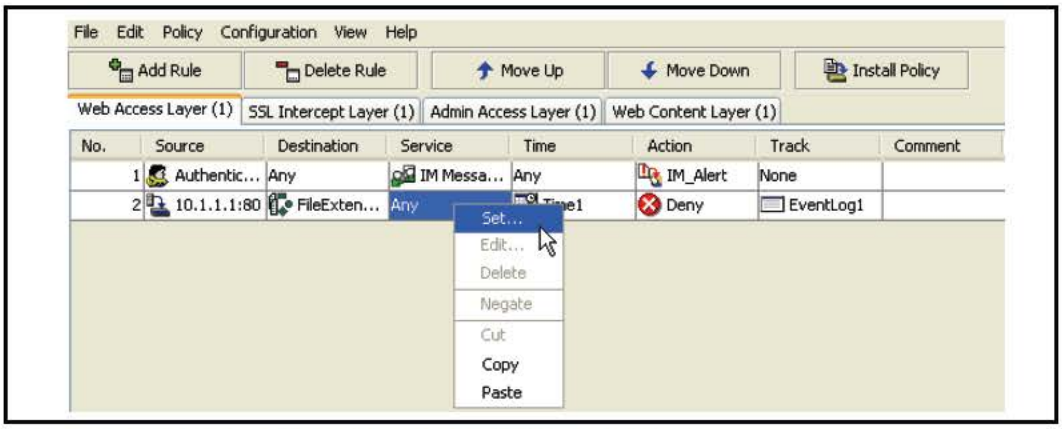


Figure 3-3. Right-click a Rule Cell to Set or Edit Object Properties

Each object type is described in "Policy Layer and Rule Object Reference" on page 38.

About Code Sharing With the Management Console

The VPM *shares* information in various lists from the current configuration in the Management Console, *not* the saved SG appliance configurations. When the VPM is launched, it inherits the state of the SG appliance from the Management Console and remains synchronous with that Management Console. This state might include configuration changes that have not yet been applied or reverted. This does not include any changes made through the CLI. When you click **Apply** in the Management Console, the configurations are sent to the SG appliance; the Management Console and the VPM become synchronous with the SG appliance.

For example, the SG appliance has two ICAP response services installed, A and B. In the Management Console, you remove service B, but do not click **Apply**. You then start the VPM and view the ICAP Response Services object. Only service A is viewable and selectable.

The VPM synchronizes the latest change from the Management Console when the following occur:

- Clicking **Revert**.
- Clicking **Apply**.
- Clicking **Policy Install**.
- Restart the Management Console.
- Log out and re-log into the Management Console.

Any information the Management Console acquires from installable lists is immediately available in the VPM. The following are the lists the VPM obtains from the Management Console:

- Access Log fields.
- Authentication character sets.
- Authentication realms.
- Bandwidth gain classes.
- Categories.
- Exceptions.
- Forwarding hosts.
- ICAP request and response services.
- Keyrings.
- SOCKS gateways.
- Websense filter services.

About VPM Components

This section describes the specific policy layer types and rule objects.

Policy Layers

The layers are:

- **Administration Authentication**—Determines how administrators accessing SG appliance must authenticate.
- **Administration Access**—Determines who can access the SG appliance to perform administration tasks.
- **DNS Access**—Determines how the SG appliance processes DNS requests.
- **SOCKS Authentication**—Determines the method of authentication for accessing the proxy through SOCKS.
- **SSL Intercept**—Determines whether to tunnel or intercept HTTPS traffic.

- **SSL Access**—Determines the allow/deny actions for HTTPS traffic.
- **Web Authentication**—Determines whether user clients that access the proxy or the Web must authenticate.
- **Web Access**—Determines what clients can and cannot access on the Web and specifies any restrictions that apply.
- **Web Content**—Determines caching behavior, such as verification and ICAP redirection.
- **Forwarding**—Determines forwarding hosts and methods.

As you create policy layers, you will create many different layers of the same type. Often, an overall policy requires layers of different types designed to work together to perform a task. For example, Authentication and Access layers usually accompany each other; an Authentication layer determines if a user or client must authenticate, and an Access layer subsequently determines where that user or client can go (what SG appliance or Web sites they can access) once they are authenticated.

Each object type is described in "Policy Layer and Rule Object Reference" on page 38.

Rule Objects

Policy layers contain rule objects. Only the objects available for that policy layer type are displayed. There are two types of objects:

- Static Objects—A self-contained object that cannot be edited or removed. For example, if you write a rule that prohibits users from accessing a specific Web site, the **Action** object you select is **Deny**.

  Static objects are part of the system and are always displayed.

- Configurable Objects—A configurable object requires parameters. For example, consider the rule mentioned in the previous item that prohibits users from accessing a specific Web site. In this case, the user is a **Source** object. That object can be a specific IP Address, user, group, user agent (such as a specific browser), and so on. Select one and then enter the required information (such as a verifiable user name or group name).

  Configurable objects do not exist until you create them. A created object is listed along with all static objects in the list dialog, and you can reuse it in other applicable policy layers. For example, an IP address can be a **Source** or **Destination** object in many different policy-layer types.

  Important: The orders of policy layers, and the order of rules within a layer are important. For more information, see "How Policy Layers, Rules, and Files Interact" on page 139.

While individual object-type menus occasionally contain entries specific to the object type, the basic menu options are:

- **Allow**—(Web Access Layer Action column only) Quick menu access; sets the policy to allow.
- **Deny**—(Web Access Layer Action column only) Quick menu access; sets the policy to deny.
- **Set**—Displays the Set Object dialog where you select an object or create a new one.

- **Edit**—Opens the Edit Object dialog where you edit an object or change to another.
- **Delete**—Removes the selected object from the current rule and restores the default.
- **Negate**—Defined as *not*. Negate provides flexibility in writing rules and designing the structure of policies. The following is a simple Web Access rule that states: "When any client tries to access a URL contained in an object of **JobSearch**, allow access."

Figure 3-4. A Simple Web Access Layer Policy Rule

Dragging the pointer to the **Destination** list, right-clicking to display the drop-down list, and clicking **Negate** invokes a red circle with a horizontal white line in the icon in the cell.

Figure 3-5. The Red Icon in the Cell Indicates Negation

Now the rule specifies allow all URLs except the ones contained in the **JobSearch** category object.

- **Cut**, **Copy**, and **Paste** are the standard paste operations with the following restrictions: you can only paste anything cut or copied from the same column in the same table and the copy and paste functions do not work across multiple layers.

The following table describes the general function of each object type:

Table 3-2. Object Type Functions

Object	Description
Source	Specifies the source attribute, such as an IP address, user, or group.
Destination	Specifies the destination attribute, such as a URL, IP address, and file extension.
Service	Specifies the service attribute, such as protocols, protocol methods, and IM file transfer limitations.
Time	Specifies day and time restrictions.
Action	Specifies what to do when the rule matches.
Track	Specifies tracking attributes, such as event log and E-mail triggers.
Comment	Optional. You can provide a comment regarding the rule.

Policy Layer/Object Matrix

The following table displays which object types are available in each policy layer.

Table 3-3. Available Object Types

Policy Layer	Source	Destination	Service	Time	Action	Track	Comment
Admin Authentication	x				x	x	x
Admin Access	x				x	x	x
DNS Access	x	x		x	x	x	x
SOCKS Authentication	x				x	x	x
SSL Intercept	x	x			x	x	x
SSL Access	x	x	x		x	x	x
Web Authentication	x	x			x	x	x
Web Access	x	x	x	x	x	x	x
Web Content		x	x		x	x	x
Forwarding	x	x	x		x	x	x

The Set Object Dialog

This section discusses the Set Object dialog used to select objects for configuration.

The object rules in all policy layer types determine the conditions for a particular policy rule. Depending on the type of policy layer, an object can be anything from a user or group to an IP address or a URL and so forth.

To create a rule, right-click a cell in an object cell. The relevant Set Object dialog displays. In this dialog, select the objects for the rule or create new objects as necessary.

Objects have type-specific icons to provide a visual aid in distinguishing among different types in the list.

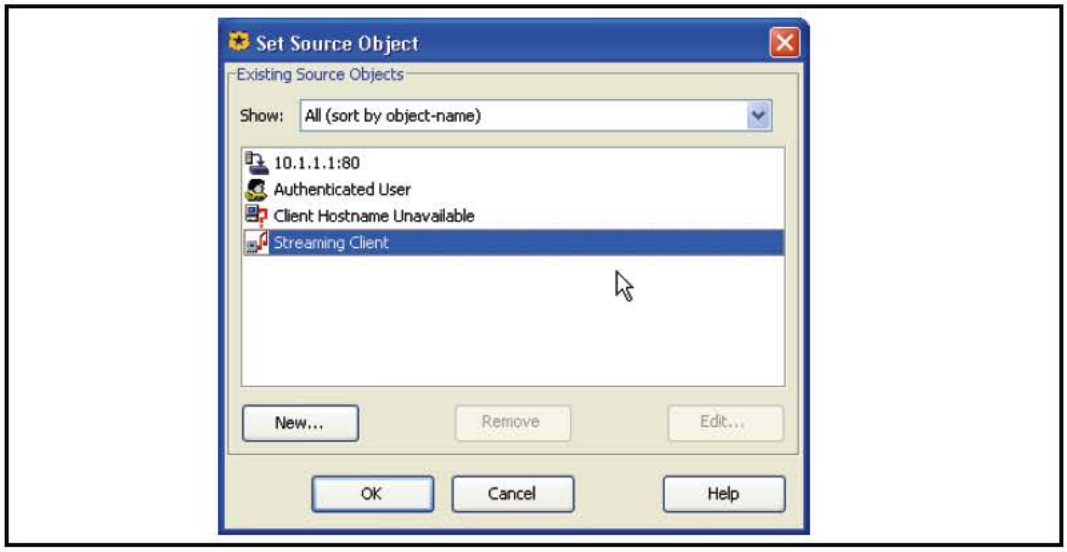


Figure 3-6. Set Source Object Dialog with Selectable Objects

The Set Object dialog only displays or allows you to create the objects allowable in the specific option of the rule type you are creating. But if more than one policy-layer type uses the same object type (for example, IP address can be a source in rules for four of the five types of policies), then those existing objects display in all Set Object dialogs, regardless of policy-layer type.

Controlling the List of Objects in the Set Object Window

As you create more policies, it is likely that the lists of existing objects in the various Set Object dialogs expand. You can restrict the display of objects in the list to a specific type by selecting an object type from the **Show** drop-down list above the objects field. The following figure demonstrates the window displayed above with the list restricted to Client IP addresses.

Figure 3-7. Limiting the Set Object Dialog view.

The Add/Edit Object Dialog

From the Set Object dialog, the Add Object dialog is used to define configurable objects. Existing configurable options can be altered using the Edit Object dialog. In terms of functionality, the two dialogs are identical.

For the initial configuration of an object, click **New** on the Set Object dialog to display the Add Object dialog. Perform the tasks required to configure the object and click **OK**. The newly named and configured object appears in the list of selectable objects in the Set Object dialog and is ready to be selected for the rule.

To edit an existing object, select an object from the list and click **Edit**. The Edit Object dialog appears with the existing parameters on display. Edit as necessary and click **OK**.

To remove an existing object, select an object from the list and click **Remove**. A secondary prompt verifies your attempt to remove the object; click **OK**. The object is deleted.

Online Help

The VPM contains its own Help module (a porting of this chapter). Each object in the VPM contains a Help button that links to the corresponding object reference in the Help file. This reference describes the purpose of the object. Interaction with other policy and references to feature-related sections in the *Blue Coat Configuration and Management Guide Suite* volumes are provided, if relevant. Also, this Help module contains navigation buttons and its own Table of Contents.

Note: The online Help file is displayed in a separate window and requires a few seconds to load and scroll to the correct object. The speed of your system might impact this slight lag time. Furthermore, this lag time increases on slower machines running JRE v1.5.

Section B: Policy Layer and Rule Object Reference

This section contains the following topics:

About the Reference Tables

The tables in this section list the static and configurable objects available for each policy layer.

Note: If viewing this document as a PDF, you can click an object name to jump to a description of that object (all objects are described in Section C). To jump back to a specific policy layer reference, click policy layer name in any object reference table that appears in the next section.

Administration Authentication Policy Layer Reference

The following table provides the objects available in the Administration Authentication policy layer.

Source Objects	Action Objects	Track Objects
Client IP Address/Subnet	Do Not Authenticate	Trace
Client Hostname	Deny	
Proxy IP Address/Port	Authenticate	
Combined Objects	Force Authenticate	

Administration Access Policy Layer Reference

The following table provides the objects available in the Administration Access policy layer.

Source Objects	Service Objects	Action Objects	Track Objects
Client IP Address/ Subnet	Service Name	Allow Read-Only Access	Event Log
Client Hostname		Allow Read-Write Access	Email
Proxy IP Address/Port		Deny	SNMP
User		Log Out/Do Not Log Out Other Users With Same IP	Trace
Group		Log Out/Do Not Log Out User	Combined Objects
Attribute		Log Out/Do Not Log Out User's Other Sessions	
User Login Address		Force Deny	
User Login Time		Set Authorization Refresh Time	
User Login Count		Set Credential Refresh Time	
Client Address Login Count		Set Surrogate Refresh Time	
Combined Objects		Combined Objects	

DNS Access Policy Layer Reference

The following table provides the objects available in the DNS Access policy layer.

Source Objects	Destination Objects	Time Objects	Action Objects	Track Objects
Client IP Address/ Subnet	DNS Response Contains No Data	Time	Bypass DNS Cache	Event Log
Proxy IP Address/ Port	DNS Response IP Address/Subnet	Combined Objects	Do Not Bypass DNS Cache	Email
DNS Request Name	RDNS Response Host		Allow DNS From Upstream Server	SNMP
RDNS Request IP Address/Subnet	DNS Response CNAME		Serve DNS Only From Cache	Trace
DNS Request Opcode	DNS Response Code		Enable/Disable DNS Imputing	Combined Objects

Source Objects	Destination Objects	Time Objects	Action Objects	Track Objects
DNS Request Class	Category		Send DNS/RDNS Response Code	
DNS Request Type	Server Connection DSCP Trigger		Send DNS Response	
DNS Client Transport	Combined Objects		Send Reverse DNS Response	
Client Connection DSCP Trigger			Reflect IP	
Combined Objects			Manage Bandwidth	
			Set Client Connection DSCP Value	
			Set Server Connection DSCP Value	
			Combined Objects	

SOCKS Authentication Policy Layer Reference

The following table provides the objects available in the SOCKS Authentication policy layer.

Source Objects	Action Objects	Track Objects
Client IP Address/Subnet	Do Not Authenticate	Trace
Client Hostname	Authenticate	
Proxy IP Address/Port	Force Authenticate	
SOCKS Version		
Combined Objects		

SSL Intercept Layer Reference

The following table provides the objects available in the SSL Forward Proxy policy layer.

Source Objects	Destination Objects	Action Objects	Track Objects
Client Hostname Unavailable	Destination IP Address/ Subnet	Enable HTTPS Intercept	Event Log
Client Hostname	Destination Host/ Port	Enable HTTPS Intercept on Exception	Email
Proxy IP Address/Port	Request URL	Combined Objects	SNMP

Source Objects	Destination Objects	Action Objects	Track Objects
Combined Objects	Request URL Category		Trace
	Server URL		Combined Objects
	Server Certificate		
	Server Certificate Category		
	Combined Objects		

SSL Access Layer Reference

The following table provides the objects available in the SSL Access Layer policy layer.

Source Objects	Destination Objects	Service Objects	Action Objects	Track Objects
Authenticated User	Destination IP Address/Subnet	Request Forwarded	Allow	Event Log
Client Hostname Unavailable	Destination Host/ Port	Client Protocol	Deny (static)	Email
Guest User	Request URL	SSL Proxy Mode	Require/Do Not Require Client Certificate	SNMP
Client IP Address/ Subnet	Request URL Category	Health Check	Force Deny	Trace
Client Hostname	Server URL	Combined Objects	Force Deny (Content Filter)	Combined Objects
Proxy IP Address/ Port	Server Certificate		Deny	
User	Server Certificate Category		Return Exception	
Group	Server Certificate		Set Client Certificate Validation	
Attribute	Server Certificate Category		Set Server Certificate Validation	
User Login Address	Server Negotiated Cipher		Combined Objects	
User Authentication Error	Server Negotiated Cipher Strength			
User Authorization Error	Server Negotiated SSL Version			
Client Certificate	Combined Objects			

Source Objects	Destination Objects	Service Objects	Action Objects	Track Objects
Client Negotiated Cipher				
Client Negotiated Cipher Strength				
Client Negotiated SSL Version				
Combined Objects				

Web Authentication Policy Layer Reference

The following table provides the objects available in the Web Authentication policy layer.

Source Objects	Destination Objects	Action Objects	Track Objects
Client Hostname Unavailable	Destination IP Address/ Subnet	Do Not Authenticate	Trace
Client IP Address/ Subnet	Destination Host/ Port	Deny	
Client Hostname	Request URL	Authenticate	
Proxy IP Address/Port	Request URL Category	Authenticate Guest	
User Agent	Combined Objects	Add Default Group	
Request Header		Force Authenticate	
Combined Objects		Authentication Charset	
		Set IP Address For Authentication	
		Permit Authentication Error	
		Permit Authorization Error	
		Combined Objects	

Section B: Policy Layer and Rule Object Reference

Web Access Policy Layer Reference

The following table provides the objects available in the Web Access policy layer.

Web Access policy layers regulate, from a general to a granular level, who or what can access specific Web locations or content.

- Users, groups, individual IP addresses, and subnets, as well as object lists comprised of any combination of these, can be subject to rules.
- Rules can include access control for specific Web sites, specific content from any Web site, individual IP addresses, and subnets.
- Actions taken can range from allowing and denying access to more finely tuned changes or limitations.
- Rules can also be subject to day and time specifications and protocol, file type, and agent delimiters.

Source Objects	Destination Objects	Service Objects	Time Objects	Action Objects	Track Objects
Streaming Client	Destination IP Address/Subnet	Using HTTP Transparent Authentication	Time	Allow	Event Log
Client Hostname Unavailable	Destination Host/ Port	Virus Detected	Combined Objects	Deny	Email
Guest User	Request URL	Client Protocol		Force Deny	
Authenticated User	Request URL Category	Service Name		Bypass Cache	SNMP
Client IP Address/ Subnet	File Extensions	Protocol Methods		Do Not Bypass Cache	
Client Hostname	HTTP MIME Types	IM File Transfer		Check/Do Not Check Authorization	Trace
Proxy IP Address/ Port	Apparent Data Type	IM Message Text		Always Verify	Combined Objects
User	Response Code	IM Message Reflection		Use Default Verification	
Group	Response Header	Streaming Content Type		Block/Do Not Block PopUp Ads	
Attribute	Response Data	ICAP Error Code		Force/Do Not Force IWA for Server Auth	
User Login Address	IM Buddy	Health Status		Log Out/Do Not Log Out Other Users With Same IP	

Section B: Policy Layer and Rule Object Reference

Source Objects	Destination Objects	Service Objects	Time Objects	Action Objects	Track Objects
User Login Time	IM Chat Room	Combined Objects		Log Out/Do Not Log Out User	
User Login Count	Server Connection DSCP Trigger			Log Out/Do Not Log Out User's Other Sessions	
Client Address Login Count	Combined Objects			Reflect/Do Not Reflect IM Messages	
User Authentication Error				Block/Do Not Block IM Encryption	
User Authorization Error				Support/Do Not Support Persistent Client Requests	
User Agent				Support/Do Not Support Persistent Server Requests	
IM User Agent				Trust/Do Not Trust Destination IP	
Request Header				Deny	
SOCKS Version				Return Exception	
IM User				Return Redirect	
P2P Client				Send IM Alert	
Client Negotiated Cipher				Modify Access Logging	
Client Negotiated Cipher Strength				Override Access Log Field	
Client Connection DSCP Trigger				Rewrite Host	
Combined Objects				Reflect IP	
P2P Client				Suppress Header	

Section B: Policy Layer and Rule Object Reference

Source Objects	Destination Objects	Service Objects	Time Objects	Action Objects	Track Objects
Client Negotiated Cipher				Control Request Header/Control Response Header	
Client Negotiated Cipher Strength				Notify User	
Client Connection DSCP Trigger				Strip Active Content	
Combined Objects				Set Client HTTP Compression	
				Set Server HTTP Compression	
				Manage Bandwidth	
				Modify IM Message	
				Return ICAP Feedback	
				Set External Filter Service	
				Set ICAP Request Service	
				Set FTP Connection	
				Set SOCKS Acceleration	
				Disable SSL Detection	
				Set Streaming Max Bitrate	
				Set Client Connection DSCP Value	
				Set Server Connection DSCP Value	
				Set ADN Connection DSCP	

Source Objects	Destination Objects	Service Objects	Time Objects	Action Objects	Track Objects
				Set Authorization Refresh Time	
				Set Credential Refresh Time	
				Set Surrogate Refresh Time	
				Combined Objects	

Web Content Policy Layer Reference

The following table provides the objects available in the Web Content policy layer.

The Web Content policy layer applies to requests independent of user identity.

Content scanning policy layers scan requested URLs and file types for viruses and other malicious code. You must have an ICAP service installed on the SG appliance to use this policy type.

Destination Objects	Action Objects	Track Objects
Destination IP Address/Subnet	Check/Do Not Check Authorization	Event Log
Destination Host/Port	Always Verify	
Request URL	Use Default Verification	Email
Request URL Category	Use Default Caching	SNMP
File Extensions	Do Not Cache	Trace
HTTP MIME Types	Force Cache	Combined Objects
Response Header	Mark/Do Not Mark As Advertisement	
Response Data	Support/Do Not Support Persistent Server Requests	
Server Connection DSCP Trigger	Enable/Disable Pipelining	
Combined Objects	Set Dynamic Categorization	
	Set External Filter Service	
	Set Client HTTP Compression	
	Set Server HTTP Compression	
	Manage Bandwidth	
	Set ICAP Request Service	
	Set ICAP Response Service	

Destination Objects	Action Objects	Track Objects
	Set TTL	
	Modify Access Logging	
	Override Access Log Field	
	Set Server Connection DSCP Value	
	Combined Objects	

Forwarding Policy Layer Reference

The following table provides the objects available in the Forwarding policy layer.

Source Objects	Destination Objects	Service Objects	Action Objects	Track Objects
Streaming Client	Destination IP Address/Subnet	Client Protocol	Send Direct	Trace
Authenticated User	Destination Host/ Port	Health Check	Integrate/Do Not Integrate New Hosts	
Guest User	Server URL	Health Status	Connect Using ADN When Possible/Do Not Connect Using ADN	
Client IP Address/ Subnet	Server Connection DSCP Trigger	Combined Objects	Allow Content From Origin Server	
Client Hostname	Combined Objects		Serve Content Only From Cache	
Proxy IP Address/ Port			Select SOCKS Gateway	
User			Select Forwarding	
Group			Reflect IP	
Attribute			Manage Bandwidth	
User Login Address			ADN Server Optimization	
User Login Time			Set IM Transport	
User Login Count			Set Streaming Transport	
Client Address Login Count			Set ADN Connection DSCP	
User Authentication Error			Set Server Connection DSCP Value	

Section B: Policy Layer and Rule Object Reference

Source Objects	Destination Objects	Service Objects	Action Objects	Track Objects
User Authorization Error			Combined Objects	
SOCKS Version				
P2P Client				
Client Connection DSCP Trigger				
Combined Objects				

Section C: Detailed Object Column Reference

This section contains the following topics:

Source Column Object Reference

A *source* object specifies the communication or Web transaction origin that is evaluated by the policy. Not all policy layers contain the same source objects.

Important: Because of character limitations required by the generated CPL, only alphanumeric, underscore, dash, ampersand, period, or forward slash characters can be used to define a source object name.

Any

Applies to any source.

Streaming Client

This is a static object. This rule applies to any request from a streaming client.

Client Hostname Unavailable

This is a static object. This rule applies if the client IP address could not be looked up with a reverse DNS query.

Authenticated User

This is a static object. This rule applies to any authenticated user.

Guest User

This is a static object. This rule applies to all guest users.

Client IP Address/Subnet

Specifies the IP address and, optionally, a subnet mask of a client. The policy defined in this rule applies only to this address or addresses on this subnet. This object is automatically named using the prefix **Client**; for example, **Client: 1.2.0.0/255.255.0.0**.

Note: See "Combined Source Object" on page 64 for related information regarding this source object.

Client Hostname

Specifies a reverse DNS hostname resolved in the reverse lookup of a client IP address. Enter the host name and select matching criteria. This object is automatically named using the prefix **Client**; for example, **Client: host.com**. If you select a matching qualifier, that attribute is appended to the object in parentheses. For example, **Client: host.com (RegEx)**.

Proxy IP Address/Port

Specifies the IP address and, optionally, a port on the SG appliance. The policy defined in this rule applies only to this address or addresses with this subnet.

User

Specifies an individual user in the form of a verifiable username or login name. Enter a user name and an authentication realm. The dialog then displays different information depending on the type of authentication realm specified. Select the appropriate realm from the drop-down list. Items in the list are taken from the realms configured by the administrator in the SG appliance.

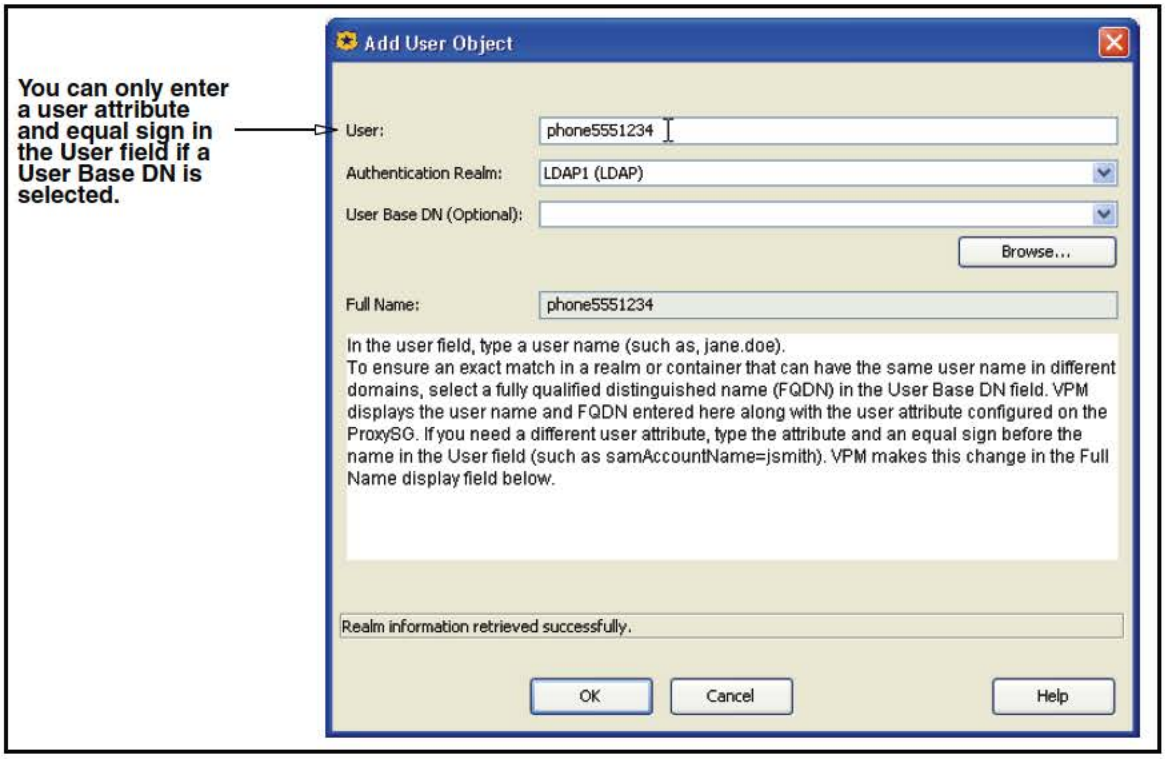
LDAP

You can optionally select a User Base DN from a drop-down list. Entries in the **User Base DN** list come from those specified by the administrator in the SG appliance. You can also edit an entry selected in the list, type a new one, or click **Browse** to manually select a name. Edited names and new names are retained in the list. Notice in the **Full Name** field that the VPM takes the User Attribute type specified by the administrator in the SG appliance (**cn=** in the following illustration), and associates it with the user name and Base DN entered here.

Important: When you configure a realm, the SG appliance assumes a default primary user attribute (**sAMAccountName** for Active Directory; **uid** for Netscape/iPlanet Directory Server/SunOne; **cn** for Novell NDS). You can accept the default or change it. Whatever is entered there is what the VPM uses here, entering it in the **Full Name** display field once a Base DN is selected.

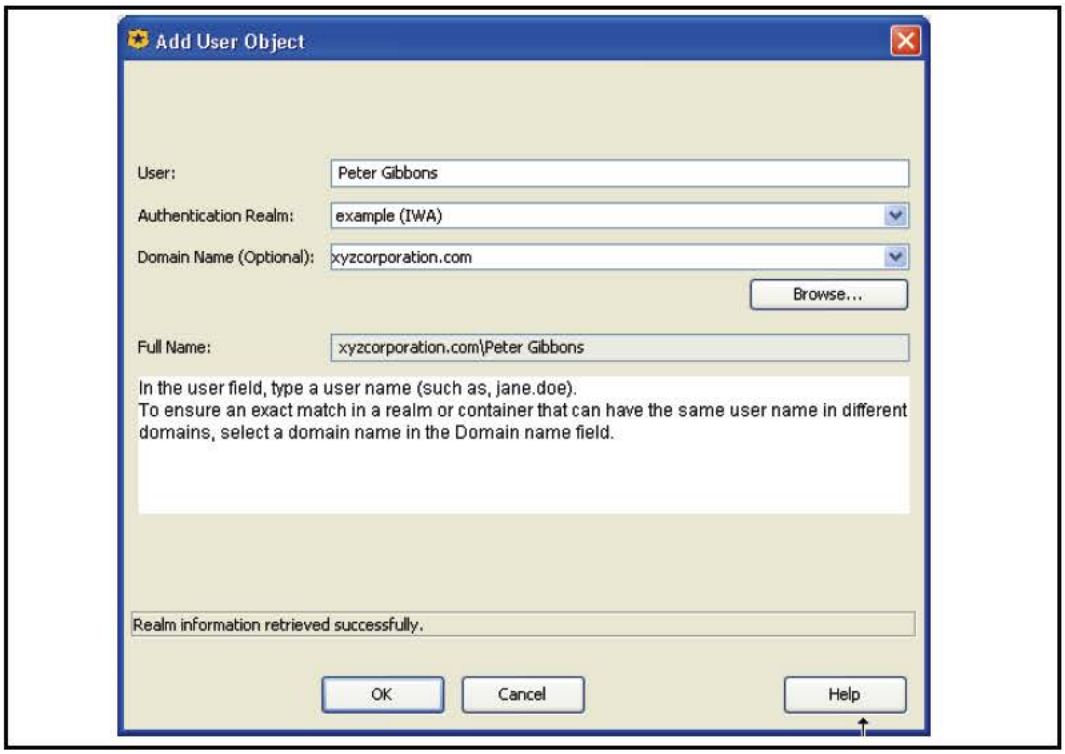
If the primary user attribute specified in the SG appliance differs from the primary user attribute specified in the directory server, enter the latter in the **User** field with the appropriate value (in the format **attribute=value**). This replaces the entry in the **Full Name** field. Examine the following screenshot. Assume that the organization uses *phone* as the primary attribute in its LDAP directory:

Section C: Detailed Object Column Reference



IWA

Entries in this list are not prepopulated. You must enter a name in the **Domain Name** field. An entered name is retained and can subsequently be selected and edited. Notice in the **Full Name** field that VPM displays domain name and user name entered above.



RADIUS

Entries in this list are not prepopulated. You must enter a name in the **User** field. An entered name is retained and can subsequently be selected and edited. Notice in the **Full Name** field that VPM displays domain name and user name entered above.

Windows SSO

Entries in this list are not prepopulated. You must enter a name in the **User** field. Entries in the **Domain Name** list come from those specified by the administrator in the SG appliance. You can also edit an entry selected in the list, type a new one, or click **Browse** to manually select a name.

Local

Entries in this list are not prepopulated. You must enter a name in the **User** field. An entered name is retained and can subsequently be selected and edited. Notice in the **Full Name** field that VPM displays domain name and user name entered above.

Certificate

If a Certificate realm is selected and that realm uses an LDAP realm as authentication realm, the Browse button is clickable. This option allows you to browse the LDAP database and select users, thus preventing typing errors possible from manually entering names in the fields. If the Certificate realm does not use an LDAP authentication realm, **Browse** is not displayed.

Netegrity SiteMinder

Entries in this list are not prepopulated. You must enter a name in the **User** field. An entered name is retained and can subsequently be selected and edited. Notice in the **Full Name** field that VPM displays domain name and user name entered above.

Oracle COREid

Entries in this list are not prepopulated. You must enter a name in the **User** field. An entered name is retained and can subsequently be selected and edited. Notice in the **Full Name** field that VPM displays domain name and user name entered above.

Policy Substitution

Entries in this list are not prepopulated. You must enter a name in the **User** field. An entered name is retained and can subsequently be selected and edited. Notice in the **Full Name** field that VPM displays domain name and user name entered above.

Sequences

Entries in this list are not prepopulated. You must enter a name in the **User** field. An entered name is retained and can subsequently be selected and edited. Notice in the **Full Name** field that VPM displays domain name and user name entered above. From the **Member Realm** drop-down list, select an authentication realm (already configured on the SG appliance). Depending on the realm type, new fields appear.

Group

Specifies a verifiable group name. Enter a user group and an authentication realm. The dialog then displays different information depending on the type of authentication realm specified.

- **Group** field—Replace the default with a verifiable group name.
- **Authentication Realm** field—Select the appropriate realm from the drop-down list. Items in the list are taken from the realms configured by the administrator in the SG appliance.
  - **LDAP**—Entries in the **Group Base DN** list come from those specified by the administrator in the SG appliance. You can also edit an entry selected in the list, or type a new one. Edited names and new names are retained in the list. Notice in the **Full Name** field that the VPM takes the User Attribute type specified by the administrator in the SG appliance (**cn=** in the following illustration), and conjoins it with the group name and Base DN entered here.

Important: When you create a group, the default attribute is **cn=** in the **Full Name** display field.

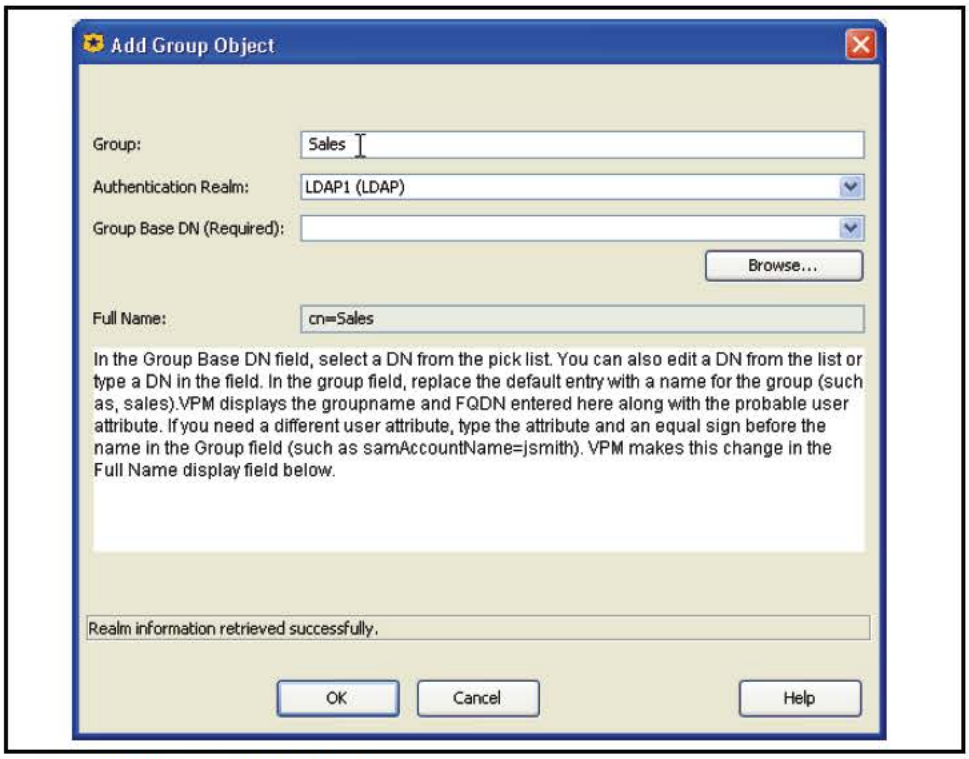


Figure 3-8. Adding a group object

If the primary user attribute specified in the SG appliance differs from the primary user attribute specified in the directory server, you need to enter the latter here. Do that by typing it in the **Group** field with the appropriate value (in the format **attribute=value**). Doing so replaces the entry in the **Full Name** field. Unlike the comparable situation when creating a user (described immediately above), when creating a group, the **Group Base DN** does not need to be selected in order to type the **attribute=value** pair in the **Group** field.

- **IWA**—Entries in this list are not prepopulated. You must enter a name in the **Domain Name** field. A name typed in is retained and can subsequently be selected and edited. Notice in the **Full Name** field that the VPM displays the domain name and group name entered above.
- **RADIUS**—Entries in this list are not prepopulated. You must enter a name in the **Group** field.
- **Windows SSO**—Entries in this list are not prepopulated. You must enter a name in the **Group** field.

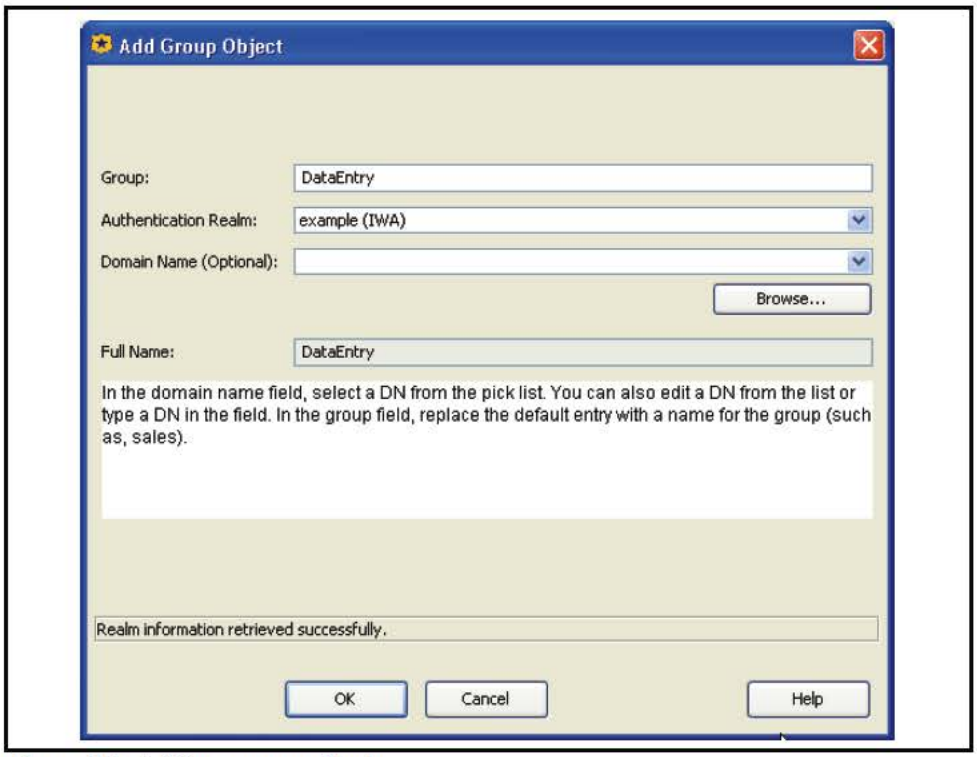


Figure 3-9. Adding a group object

- **Local**—Entries in this list are not prepopulated. You must enter a name in the **Group** field. A name typed in is retained and can subsequently be selected and edited. Notice in the **Full Name** field that VPM displays the group name entered above.
- **Certificate**—If a Certificate realm is selected and that realm uses an LDAP realm as authentication realm, the **Browse** button is clickable. This option allows you to browse the LDAP database and select users, thus preventing typing errors possible from manually entering names in the fields. If the **Certificate** realm does not use an LDAP authentication realm, **Browse** is not displayed.
  - **Netegrity SiteMinder**—Entries in this list are not prepopulated. You must enter a name in the **Group** field. A name typed in is retained and can subsequently be selected and edited. Notice in the **Full Name** field that VPM displays the group name entered above.
  - **Oracle COREid**—Entries in this list are not prepopulated. You must enter a name in the **Group** field. A name typed in is retained and can subsequently be selected and edited. Notice in the **Full Name** field that VPM displays the group name entered above.
  - **Policy Substitution**—Entries in this list are not prepopulated. You must enter a name in the **Group** field. A name typed in is retained and can subsequently be selected and edited. Notice in the **Full Name** field that VPM displays the group name entered above.

- **Sequences**—Entries in this list are not prepopulated. You must enter a name in the **Group** field. An entered name is retained and can subsequently be selected and edited. Notice in the **Full Name** field that VPM displays domain name and user name entered above. From the **Member Realm** drop-down list, select an authentication realm (already configured on the SG appliance). Depending on the realm type, new fields appear.

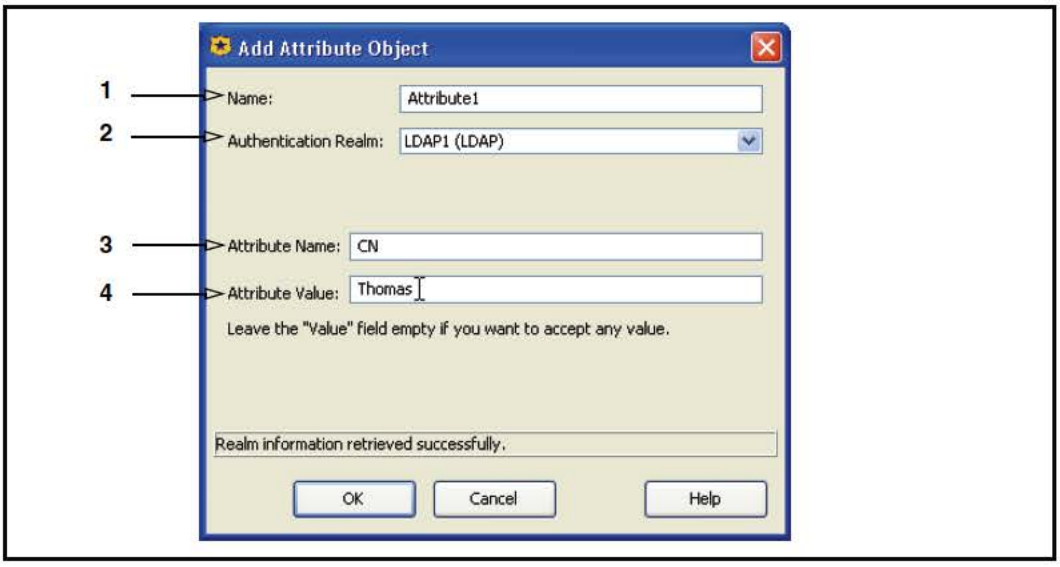
Attribute

Specifies an LDAP or Radius realm-specific attributes.

LDAP

Specifies a specific LDAP attribute (and optional value).

Specify an LDAP attribute:



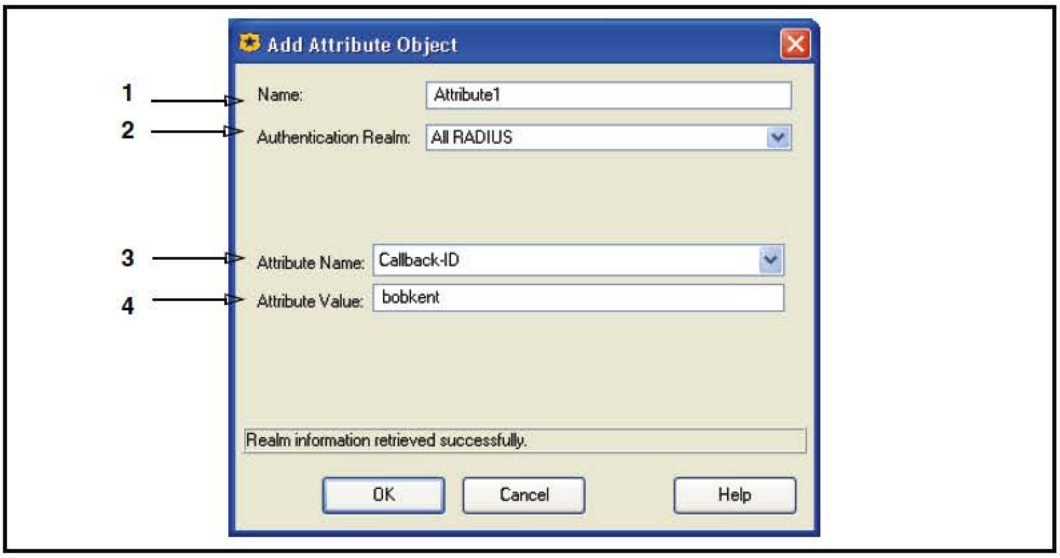
1. In the **Name** field, enter a name for the object or leave as is to accept the default.
2. From the **Authentication Realm** drop-down list, select **All LDAP** or a specific realm.
3. In the **Attribute Name** field, enter a valid attribute.
4. In the **Attribute Value** field, enter value for the specified LDAP attribute, or leave blank to accept any value.

   The above example sets a Common Name (**CN**) attribute with the value of **Kent** to the **LDAP1** realm.

RADIUS

Specifies a RADIUS attribute.

To specify a RADIUS attribute:



1. In the **Name** field, enter a name for the object or leave as is to accept the default.
2. Select **All RADIUS** or a specific realm.
3. Select an **Attribute Name.**
4. Enter an **Attribute Value** for the **Attribute Name.**

User Login Address

The condition matches the IP address used to login. Serves as a request parameter for Windows Single Sign-On (SSO).

User Login Time

This condition matches the number of seconds since the current login started, and can limit the length of a login session.

User Login Count

This condition matches the number of times that a specific user is logged in with the current realm. This condition ensures that a user is only logged in at one workstation. If the condition is combined with the `user.login.log_out_other` property, old logins on other workstations are automatically logged out.

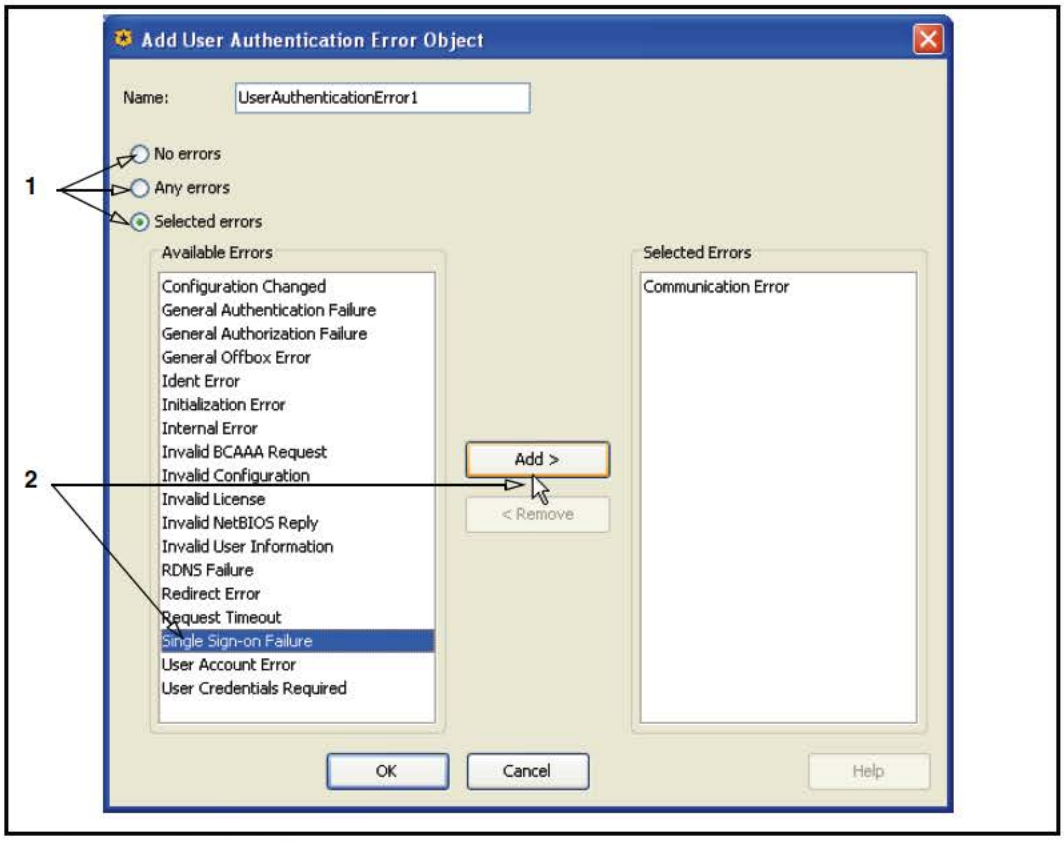
Client Address Login Count

This condition matches and can limit the number of different users who are logged into the current IP address.

User Authentication Error

Checks for a matches of specified user authentication errors.

To specify a User Authentication Error object:



1. Select one of the following:
   - **None**: Authentication was attempted and no user errors occurred.
   - **Any**: Authentication was attempted a user error occurred.
   - **Selected errors**: Authentication was attempted and one of the selected errors occurred.
2. If you selected **Selected errors**:
   a. Select one or more error types (use Control + Left-click to highlight multiple errors).
   b. Click **Add** to move the errors to the **Selected** field.
   c. Name the object or accept the default name.
3. Click **OK**.

Note: If authentication fails and no default groups are added through policy (see Guest Authentication and Default Groups), the group conditions always evaluate to false. Verify group conditions if you permit authentication errors, especially in scenarios where users are denied based on group membership.

User Authorization Error

Checks for a match of specified user authorization errors.

To specify a User Authorization Error object:

1. Select one of the following:
   - **None**: Authorization was attempted and no user errors occurred.
   - **Any**: Authorization was attempted a user error occurred.
   - **Selected errors**: Authorization was attempted and one of the selected errors occurred.
2. If you selected **Selected errors**:
   a. Select one or more error types (use Control + Left-click to highlight multiple errors).
   b. Click **Add** to move the errors to the **Selected** field.
   c. Name the object or accept the default name.
3. Click **OK**.

Note: If authorization fails and no default groups are added through policy (see Guest Authentication and Default Groups), the group conditions always evaluate to false. Verify group conditions if you permit authorization errors, especially in scenarios where users are denied based on group membership.

DNS Request Name

Specifies a DNS request. Enter the host name and select matching criteria. This object is automatically named using the prefix **DNS**; for example, **DNS: host.com**. If you select a matching qualifier, that attribute is appended to the object in parentheses. For example, **DNS: host.com (RegEx)**.

RDNS Request IP Address/Subnet

Specifies the reverse DNS IP address and, optionally, a subnet mask. The policy defined in this rule applies only to this address or addresses on this subnet. This object is automatically named using the prefix **RDNS**; for example, **RDNS: 5.6.0.0/255.255.0.0**.

DNS Request Opcode

Specifies OPCODEs to represent in the DNS header.

To specify a DNS Request OPCODE object:

1. In the **Name** field, enter a custom name or leave as is to accept the default.
2. Select one or more of the OPCODEs.
3. Click **OK**.

DNS Request Class

Specifies the DNS request class (QCLASS) properties.

To specify a DNS request class object:

1. In the **Name** field, enter a custom name or leave as is to accept the default.

2. Select one or more of the request classes.
3. Click **OK**.

DNS Request Type

Specifies the DNS request types (QTYPE) attributes.

To specify a DNS Request Type object:

1. In the **Name** field, enter a custom name or leave as is to accept the default.
2. Select one or more of the request types.
3. Click **OK**.

DNS Client Transport

Specifies the DNS client transport method, UDP or TCP.

To specify a DNS Client Transport object:

1. Select **UDP Transport** or **TCP Transport**. This object is automatically named using the prefix **DNS**; for example, **DNS: Client Transport UDP**.
2. Click **OK**.

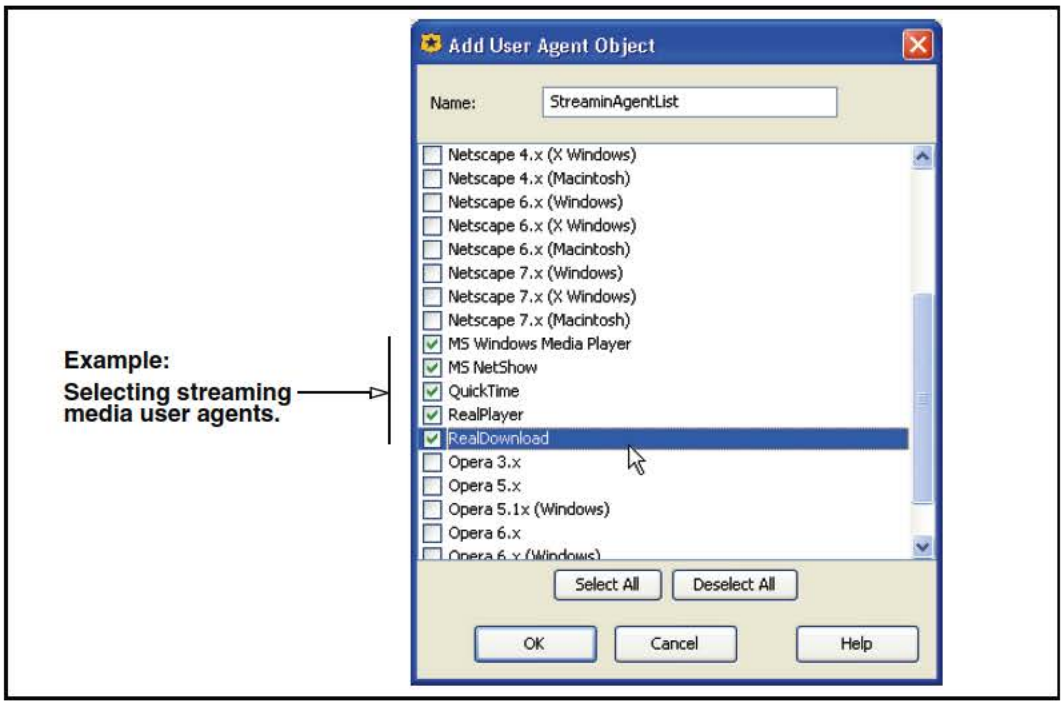
SOCKS Version

Specifies the SOCKS version, 4 or 5. This object is automatically named as **SOCKSVersion4** or **SOCKSVersion5**.

User Agent

Specifies one or more agents a client might use to request content. The choices include specific versions of: Microsoft Internet Explorer, Netscape Communicator, Microsoft Windows Media Player and NetShow, Real Media RealPlayer and RealDownload, Apple QuickTime, Opera, and Wget.

The policy defined in this rule applies to these selected agents. You can name this list and create other custom lists to use with other policy layer rules.



Note: If you require a user agent not contained in this list, use the **Request Header** object, which can contain user agent specified as a header.

IM User Agent

Checks the specified string for a match in the user agent provided by the IM client. For example, specify the string **Lotus** to distinguish between the Lotus AOL client and the standard AOL client.

To specify a User Agent:

1. In the **IM User Agent** field, enter a string.
2. From the drop-down list, select a matching criteria.
3. Click **Add**.

Request Header

Specifies the rule applies to requests containing a specific header. Blue Coat supplies a list of standard headers, but you can also select a custom header.

To specify a request header:

1. In the **Name** field, enter a custom name or leave as is to accept the default.
2. From the **Show** drop-list select the viewing field from **All** to **Standard** or **Custom**, as desired. **Standard** displays only the default standard headers. **Custom** displays any admin-defined headers that exist.
3. From the **Header Name** drop-list, select a standard or custom header or enter a new custom header name.
4. In the **Header Regex** field, enter the header values to which this rule applies.

Client Certificate

Allows for testing common name and subject fields in client certificates.

IM User

Specifies an IM user by their handle. IM traffic sent to or from this user is subject to this rule. You can enter a complete user ID, a string that is part of a user ID, or a string with a regular expression. Select the match type from the drop-down list to the right (**Exact**, **Contains**, or **RegEx**).

Example: Designating Bob Kent in Purchasing as IM user number 1.

P2P Client

Specifies peer-to-peer (P2P) clients.

To specify P2P clients:

1. In the **Name** field, enter a name for the object or accept the default.
2. Select **All P2P Clients** (all protocols become selected), or one or more P2P protocols.

3. Click **OK**.

Client Negotiated Cipher

Allows the testing of the SSL cipher in use between the SG appliance and the browser. Select a code from the drop-down list.

To specify a client negotiated cipher:

1. In the **Name** field, enter a name for the object or accept the default.
2. Select one or more cipher codes valid for this rule**.**
3. Click **OK**.

Client Negotiated Cipher Strength

Tests the cipher strength between a SG appliance-to-browser (client) HTTPS connection.

To specify a client negotiated cipher strength:

1. In the **Name** field, enter a name for the object or accept the default.
2. Select one or more of the strength options valid for this rule **Export**, **High**, **Medium**, or **Low.**
3. Click **OK**.

Low, **Medium**, and **High** strength ciphers are *not* exportable.

Client Negotiated SSL Version

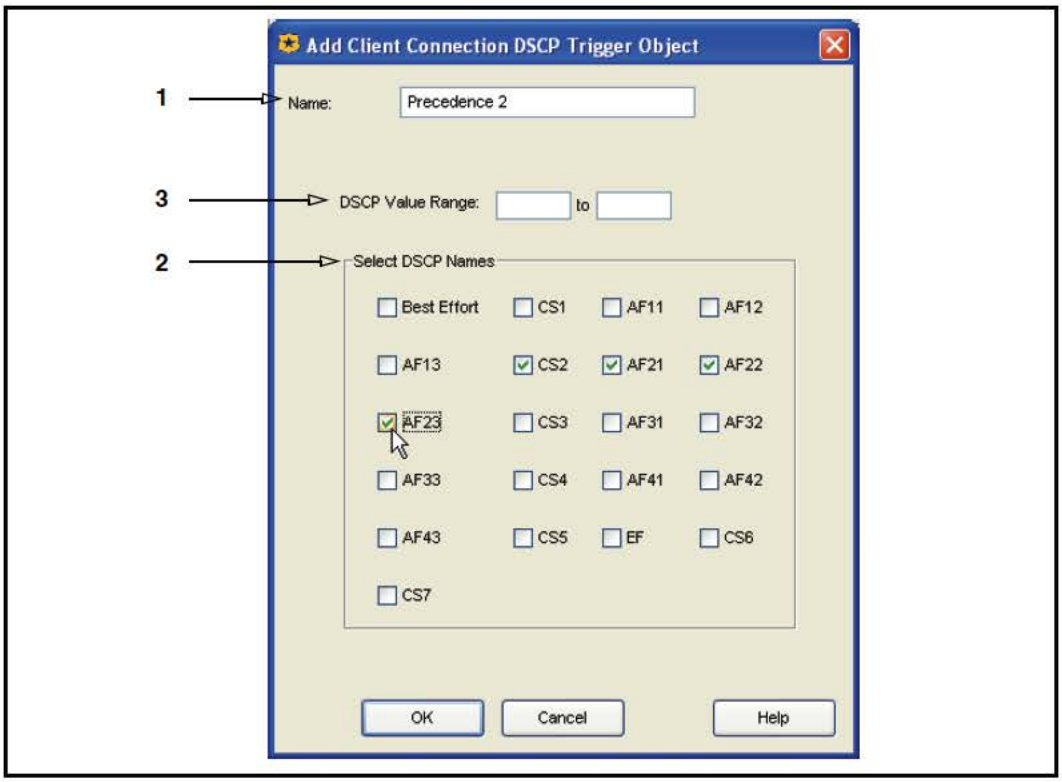
Tests the SSL version between a SG appliance-to-browser (client) HTTPS connection.

To specify a client negotiated SSL version:

1. In the **Name** field, enter a name for the object or accept the default.
2. Select one or more of the version options valid for this rule **SSL 2.0**, **SSL 3.0**, or **TLS 1.0**.
3. Click **OK**.

Client Connection DSCP Trigger

Tests the inbound differentiated service code point (DSCP) value of primary client-to-SG appliance connections. After testing DSCP bits (in the IP header), additional policy dictates how to handle traffic associated with the *type of service*.



To specify DSCP values to test against inbound client connections:

1. In the **Name** field, enter a name for the object or accept the default. This example creates an object that tests for an IP Precedence of 2 or any Assured Forwarding Class (AFC) of type 2 (for low, medium, and high drop rates).
2. Select IP Precedence values (denoted by **CS**) and Assured Forwarding Classes (Denoted by **AF**) as required).
3. (Optional) Rather than select Precedence and AFC values, enter a DSCP value range. The valid range is **0** to **63**. Blue Coat does not recommend this option. Most applications fit into one of the defined values.

For conceptual information about configuring the SG appliance to manipulate traffic based on type of service, refer to "Managing QoS and Differential Services" on page 194.

Combined Source Object

Allows you to create an object that combines different source types. See "Using Combined Objects" on page 128.

Note: Blue Coat strongly recommends that combined objects with large lists of Client IP Address/Subnet values (see "Client IP Address/Subnet" on page 50) do not contain other source objects. If other source objects are present, the policy evaluation might experience a significant performance degradation.

Source Column/Policy Layer Matrix

The following matrix lists all of the **Source** column objects and indicates which policy layer they apply to.

Object	Admin Auth	Admin Acc	DNS Acc	SOCKS Auth	SSL Int	SSL Acc	Web Auth	Web Acc	Web Cont	Fwding
Streaming Client								x		
Client Hostname Unavailable					x	x	x	x		
Authenticated User						x		x		x
Guest User						x		x		
Client IP Address/Subnet	x	x	x	x	x	x	x	x		x
Client Hostname	x			x	x	x	x	x		x
Proxy IP Address/Port	x	x	x	x	x	x	x	x		x
User		x				x		x		x
Group		x				x		x		x
Attribute		x				x		x		x
User Login Address		x				x		x		x
User Authentication Error						x		x		x
User Authorization Error						x		x		x
User Login Time		x				x		x		x
User Login Count		x				x		x		x
Client Address Login Count		x				x		x		x
DNS Request Name			x							
RDNS Request IP Address/Subnet			x							
DNS Request Opcode			x							
DNS Request Class			x							
DNS Request Type			x							
DNS Client Transport			x							
SOCKS Version				x				x		x
User Agent							x	x		
IM User Agent								x		
Request Header							x	x		
Client Certificate						x				
IM User								x		

Object	Admin Auth	Admin Acc	DNS Acc	SOCKS Auth	SSL Int	SSL Acc	Web Auth	Web Acc	Web Cont	Fwding
P2P Client								x		x
Client Negotiated Cipher						x		x		
Client Negotiated Cipher Strength						x		x		
Client Negotiated SSL Version						x				
Client Connection DSCP Trigger			x					x		x
Combined Objects	x	x	x	x	x	x	x	x		x

Destination Column Object Reference

A *destination* object specifies the communication or Web traffic destination that is evaluated by the policy. Not all policy layers contain the same destination objects.

Important: Because of character limitations required by the generated CPL, only alphanumeric, underscore, dash, ampersand, period, or forward slash characters can be used to define a destination object name.

Any

Applies to any destination.

DNS Response Contains No Data

This is a static object.

Destination IP Address/Subnet

Specifies the IP address and, optionally, a subnet mask of a destination server. The policy defined in this rule only applies to this address only or addresses within this subnet. This object is automatically named using the prefix **Destination**; for example, **Destination: 1.2.0.0/255.255.0.0**.

Destination Host/Port

Specifies the hostname or port of a destination server. The policy defined in this rule applies to this host on this port only. Enter the host name and port number, and select matching criteria. This object is automatically named using the prefix **Destination**; for example, **Destination: company.com:80**.

Request URL

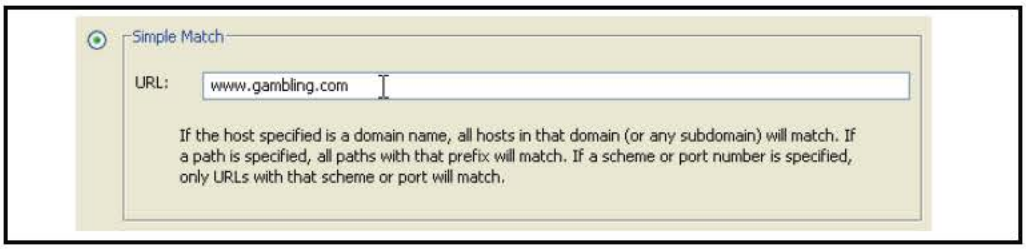
Applies to a URL request sent by the client to the SG appliance.

To check for a match against requested URL

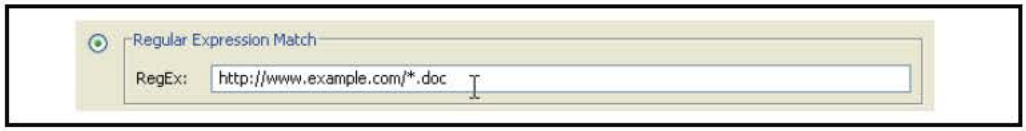
Select an option and enter the required information in the fields:

Section C: Detailed Object Column Reference

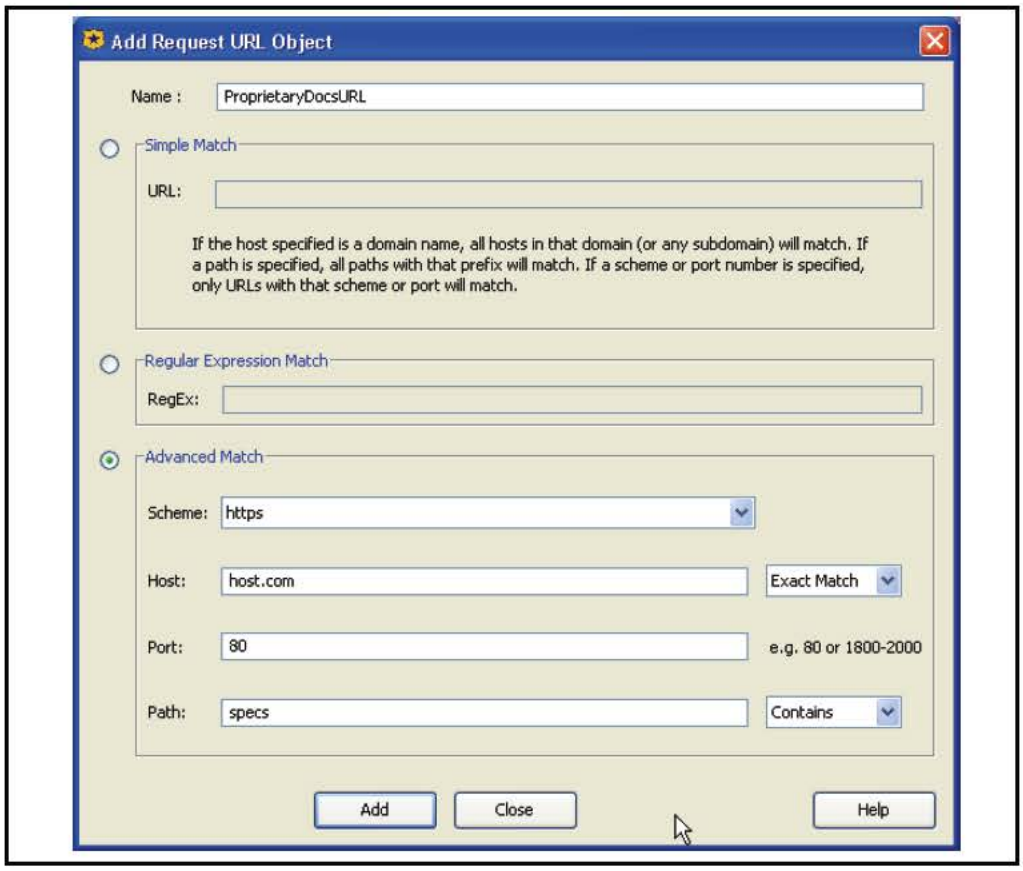
- **Simple Match**—Matches a partial URL. If a host name is specified, all hosts in that domain or subdomain match; if a path is specified, all paths with that path prefix match; if a scheme or port number is specified, only URLs with that scheme or port match. This object is automatically named using the prefix **URL**; therefore, the object is displayed in the rule as **URL: *host*.com**.



- **Regular Expression Match**—Specifies a regular expression. This object is automatically named using the prefix **URL**; therefore, the object is displayed as **URL: host.com (RegEx)**.



- **Advanced Match**—Specifies a scheme (protocol), host, port range, and/or path. Unlike the other options on this dialog, selecting **Advanced Match** allows you to enter a name at the top of the dialog to name the object. With host and path, you can select from the drop-down list to match exactly as entered or parts thereof: **Exact Match**, **Contains**, **At Beginning**, **At End**, or **RegEx**. If you select a matching qualifier, that attribute is appended to the object in parentheses. For example, **URL: host.com (Contains)**.



Request URL Category

Allows you to create and customize categories of URLs. Requested URLs are checked for matches and categorized and evaluated for further action dependent upon content filtering policy.

- **Policy**—Displays all current pre-defined and user created URL categories. This includes all category-related configurations created in the VPM, as well as in the Local and Central policy files (once installed). Select and deselect categories as required.

  You can also create new categories from this dialog, which is similar to the dialog accessed through the VPM Menu Bar as described in "Creating Categories" on page 134.

  If you enable a service, such a content filter, those relevant categories appear in this object.

- Blue Coat—If you are employing Blue Coat Web Filter, those categories appear here.

- **System**—Displays hard-coded SG appliance configurations. These are not editable, but you can select or deselect them.

Create a policy category:

1. Select **Policy**; click **Add**. The Object Name dialog appears.
2. Name the category and click **OK**.

3. Drop the **Policy** list and select the created category; click **Edit URLs**. The Edit Locally Defined Category Object dialog appears.
4. Enter URLs appropriate for the content filter category you are creating; click **OK**.
5. Click **OK**.

Note: If one or more other administrators have access to the SG appliance through other workstations and are creating categories either through VPM or with `inline` commands, consider that newly-created or edited categories are not synchronized until the policy is installed. When the policy is installed with VPM, the categories are refreshed. If confusion occurs, select the **File>Revert to Existing Policy on** SG **Appliance** option to restore the policy to the previous state and reconfigure categories.

Category Hierarchy Behavior

Once categories have been created, they can be selected and deselected as required. If you create sub-categories (a parent and child category hierarchy), the category selection behavior is the following:

- Selecting a parent category automatically selects all child categories if no child categories are already selected.

- Deselecting a parent category automatically deselects all child categories if all child categories are already selected.
- If one or more of the child categories are already selected or deselected, selecting or deselecting the parent category does *not* affect child categories—the status of selected or deselected remains the same.

This behavior applies to as many levels as you create.

Category

Functions the same as "Request URL Category" on page 68, but this object is unique to the DNS Access Layer.

Server URL

This object is functions the same as the "Request URL" on page 66 object, but applies to a URL sent from the SG appliance to a server. If the SG appliance is performing URL rewrites, the URL sent from the client might change, which requires another URL matching check.

Server Certificate

Allows testing of server certificate attributes to be used by the SG appliance-to-server HTTPS connections. Select one of the options:

- **Hostname**: This is the hostname you want to match in the server certificate. After you enter the hostname, select from the dropdown list one of the following: **Exact Match**, **Contains**, **At Beginning**, **At End**, **Domain**, or **Regex**.
- **Subject**: This is the fully qualified subject name in the server certificate. After you enter the subject, select from the dropdown list one of the following: **Exact Match**, **Contains**, **At Beginning**, **At End**, **Domain**, or **Regex**.

Server Certificate Category

Functions the same as the "Request URL Category" on page 68 object, but the piece of information used for matching and categorizing is the hostname in the server certificate.

Server Negotiated Cipher

Tests the cipher suites used in a SG appliance-to-server connection.

To specify a server-negotiated cipher:

1. In the **Name** field, enter a name for the object or accept the default.
2. Select one or more cipher codes valid for this rule**.**
3. Click **OK**.

Server Negotiated Cipher Strength

Specifies the cipher strength between a SG appliance-to-server HTTPS connection.

To specify a server-negotiated cipher strength:

1. In the **Name** field, enter a name for the object or accept the default.
2. Select one or more of the strength options valid for this rule **Export**, **High**, **Medium**, or **Low.**
3. Click **OK**.

Low, **Medium**, and **High** strength ciphers are *not* exportable.

Server Negotiated SSL Version

Specifies the SSL version between a SG appliance-to-server HTTPS connection.

To specify a server-negotiated SSL version:

1. In the **Name** field, enter a name for the object or accept the default.

2. Select one or more of the strength options valid for this rule **SSL 2.0**, **SSL 3.0**, or **TLS 1.0.**
3. Click **OK**.

File Extensions

Creates a list of file extensions. The rule is triggered for content with an extension matching any on the list. You can create multiple lists that contain various extensions to use in different rules. For example, create a list called **Images**, and select file extension types such as **GIF**, **JPEG**, **BMP**, **XPM**, and so on.

HTTP MIME Types

Creates a list of HTTP MIME content types. The rule is triggered for content matching any on the list. You can create multiple lists that contain various MIME types to use in different rules. For example, create a list called **MicrosoftApps**, and select MIME types **application/vnd.ms-excel**, **application/vnd.ms-powerpoint**, **application/vnd.ms-project**, and **application/vnd.works**.

Note: If you require a MIME type not contained in this list, use a Request URL object that uses the **At End** matching criteria.

Apparent Data Type

The options in this object identify data content associated with Microsoft DOS and Windows executable files. When used in a deny policy, the purpose of this object to deny executable downloads and block *drive-by* installation of spyware.

To specify apparent data type:

1. In the **Name** field, enter a name for the object or accept the default.
2. Select one or both of the following data types:
   - **DOS/Windows Executable**: Any type of Windows executable file, such as `.exe` files (the most common type of Microsoft executable file, which is self-extracting); `.dll` files (also self-extracting, but require another executable file), and `.ocx` files (ActiveX control files that can be installed if the browser security level is set to low). Windows PE, LE, and NE executable types are recognized.
   - **Microsoft Cabinet File**: Although not executable themselves, `.cab` (cabinet) files are used by spyware programs to propagate ActiveX controls. Code in HTML pages reference `.cab` files, which, from the inside, instruct the browser to download and extract ActiveX components.
3. Click **OK**.

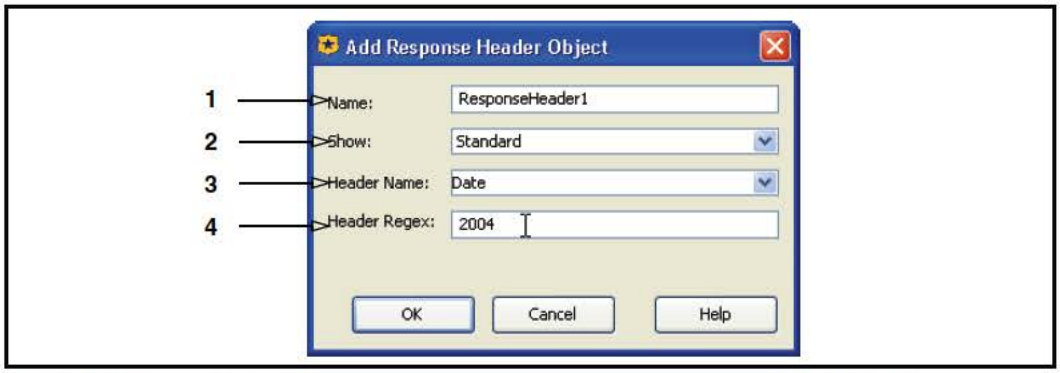
Response Code

Specifies the rule applies to content responses containing a specific HTTP code. Select a code from the drop-down list. You can name the rule object or accept the default name.

Response Header

Specifies the rule applies to content responses containing a specific header. Blue Coat supplies a list of standard headers, but you can also enter a custom header.

To specify a response header:

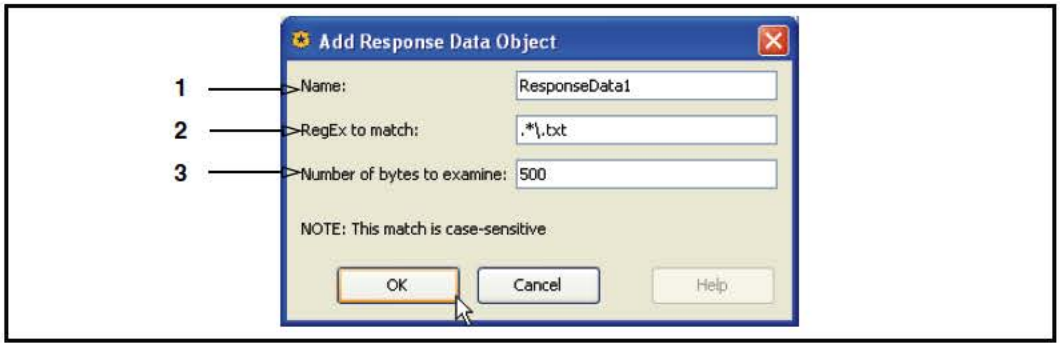


1. In the **Name** field, enter a custom name or leave as is to accept the default.
2. From the **Show** drop-down list select the viewing field from **All** to **Standard** or **Custom**, as desired. **Standard** displays only the default standard headers. **Custom** displays any admin-defined headers that exist.
3. From the **Header Name** drop-down list, select a standard or custom header.
4. In the **Header Regex** field, enter the header string this rule applies to.

Response Data

Specifies the rule applies to content responses containing specific regular expressions.

To specify a regular expression header:



1. In the **Name** field, enter a custom name or leave as is to accept the default.
2. In the **RegEx to match** field, enter the regular expression string to match.
3. In the **Number of bytes to examine** field, enter how many object bytes are scanned for the match.
4. Click **OK**.

IM Buddy

Specifies an IM buddy by their handle. IM traffic sent to or from this buddy is subject to this rule. You can enter a complete buddy ID, a string that is part of a buddy ID, or a string with a regular expression. Select the match type from the drop-down list to the right (**Exact**, **Contains**, or **RegEx**).

IM Chat Room

Specifies an IM chat room by name or other condition. IM traffic sent to this chat room is subject to this rule.

To create a chat room condition:

1. In the **Name** field, enter a name for the object or leave as is to accept the default.
2. Select one or more of the following conditions:
   a. **Room ID**—Specifies a specific IM chat room by its name. Enter a name and from the drop-down list select an option: **Exact Match**, **Contains**, or **RegEx**.
   b. **Type**—Specifies whether the room is **Private** or **Public**.
   c. **Invite Only**—Specifies to trigger if user must be invited or not.
   d. **Voice Enabled**—Specifies whether room supports voice chat or not.
   e. **Conference**—Specifies whether room has conference capability or not.
3. Click **OK**.

DNS Response IP Address/Subnet

Specifies the destination DNS IP address and, optionally, a subnet mask. The policy defined in this rule only applies to DNS responses containing this address or addresses within this subnet. This object is automatically named using the prefix **DNS**; for example, **DNS: 1.2.3.4/255.255.0.0**.

RDNS Response Host

Specifies a reverse DNS response hostname resolved in the reverse lookup of a client IP address. Enter the host name and select matching criteria. This object is automatically named using the prefix **RDNS**; for example, **RDNS: host.com**. If you select a matching qualifier, that attribute is appended to the object in parentheses. For example, **RDNS: host.com (RegEx)**.

DNS Response CNAME

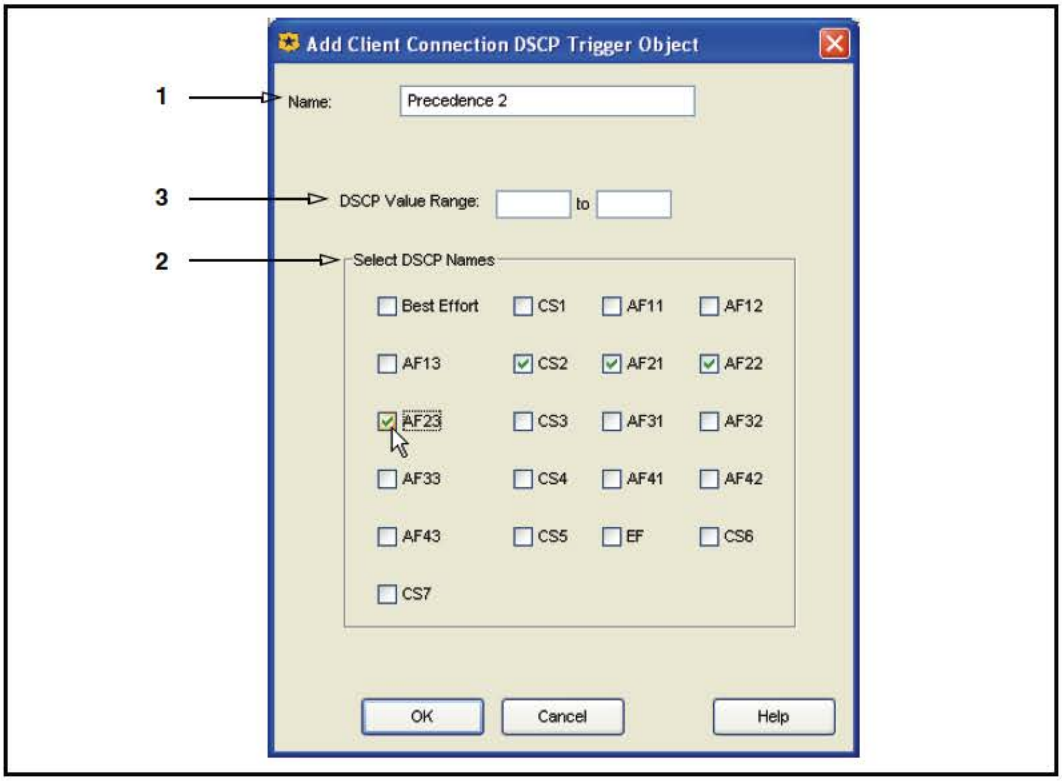
Specifies the rule applies to DNS CNAME responses matching a given hostname. Enter the host name and select matching criteria. This object is automatically named using the prefix **DNS CNAME**; therefore, the object is displayed as **DNS CNAME: host.com**.

DNS Response Code

Specifies the rule applies to DNS responses containing a specific DNS Response code. Select one or more codes from the list. You can name the rule object or accept the default name.

Server Connection DSCP Trigger

Tests the inbound differentiated service code point (DCSP) value of primary server-to-SG appliance connections. By testing DCSP bits (in the IP header), additional policy dictates how to handle traffic associated with the *type of service*.



To specify DSCP values to test against inbound server connections:

1. In the **Name** field, enter a name for the object or accept the default. This example creates an object that tests for an IP Precedence of 2 or any Assured Forwarding Class (AFC) of type 2 (for low, medium, and high drop rates).
2. Select IP Precedence values (denoted by **CS**) and Assured Forwarding Classes (Denoted by **AF**) as required).
3. (Optional) Rather than select Precedence and AFC values, enter a DSCP value range. The valid range is **0** to **63**. Blue Coat does not recommend this option. Most applications fit into one of the defined values.

For conceptual information about configuring the SG appliance to manipulate traffic based on type of service, refer to "Managing QoS and Differential Services" on page 194.

Combined Destination Objects

Allows you to create an object that combines different destination types. Refer to "Using Combined Objects" on page 128.

Destination Column/Policy Layer Matrix

The following matrix lists all of the **Destination** column objects and indicates which policy layer they apply to.

Object	Admin Auth	Admin Acc	DNS Acc	SOCKS Auth	SSL Int	SSL Acc	Web Auth	Web Acc	Web Cont	Fwding
Destination IP Address/Subnet					x	x	x	x	x	x
Destination Port					x	x	x	x	x	x
Request URL					x	x	x	x	x	x
Request URL Category					x	x	x	x	x	
Category			x							
Server URL					x	x				
Server Certificate					x	x				
Server Certificate Category					x	x				
Server Negotiated Cipher						x				
Server Negotiated Cipher Strength						x				
Server Negotiated SSL Version						x				
File Extensions								x	x	
HTTP MIME Types								x	x	
Apparent Data Type						x				
Response Header								x		
Response Code								x		

Object	Admin Auth	Admin Acc	DNS Acc	SOCKS Auth	SSL Int	SSL Acc	Web Auth	Web Acc	Web Cont	Fwding
Response Data								x		
IM Buddy								x		
IM Chat Room								x		
DNS Response IP Address/ Subnet			x							
RDNS Response Host			x							
DNS Response CNAME			x							
DNS Response Code			x							
Server Connection DSCP Trigger			x					x	x	x
Combined Objects			x				x	x	x	x

Service Column Object Reference

A *service* object specifies a service type, such as a protocol, that is evaluated by the policy. Not all policy layers contain the same service objects.

Important: Because of character limitations required by the generated CPL, only alphanumeric, underscore, dash, ampersand, period, or forward slash characters can be used to define a service object name.

Any

Applies to any service.

Using HTTP Transparent Authentication

This is a static object. The rule applies if the service is using HTTP transparent authentication.

Virus Detected

This is a static object. The rule applies if ICAP scanning detects a virus.

Request Forwarded

This is a static object. Automatically created when upgrading from SGOS 4.2.x to SGOS 5.2.x. Refer to the *Blue Coat SGOS 5.2 Upgrade/Downgrade Guide*.

Client Protocol

Specifies the client protocol types and subsets. From the first drop-down list, select a type from the drop-down list: **CIFS, Endpoint Mapper, FTP**, **HTTP**, **HTTPS**, **Instant Messaging**, **P2P**, **Shell**, **SOCKS**, **SSL**, **Streaming**, or **TCP Tunneling**.

The second drop-down list allows you to select a protocol subset (these options vary depending on the selected protocol):

- **All**—Applies to all communication using this type of protocol.
- **Pure**—Applies if the protocol is using a direct connection.
- **Over**—Applies if a protocol is communicating through a specific transport method.

Service Name

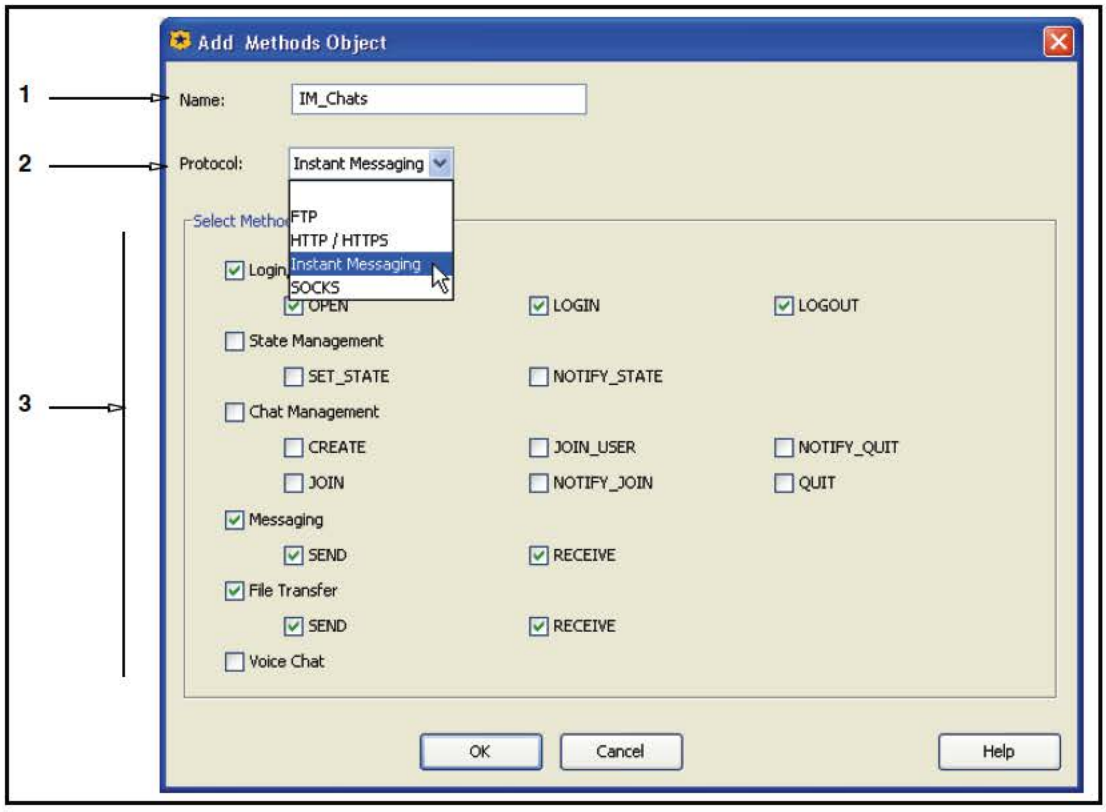
Specify any default or custom proxy service that exists on the SG appliance (created from the Management Console: **Configuration > Services > Proxy Services**).

- The **Web Access Layer** only displays and accepts proxy services.
- The **Admin Access Layer** only displays and accepts console services.

Protocol Methods

Specifies the protocol methods that trigger a rule.

To specify a protocol method:



1. In the **Name** field, enter a name or accept the default.
2. From the **Protocol** drop-down list, select one of the options: **FTP, HTTP, HTTPS, Instant Messaging, SOCKS**.
3. Select connection methods. These options vary depending on the selected protocol. The above example demonstrates basic Instant Messaging connections.

4. Click **OK**.

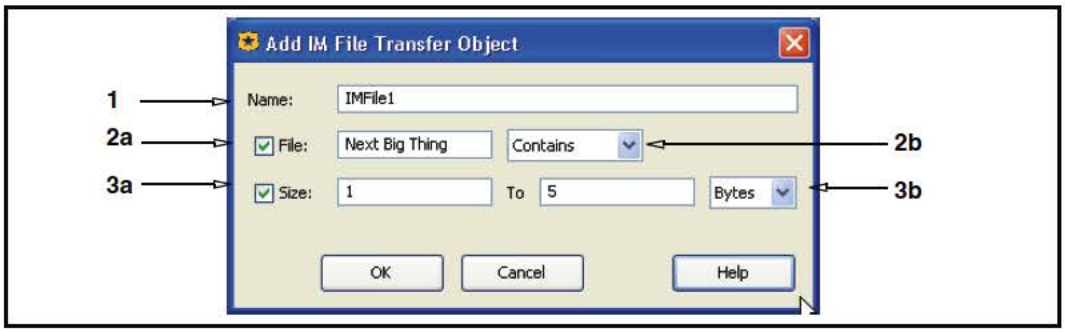
SSL Proxy Mode

Specifies the deployment mode of the SSL proxy: **HTTPS Forward Proxy requests, HTTPS Reverse Proxy requests, Unintercepted SSL requests**. This objects allows you to apply policy to a subset of SSL traffic going through the SG appliance. For example, this object can be used to enforce strong cipher suites for HTTPS reverse proxy requests while, allowing all ciphers suites for HTTPS forward proxy requests.

IM File Transfer

Specifies the rule is applied to IM file transfers, which can be triggered by matching the file name, file size, or both.

To specify IM file transfer parameters:

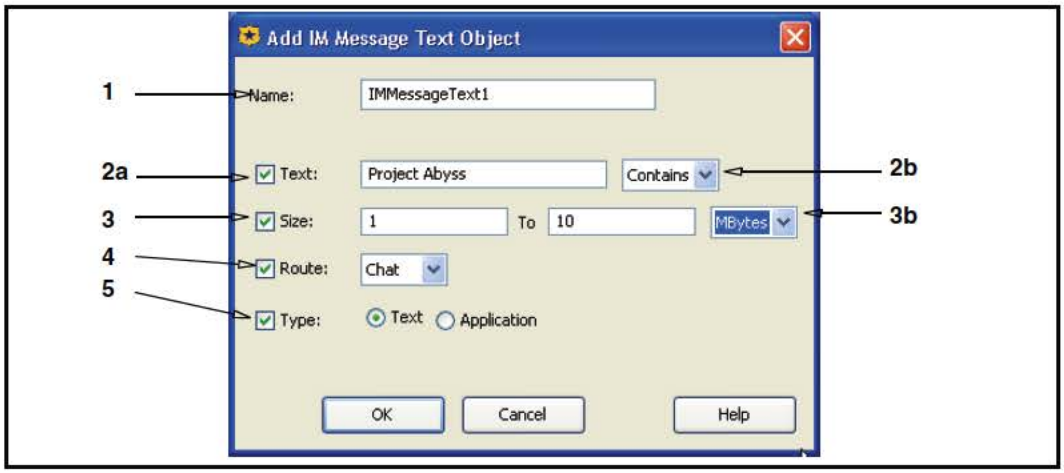


1. In the **Name** field, enter a name for the object or accept the default.
2. To trigger by file name:
   a. Select **File**; in the **File** field, specify a file name;
   b. From the drop-down list, select if file is matched exactly (**Exact Match**), if the file contains the name (**Contains**), or matched by regular expression (**RegEx**).
3. To trigger by message size:
   a. Select **Size** and enter a range.
   b. From the drop-down list, select the size attribute: **Bytes**, **Kbytes**, **MBytes**, or **GBytes**.

IM Message Text

Specifies the rule is applied to IM message text, which can be triggered by any or all of the following: matching content keywords, message size, service type, and whether the content type is text or application.

To specify IM message text parameters:



1. In the **Name** field, enter a name for the object or accept the default.
2. To trigger by content keywords:
   a. Select **Text**; in the **Text** field, specify a keyword.
   b. From the drop-down list, select if the file contains the text (**Contains**), or if it is to be matched by regular expression (**RegEx**).
3. To trigger by message size:
   a. Select **Size**; enter a range.
   b. From the drop-down list, select the size attribute: **Bytes**, **Kbytes**, **MBytes**, or **GBytes**.
4. To specify the message route, select Route. From the drop-down list, select **Service**, **Direct**, or **Chat**.
5. To specify message type, select **Text** or **Application**.
   - **Text** specifies messages entered by a user.
   - **Application** specifies messages sent by the client application, such as typing notifications.

IM Message Reflection

Allows policy to test whether or not reflection is enabled for the current message and, if enabled, whether it is possible.

- **Succeeded**—IM reflection is enabled and is performed for this message.
- **Failed**—IM reflection is enabled, but not possible for this message because the recipient is not connected through this SG appliance.
- **Disabled**—IM reflection is not enabled for this message.

The objects are automatically named based on the selection and can be used in any rule.

Streaming Content Type

Specifies streaming protocols.

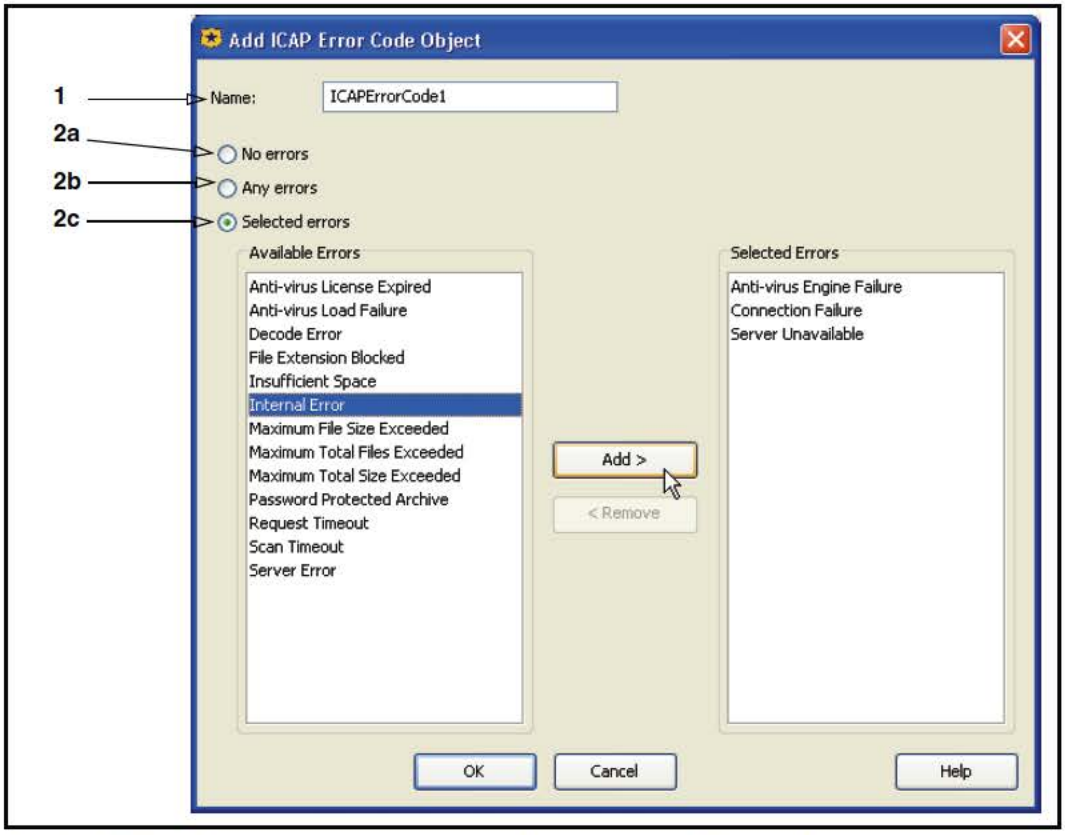
To specify streaming protocols:

1. In the **Name** field, enter a name for the object or accept the default.
2. Select **All Streaming Content** (all protocols become selected), or one or more streaming protocols.
3. Click **OK**.

ICAP Error Code

Defines an object that recognizes one or more ICAP error codes returned during an antivirus scan. The rule applies if the scan returns the specified errors.

To specify ICAP error codes:

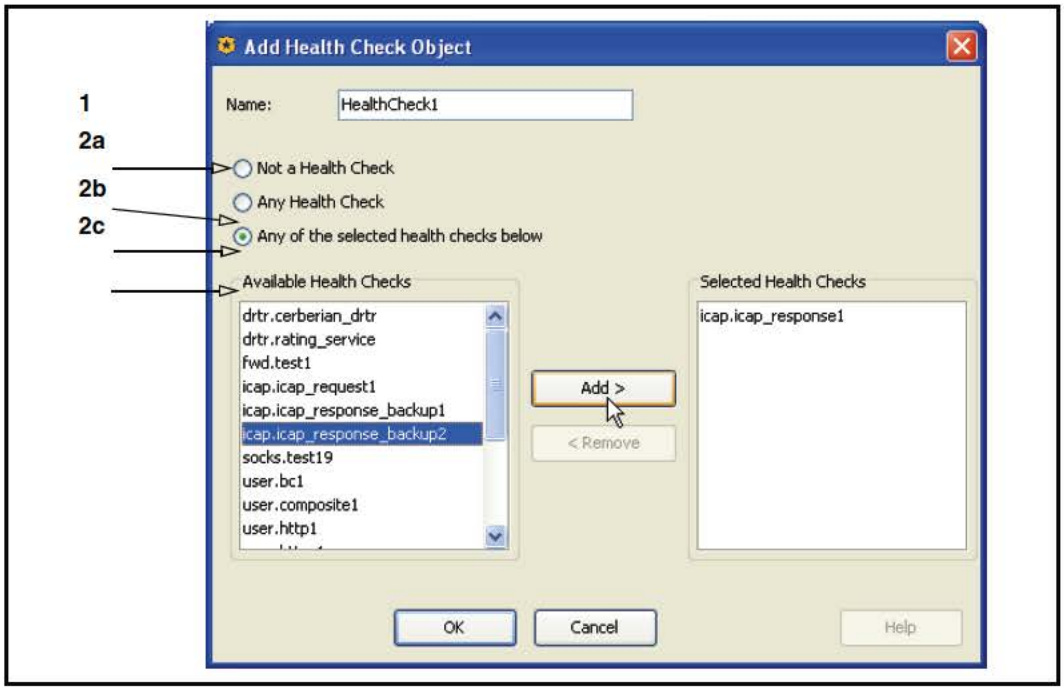


1. In the **Name** field, enter a name for the object or accept the default.
2. Select an option:
   a. **No errors**—An ICAP scan was performed without scanning errors.
   b. **Any errors**—An ICAP error code was returned during a scan.
   c. **Selected errors**—An ICAP error code of a specific type or types. In the **Available Errors** field, select one or more ICAP error codes (press and hold the Control key to select more than one type or the Shift key to select a block of types). Click **Add**.
3. Click **OK**.

Health Check

This condition tests whether the current transaction is a health check transaction. Optionally, the condition tests whether the transaction is that of a specific health check.

To create a Health Check object:



1. Select one of the following:
   - **Not a health check**: Transaction is not identified as a health check.
   - **Any Health Check**: A health check service of any type was matched.
   - **Any of the selected health checks below**: A health check of the selected types was matched.
2. If you selected **Any of the selected health checks below**:
   a. Select one or more error types (use Control + Left-click to highlight multiple errors).
   b. Click **Add** to move the errors to the **Selected** field.
   c. Name the object or accept the default name.
3. Click **OK**.

Health Status

This conditions tests whether the target of the specified health check is health or sick.

Combined Service Objects

Allows you to create an object that combines different service types. Refer to "Using Combined Objects" on page 128.

Service Column/Policy Layer Matrix

The following matrix lists all of the **Service** column objects and indicates which policy layer they apply to.

Object	Admin Auth	Admin Acc	DNS Acc	SOCKS Auth	SSL Int	SSL Acc	Web Auth	Web Acc	Web Cont	Fwding
Using HTTP Transparent Authentication								x		
Request Forwarded						x				
Virus Detected								x		
Client Protocol						x		x	x	x
Service Name		x						x		
Protocol Methods								x	x	x
SSL Proxy Mode						x				
IM File Transfer								x		
IM Message Text								x		
IM Message Reflection								x		
Streaming Content Type								x		
ICAP Error Code								x		
Health Status								x		x
Health Check						x				x
Combined Objects						x		x	x	x

Time Column Object Reference

A *time* object specifies a block of time or time trigger that determines client access regarding other parameters in the rule (such Web sites and content types). Currently, the **Time** object is only applicable to the Web Access Layer.

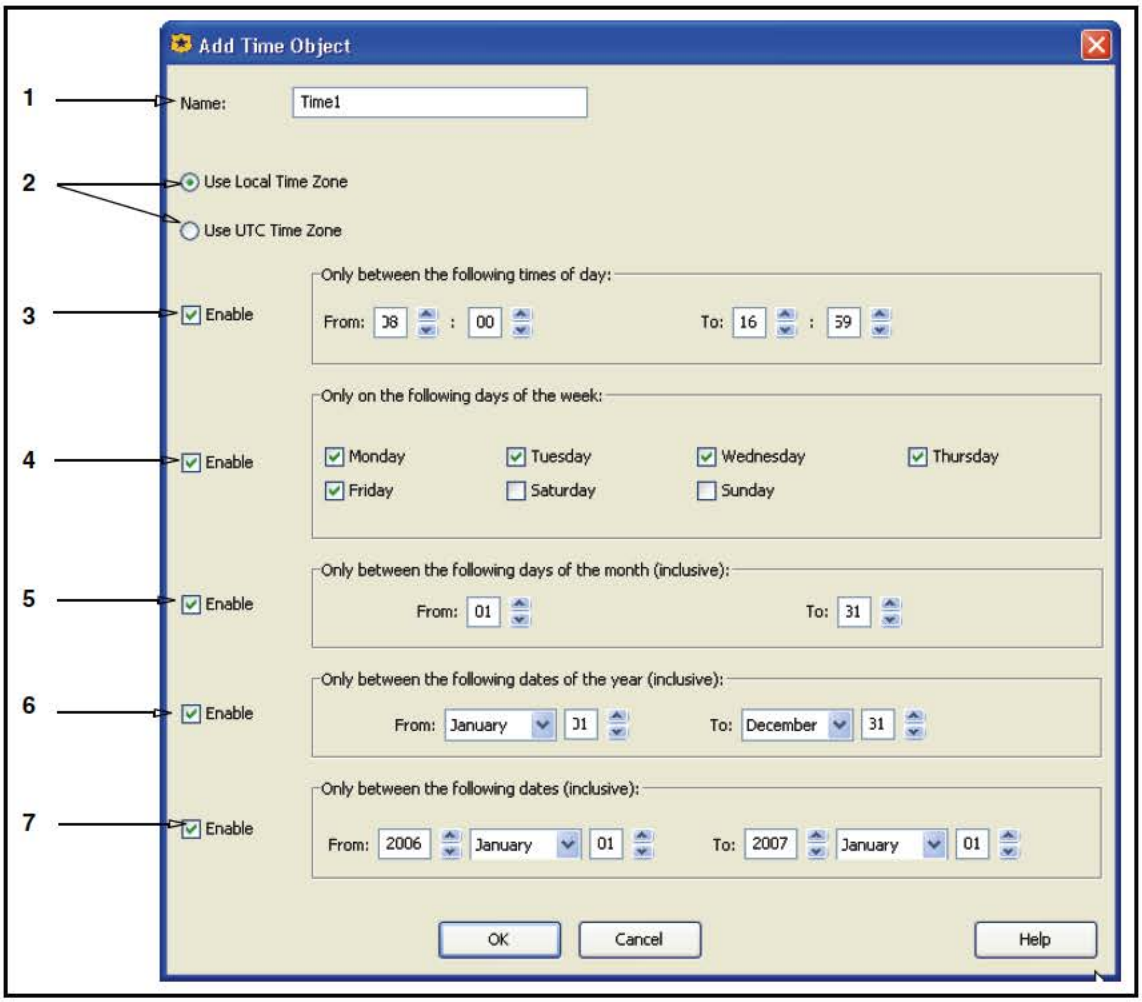
Any

Applies anytime.

Time

Specifies the time restrictions.

To configure time restrictions:



1. In the **Name** field, enter a name for the object or leave to accept the default.
2. Select **Use Local Time Zone** or **Use UTC Time Zone**.

   Local time sets the rule to follow the SG appliance internal clock. UTC sets the rule to use the Universal Coordinated Time (also known as Greenwich Mean Time or GMT).
3. To specify a range for any given day, select **Enable**; in the **Specify Time of Day Restriction (hh:mm)** field, configure the times. The time style is military.

   The range can be contained within one 24-hour calendar day, or overlap days. For example, configuring the time to range from 22:00 to 06:00 sets a limit from 10 at night to 6 the following morning.
4. To specify a day of the week restriction, select **Enable**; in the **Specific Weekday Restriction** field, select one or more days.
5. To specify a day of the month range restriction, select **Enable**; in the **Specify Day of Month Restriction** field, select the days, which are numbered from 01 to 31. To limit the range to specific day, configure the numbers to be the same. For example, selecting 22 and 22 specifies the rule to apply only the 22nd day of every month.

6. To specify a restriction that spans one or more months, select **Enable;** in the **Specify Annually-Recurring Date Restriction** field, select the month and day ranges. This calendar restriction applies every year unless the restriction is altered.

   Overlapping months is allowed, similar to the behavior of day ranges in Step 3.

7. To specify a one-time only restriction, select **Enable**; in the **Specify Non-Recurring Date Restriction** field, select the year, month, and day ranges. This calendar restriction applies only during the time specified and will not repeat.

8. Click **OK**.

Combined Time Object

Allows you to combine a time object that adheres to multiple time restrictions. See "Using Combined Objects" on page 128.

Time Column/Policy Layer Matrix

The following matrix lists all of the **Time** column objects and indicates which policy layer they apply to.

Object	Admin Auth	Admin Acc	DNS Acc	SOCKS Auth	SSL Int	SSL Acc	Web Auth	Web Acc	Web Cont	Fwding
Time			x					x		
Combined Objects			x					x		

Action Column Object Reference

An *action* object determines which action to take if other parameters, such as source, destination, service, and time requirements validate the rule

Important: Because of character limitations required by the generated CPL, only alphanumeric, underscore, and dash characters can be used to define an action object name.

Allow

This is a static object. Selecting this overrides other related configurations and the specified user requests are allowed.

Deny

This is a static object. Selecting this overrides other related configurations and denies the specified user requests.

Force Deny

This is a static object. Forces a request to be denied, regardless if rules in subsequent layers would have allowed the request.

Force Deny (Content Filter)

This is a static objects Forces a request to be denied, regardless if rules in subsequent layers would have allowed the request. In the access logs, the Content Filter moniker allows you to identify policy denies based on content filtering versus other reasons.

Allow Content From Origin Server

This is a static object.

Connect Using ADN When Possible/Do Not Connect Using ADN

These are static objects. Connect Using ADN When Possible instructs the SG appliance to use the byte caching tunnels (used in Application Delivery Network (ADN) deployments). Do Not Connect Using ADN prevents the use of tunnel connections.

Allow Read-Only Access

This is a static object. Grants full access to view data on the appliance.

Allow Read-Write Access

This is a static object. Grants full access to view and manipulate data on the appliance.

Do Not Authenticate

This is a static object. Selecting this overrides other configurations and the specified users are not authenticated when requesting content.

Authenticate

Creates an authentication object to verify users. An authentication realm must exist on the SG appliance to be selected through VPM.

Note: In the SOCKS Authentication policy layer, the object is **SOCKS Authenticate**.

To create an Authenticate object:

1. In the **Name** field, enter a name for the object or leave as is to accept the default.
2. From the **Realm** drop-down list, select an authentication realm, which must already exist on the SG appliance.
3. Optional (in the Web Authentication policy layer only): from the **Mode** drop-down list, select a mode. The mode determines the way the SG appliance interacts with the client for authentication specifying the challenge type and the accepted surrogate credential:
   - **Auto**—The default; the mode is automatically selected, based on the request. Selects among proxy, origin-IP, and origin-IP-redirect, depending on the type of connection (explicit or transparent) and the transparent authentication cookie settings.
   - **Form Cookie**—For forms-based authentication: cookies are used as surrogate credentials. The cookies are set on the OCS domain only, and the user is presented with the form for each new domain. This mode is most useful in reverse proxy scenarios where there are a limited number of domains.
   - **Form Cookie Redirect**—The user is redirected to the authentication virtual URL before the form is presented. The authentication cookie is set on both the virtual URL and the OCS domain. The user is only challenged when the credential cache entry expires.
   - **Form IP**—The user's IP address is used as a surrogate credential. The form is presented whenever the user's credential cache entry expires.
   - **Form IP Redirect**—This is similar to **Form IP** except that the user is redirected to the authentication virtual URL before the form is presented.
   - **Proxy**—For explicit forward proxies: the SG appliance uses an explicit proxy challenge. No surrogate credentials are used. This is the typical mode for an authenticating explicit proxy.
   - **Proxy IP**—The SG appliance uses an explicit proxy challenge and the client's IP address as a surrogate credential.

- **Origin**—The SG appliance acts like an OCS and issues OCS challenges. The authenticated connection serves as the surrogate credential.
- **Origin IP**—The SG appliance acts like an OCS and issues OCS challenges. The client IP address is used as a surrogate credential.
- **Origin Cookie**—For transparent proxies: for clients that understand cookies but do not understand redirects; the SG appliance acts like an origin server and issues origin server challenges. The surrogate credential is used.
- **Origin Cookie Redirect**—For transparent forward proxies: the client is redirected to a virtual URL to be authenticated, and cookies are used as the surrogate credential. The SG appliance does not support origin-redirects with the CONNECT method.
- **Origin IP Redirect**—Significantly reduces security; only useful for reverse proxy and when clients have unique IP addresses and do not understand cookies (or you cannot set a cookie). Provides partial control of transparently intercepted HTTPS requests. The client is redirected to a virtual URL to be authenticated, and the client IP address is used as a surrogate credential. The SG appliance does not support origin-redirects with the CONNECT method.
- **SG2**—The mode is selected automatically, based on the request using the SGOS 2.x-defined rules.

4. (Optional) If you selected a **Form** mode in Step 3, the **Authentication Form**, **New Pin Form**, and **Query Form** drop-down lists becomes active.
   - **Authentication Form**—When forms-based authentication is in use, this property selects the form used to challenge the user.
   - **New Pin Form**—When forms-based authentication is in use, this selects the form to prompt user to enter a new PIN.
   - **Query Form**—When forms-based authentication is in use, this selects the form to display to the user when a yes/no questions needs to be answered.

   Note: The **New Pin Form** and the **Query Form** are only used with RSA SecurID authentication.

   In most deployments, the default form settings should be adequate. However, if in your enterprise you have customized authentication forms configured (on the SG appliance Management Console: **Configuration > Authentication > Forms**), you can select them from the drop-down lists. For example, **HR_PIN**.
5. Click **OK**.

Users are prompted to provide a valid user name and password.

Authenticate Guest

Allows a user to be authenticated as a guest user. One scenario is to allow access to a user who might otherwise be considered unauthenticated. Another is where no authentication is required, but you want to track access. For more information, see the Controlling Access to the Internet and Intranet chapter in *Volume 4: Securing the Blue Coat SG Appliance*.

To create an Authenticate Guest object:

1. In the **Name** field, name the object or accept the default.
2. In the **Guest Username** field, enter the name the guest is given. This name appears in the access logs.
3. In the **Guest Realm** area, select one of the following options:
   - **Use realm:**
   - **Use realm from previous authenticate request:**
4. In the Guest Surrogate Refresh Time area, select one of the following options:
   - **Use realm's surrogate refresh time:**
   - **User surrogate refresh time:**

Add Default Group

A default group can be assigned to any realm. You can assign users to these groups, which are valid when authorization succeeds, fails, or not attempted. Default groups support guest users, which are users who are not authenticated against a realm, but are given a guest name and allowed access to specific information. For example, you create a default group that all guest users are assigned to, which makes it easier to track and log.

Default Groups are configured the same as described in "Group" on page 53.

Force Authenticate

Forces the user to authenticate even though the request is going to be denied for reasons that do not depend on authentication. This action is useful to identify a user before the denial so that the username is logged along with the denial. See *Volume 4: Securing the Blue Coat SG Appliance* for a description of the fields in this object.

Note: In the SOCKS Authentication policy layer, the object is **Force SOCKS Authenticate**.

Bypass Cache

This is a static object. Prevents the cache from being queried when serving a proxy request, and prevents the response from the origin server from being cached.

Do Not Bypass Cache

This is a static object. The SG appliance always checks if the destination is cached before going to the origin server; also, the content is cached if cacheable.

Bypass DNS Cache

This is a static object. Prevents the request from querying the DNS cache list of resolved lookup names or addresses.

Do Not Bypass DNS Cache

This is a static object. The SG appliance always queries the DNS cache list of resolved lookup names or addresses.

Allow DNS From Upstream Server

This is a static object. Allows the SG appliance to send requests for data not currently cached to DNS servers.

Serve DNS Only From Cache

This is a static object. Instructs the SG appliance to only serve a DNS request from content that is already cached.

Enable/Disable DNS Imputing

These are static objects. If DNS imputing is enabled, the SG appliance appends the suffixes in the DNS imputing list to host names looked up when the original name is not found.

Check/Do Not Check Authorization

These are static objects. These actions control whether or not the SG appliance forces a request to be sent to an upstream server every time to check authorization, even if the content is already cached. The check action is not usually required for upstream origin content servers performing authentication, as the SG appliance automatically tracks whether content required authentication in each case. However, it can be required when an upstream proxy is performing proxy authentication because of the way some proxies cache credential information, causing them not to reliably challenge every request. When requests are directed to an upstream proxy which operates in this manner, enabling Check Authorization ensures that all such requests are properly authorized by the upstream proxy before the content is served from the local cache.

Always Verify

This is a static object. Cached content is always verified for freshness for the sources, destinations, or service specified in the rule. For example, the CEO and Executive Staff always require content to be the most recent, but everyone else can be served from the cache.

Use Default Verification

This is a static object. Overrides the **Always Verify** action and instructs the SG appliance to use its default freshness verification.

Block/Do Not Block PopUp Ads

These are a static objects. Blocks or allows pop up windows. Blue Coat recommends creating separate Web Access policy layers that only contain pop up blocking actions. Furthermore, many Web applications require pop up windows. As it is unlikely that your Intranet contains pages that pop up unwanted advertising windows, Blue Coat recommends disabling pop up blocking for your Intranet. For example:

- Web Access rule 1: Specify the Intranet IP address and subnet mask in the **Destination** column and select **Do Not Block Popup Ads** in the **Action** column.
- Web Access rule 2: Select **Block Popup Ads** in the **Action** column.

As you continue to modify policy, you can add more policy layers to block or allow specific IP addresses, but the policy layer as defined in the Web Access rule 2 above *must* always be positioned last. Blocking pop up ads is the default if a previous policy rule does not trigger.

For more concept information about pop up windows, see Section A: "Blocking Pop Up Windows" on page 170.

Force/Do Not Force IWA for Server Auth

These are static objects. When configured for explicit proxy, Internet Explorer (IE) does not support an IWA challenge from an origin server. If **Force IWA for Server Auth** is applied, the SG appliance converts the 401-type server authentication challenge to a 407-type proxy authentication challenge, which IE supports. The SG appliance also converts the resulting Proxy-Authentication headers in client requests to standard server authorization headers, which allows an origin server IWA authentication challenge to pass through when IE is explicitly proxied through the SG appliance.

Log Out/Do Not Log Out Other Users With Same IP

These are static objects. If more than one user is logged in at the IP address of the current transaction, this property logs out all users from the current IP address except the user of the current transaction.

Log Out/Do Not Log Out User

These are static objects. This property logs out the login referenced by the current transaction.

Log Out/Do Not Log Out User's Other Sessions

These are static objects. If a user is logged in at more than one IP address, this property logs out the user from all IP address except the IP address of the current transaction.

Reflect/Do Not Reflect IM Messages

These are static objects. IM traffic can be contained and restricted to the network so that it never reaches the IM server. A hierarchy of SG appliances manage the traffic and routes it depending on each SG appliance fail open and fail closed configurations. For detailed information about this deployment, refer to the Instant Messaging chapter in *Volume 3: Web Communication Proxies*.

Support/Do Not Support Persistent Client Requests

These are static objects. Allowing persistent connections to the SG appliance from clients reduces load improves the all-around performance of the network. This object specifies whether or not to allow persistent server connections.

Support/Do Not Support Persistent Server Requests

These are static objects. If the back-end authentication authority (such as LDAP, RADIUS, or the BCAAA service) receives large numbers of requests, you can configure the SG appliance to use persistent connections to the server. This dramatically reduces load on the back-end authentication authority and improves the all-around performance of the network. This object specifies whether or not to allow persistent server connections.

Block/Do Not Block IM Encryption

These are static objects. AOL IM provides the option for clients to send encrypted messages through both standard messaging (through the IM service) and direct connection messaging. These objects allow you to block or not block the ability to send encrypted messages through AOL IM. For detailed information about security benefits of this feature, refer to the Instant Messaging chapter in *Volume 3: Web Communication Proxies*.

Require/Do Not Require Client Certificate

These are static objects. For the SSL Proxy, specifies whether a client (typically a browser) certificate is required or not.

- In forward proxy deployments, this is used to either request consent certificates or to support certificate realm authentication.
- In reverse proxy deployments, client certificates are requested for certificate realm authentication.

Also, see "Set Client Certificate Validation" on page 93.

Trust/Do Not Trust Destination IP

These are static objects. The Trust Destination IP object instructs the SG appliance to trust the IP address sent by the client, forgoing a DNS lookup. This is designed for transparent and ADN deployments. Conversely, the Do Not Trust Destination IP instructs the SG appliance to always perform a DNS lookup.

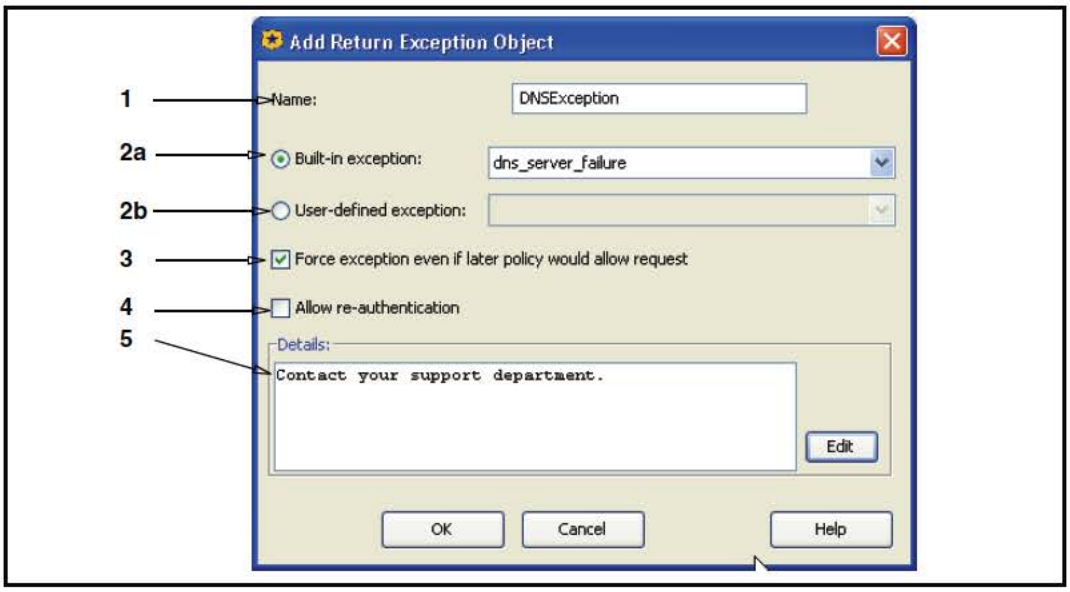
Deny

This object provides the same functionality as the "Force Deny" on page 84 object, but provides the option to re-allow authentication and insert substitution strings.

Return Exception

Allows you to select exception types and associate a custom message, if desired. Blue Coat provides a list of standard exceptions, but VPM also accepts user-defined values.

To create a Return Exception object:



1. In the **Name** field, enter a name for the object or leave as is to accept the default.
2. Perform one of the following:
   a. Standard exception type: select one from the **Built-in exception** drop-down list.
   b. Custom exception (which already must be defined on the SG appliance) type: select one from the **User-defined exception** drop-down list.
3. Optional: Select **Force exception even if later policy would allow request** to supersede other policy that applies to this request.
4. Optional: Select **Allow re-authentication** to allow the user to re-enter credentials should the first attempt fail.
5. Optional: in the **Details** field, enter a message that is displayed along with the summary and exception ID on the exception page displayed to the user when the exception is returned.

The above example creates an object named **DNSException2** that upon a DNS server failure returns a message to the user instructing them to contact their support person.

To create custom exceptions, see Section D: "Defining Exceptions" on page 176.

Return Redirect

Aborts the current transaction and forces a client request to redirect to a specified URL. For example, used to redirect clients to a changed URL; or redirecting a request to a generic page stating the Internet access policy. Applies only to HTTP transactions.

Note: Internet Explorer (IE) ignores redirect responses from FTP over HTTP requests, although Netscape Navigator obeys the redirect. To avoid problems with IE, do not use redirect when `url.scheme=ftp`.

If the URL that you are redirecting the browser to also triggers a redirect response from your policy, then you can put the browser into an infinite loop.

In the **Name** field, enter a name for the object (or leave as is to accept the default); in the **URL** field, enter the redirect destination URL.

Example

An object that redirects clients to an HTML policy statement page.

Set Client Certificate Validation

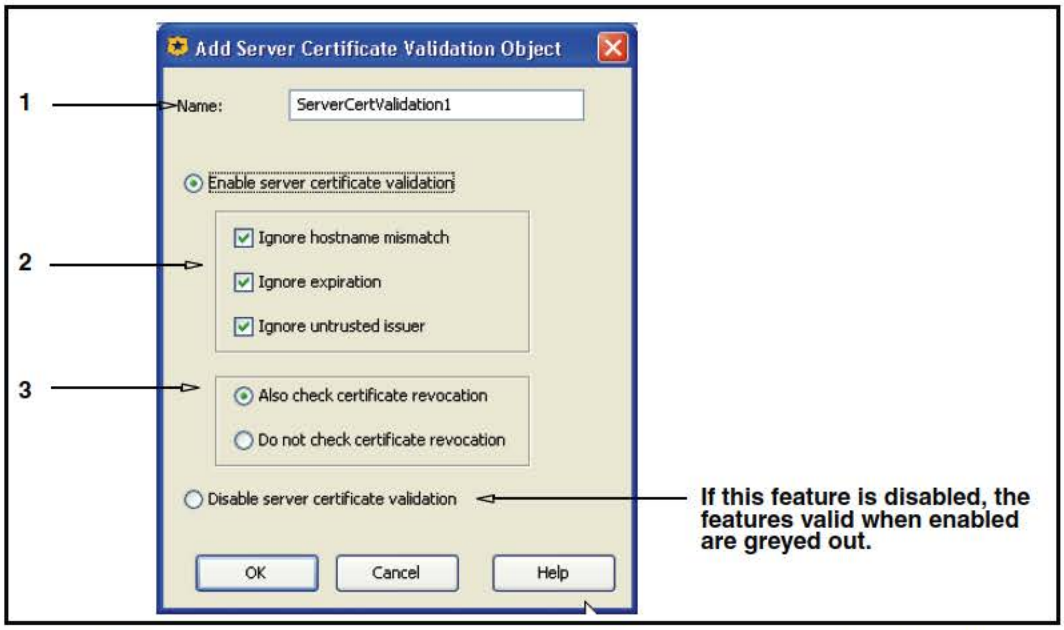
If a client certificate is requested (see "Require/Do Not Require Client Certificate" on page 91), this object specifies whether the requested client certificate is validated.

If **Also check certificate revocation** is selected, this is checked from a Certificate Authority. For information on using CRL, refer to *Volume 2: Proxies and Proxy Services*.

Set Server Certificate Validation

This feature is enabled by default. The SSL Proxy performs checks on server certificates. To mimic the overrides supported by browsers, the SSL Proxy can be configured to ignore failures for the first three checks in the list.

To add a Server Certificate Validation object:



1. In the **Name** field, enter a name for the object or leave as is to accept the default.
2. (Optional) Select one or more to ignore certain failures:
   - **Ignore a hostname mismatch**: Ignores the comparison of hostname in URL and certificate (intercepted connections only).
   - **Ignore certificate expiration**: Ignores the verification of certificate dates.
   - **Ignore untrusted issuer**: Ignores the verification of issuer signature.
3. The certificate revocation list (CRL) option:

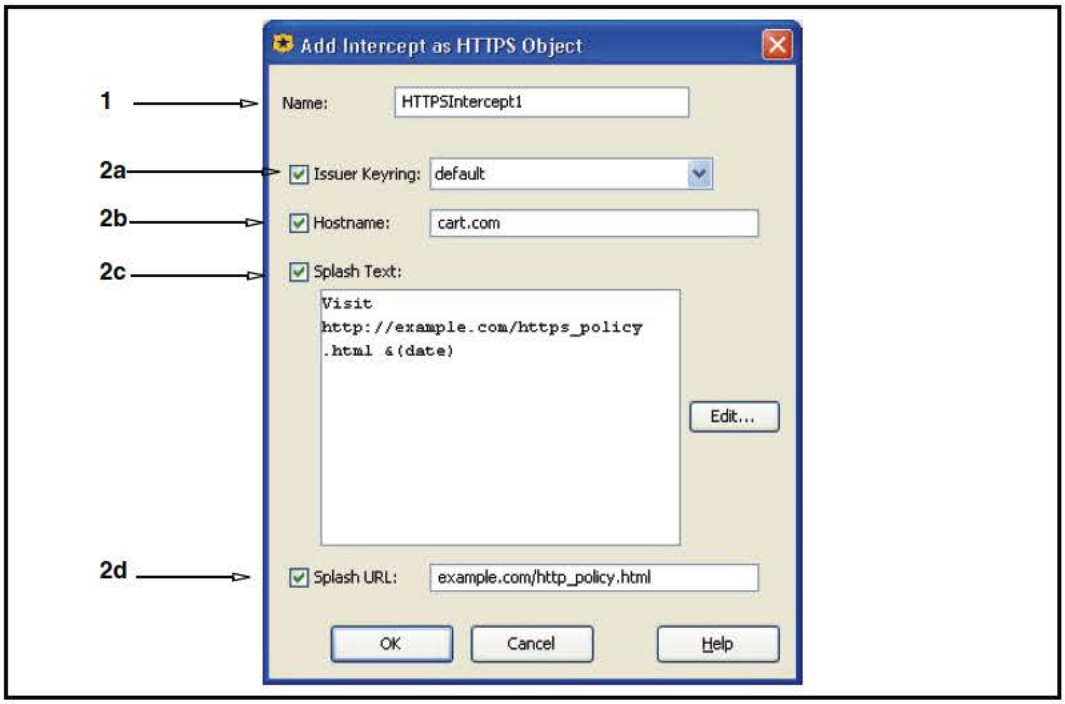
   If **Also check certificate revocation** is selected, this is checked from a Certificate Authority. For information on using CRL, refer to *Volume 4: Securing the Blue Coat SG Appliance*.

Note: Two built-in exceptions can be used to notify the user that the verification of the server's certificate failed: `exception.ssl_server_cert_expired` and `exception.ssl_server_unknown_ca`. For information on using exceptions, see Chapter 4: Advanced Policy, Section D: "Defining Exceptions" on page 176.

Enable HTTPS Intercept

The HTTPS Intercept object enables the SG appliance to act as an HTTPS Forward Proxy, providing performance gains and security (authentication, content filtering, anti-virus scanning) for HTTPS traffic before it is delivered to clients. This object allows HTTPS content to be intercepted and examined.

To create an HTTPS Intercept object:



1. In the **Name** field, enter a name for the object or leave as is to accept the default.
2. To allow SSL content to be examined, select:
   a. **Issuer Keyring**: Accept the default keyring or select this option and from the drop-down list select a previously generated keyring. This is the keyring used for signing emulated certificates.
   b. **Hostname:** The hostname you enter here is the hostname in the emulated certificate.
   c. **Splash Text**: The limit is 200 characters. The splash text is added to the emulated certificate as a certificate extension. The splash text is added to the emulated certificate as a certificate extension. For example:

      ```
      Visit http://example.com/https_policy.html
      ```

      To add substitution variables to the splash text, click **Edit** and select from the list.
   d. **Splash URL**: The splash text is added to the emulated certificate as a certificate extension.

The SSL splash can be caused by such occurrences as when a browser receives a server certificate signed by an unknown CA, or a host miss-match.

Note: Not all browsers display the splash text and splash URL correctly.

Enable HTTPS Intercept on Exception

An HTTP Intercept on Exception object is used to intercept SSL traffic if there is an exception, such as a certificate error or policy denial. This differs from the HTTPS Intercept object, which intercepts all HTTPS traffic. For information on configuring an HTTPS Intercept on Exception object, see"Enable HTTPS Intercept" on page 94.

Disable SSL Intercept

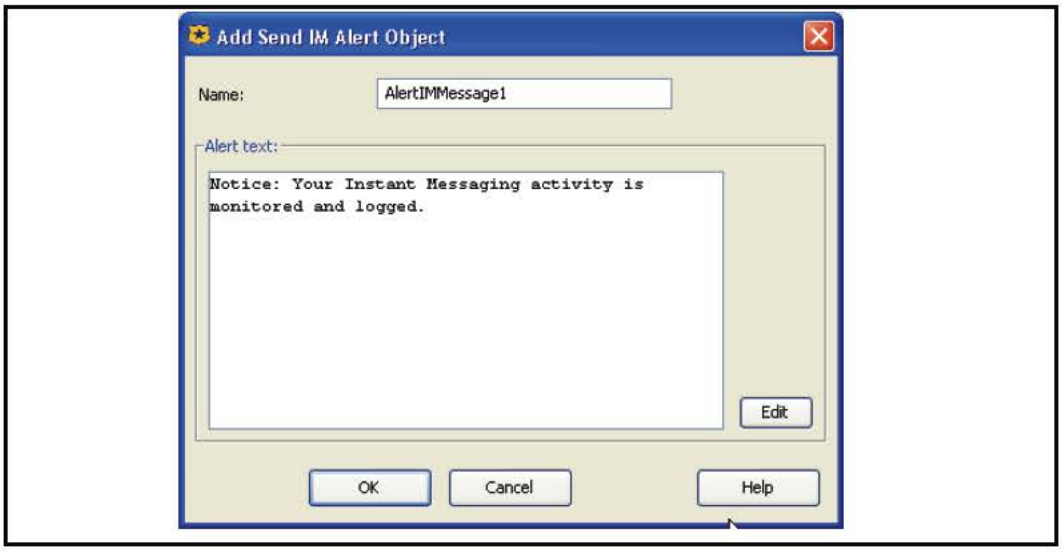
This is a static object. Selecting this object disables HTTPS interception.

Send IM Alert

Defines a message that is sent to an IM user by the SG appliance. The message is triggered by the IM parameters defined in the policy (for example, client login, sent or received messages, and buddy notification). *Volume 3: Web Communication Proxies* provides more information about regulating IM through the SG appliance, as well as VPM examples.

Example

A message that informs IM users their messaging is logged.



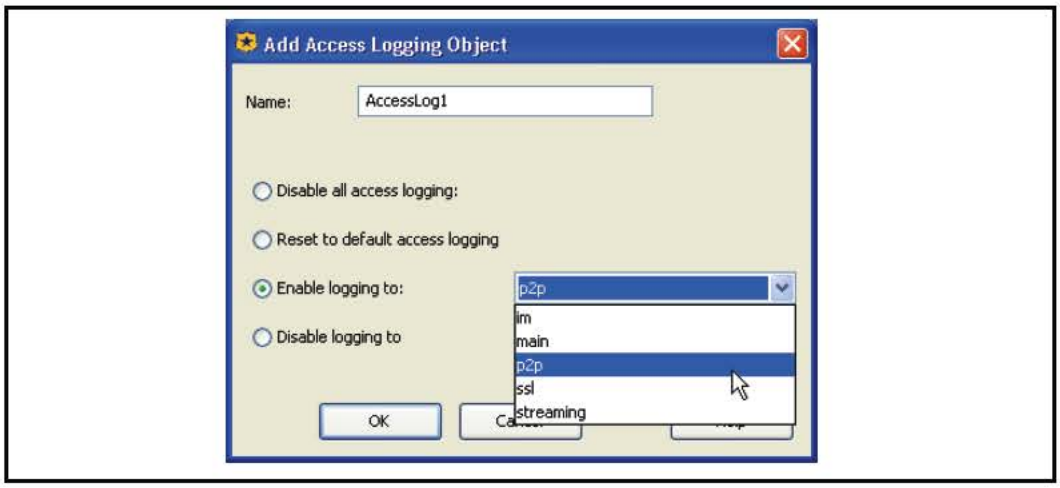
Modify Access Logging

Defines access logging behavior.

- **Disable all access logging**—No activity is logged for the requests matched by the rule.
- **Reset to default logging**—Resets to logging the request to the default log specified by the SG appliance configuration, if one exists.
- **Enable logging to**—Enables logging of requests matched by this rule to the specified log.
- **Disable logging to**—Disables logging of requests matched by this rule to the specified log.

Example:

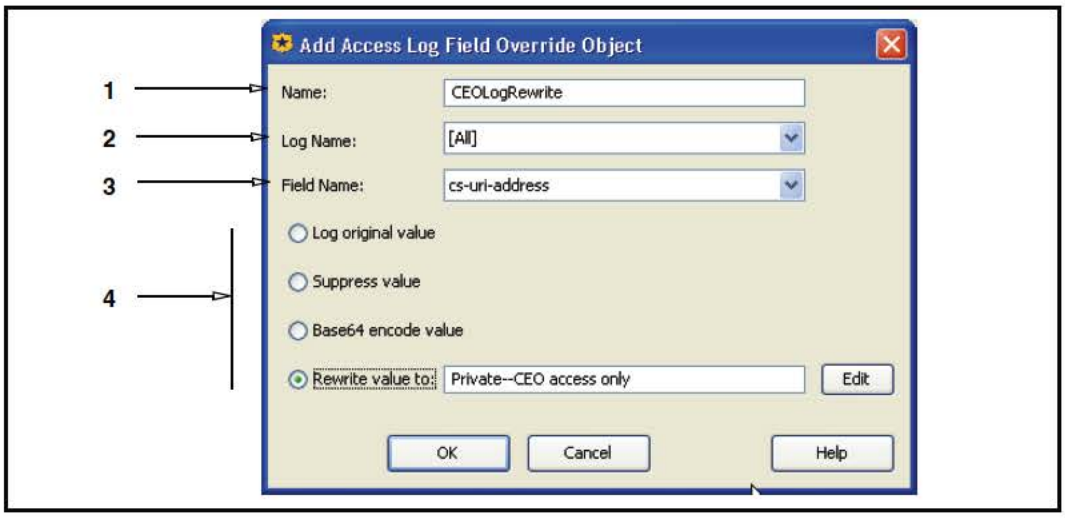
Enable logging P2P logging for this rule.



Override Access Log Field

Allows you to manipulate access log entries. For any specific log value, you can suppress the value, encode the value in Base64, or rewrite the value.

To override access log fields:



1. In the **Name** field, enter a name for the object or leave as is to accept the default.
2. From the **Log Name** drop-down list, select a log (must already be configured on the SG appliance).
3. From the **Field Name** drop-down list, select an access log field.
4. Select one of the following:
   - **Log original value**—Records unmodified value in the access log.
   - **Suppress value**—Prevents value from appearing in the access log.
   - **Base64 encode value**—Records an encoded version of the value in the access log.

- **Rewrite value**—In the field, enter a string that replaces the value. Clicking **Edit** calls the Select The Rewrite String dialog. The substitution variables instruct the SG appliance to append specific information to the object. The variables are categorized alphabetically, according to prefix.

  Note: Some variables do not have prefixes, which allows you to substitute the value with information defined by other field types.

5. Click **OK**.

The above example creates an object called **CEOLogRewrite** that suppresses log entries so persons, such as support personal, cannot view economically sensitive information to gain improper knowledge.

Rewrite Host

Rewrites host component of a URL, specifying either Windows Media, Real Media, or all protocols. Use this to redirect the request to a different host. For example, rewrite `www.traning1.com` to `www.training2.com`. You can create and name multiple rewrites, but you can only specify one per rule.

To specify a rewrite:

1. In the **Name** field, enter a name or leave as is to accept to the default.
2. From the **Scheme drop**-down list, **Windows Media**, **Real Media**, or **All** to rewrite all URLs, regardless of protocol.
3. In the **Pattern** field, enter a host name.
4. In the **Replacement** field, enter the name the pattern is rewritten to.
5. Click **OK**.

Reflect IP

Specifies which IP address is used when making connections to upstream hosts.

To create a Reflect IP object:

1. In the **Name** field, enter name for the object or leave as is to accept the default.
2. In the **In outgoing client IP, reflect** field, select one of the following:
   - **Do not reflect IP**—Disables reflecting IPs; the SG appliance uses the IP address of the interface that request is sent out on.
   - **Incoming client IP [IP spoofing]**—Reflects the client IP address.
   - **Incoming proxy IP**—Reflects the IP address of where the request arrived to.
   - **Proxy IP**—Specifies to reflect a specific IP of the SG appliance; enter the IP address in the field.
   - **Use services configuration**—Specifies whether to reflect IP in the configuration of the service which is used to process the request.
3. Click **OK**.

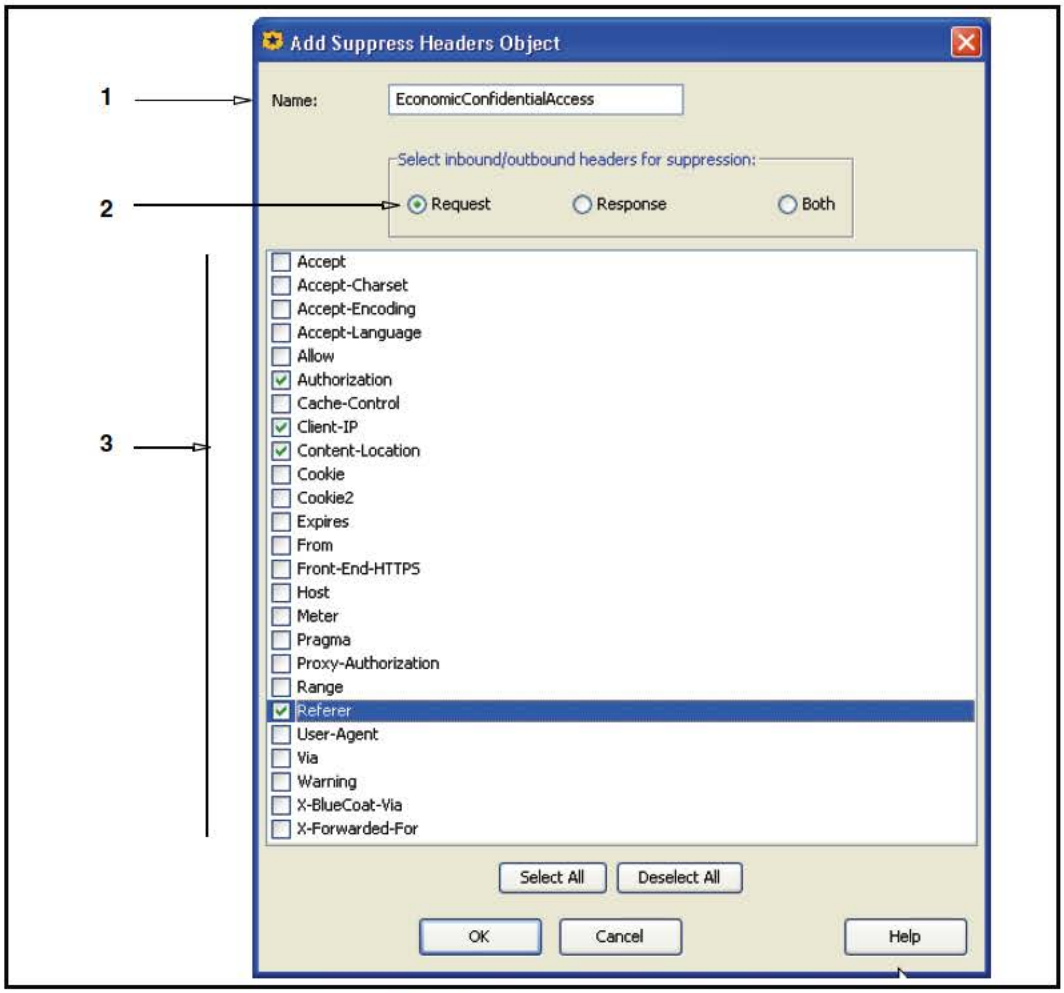
Example

The above example reflects another IP address configured on the SG appliance.

Suppress Header

Specifies one or more standard headers that are suppressed (not transmitted) on the outbound request, the outbound response, or both.

To create a Suppress Header object:



1. In the **Name** field, enter name for the object or leave as is to accept the default.
2. Select **Request**, **Response**, or **Both**. The valid headers vary for requests and responses. **Both** displays a small subset of headers valid for requests and responses.
3. Select one or more header types from the list.
4. Click **OK**.

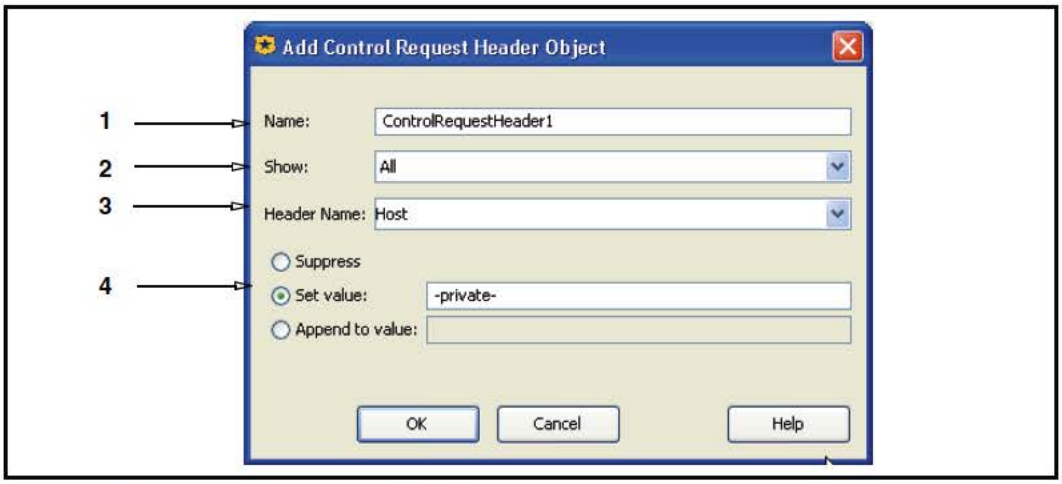
The above example creates an object called EconomicConfidentialAccess to be used in a rule suppresses headers so specified users can access economically sensitive information without people, such as support personal, being able to gain knowledge of sources.

Control Request Header/Control Response Header

Allows you to control and modify request or response headers by:

- Inserting a header with a specific value.
- Rewriting the value of a specific header.
- Suppressing a specific header.

To create a Request Header or Control Response Header object:



1. In the **Name** field, enter name for the object or leave as is to accept the default.
2. From the **Show** drop-list select the viewing field from **All** to **Standard** or **Custom**, as desired. **Standard** displays only the default standard headers. **Custom** displays any admin-defined headers that exist.
3. From the **Header Name** list, select a standard (pre-defined) header or a custom header if one has been defined.
4. Select an action:
   - Suppress—The header is not visible.
   - Set value—Replace the header with a string or value.
   - Append to value—Add a string or value to the existing header.
5. Click **OK**.

Notify User

This action displays a notification page in the user's Web browser. A user must read the notification and click an **Accept** button before being allowed to access the Web content. You can customize the following:

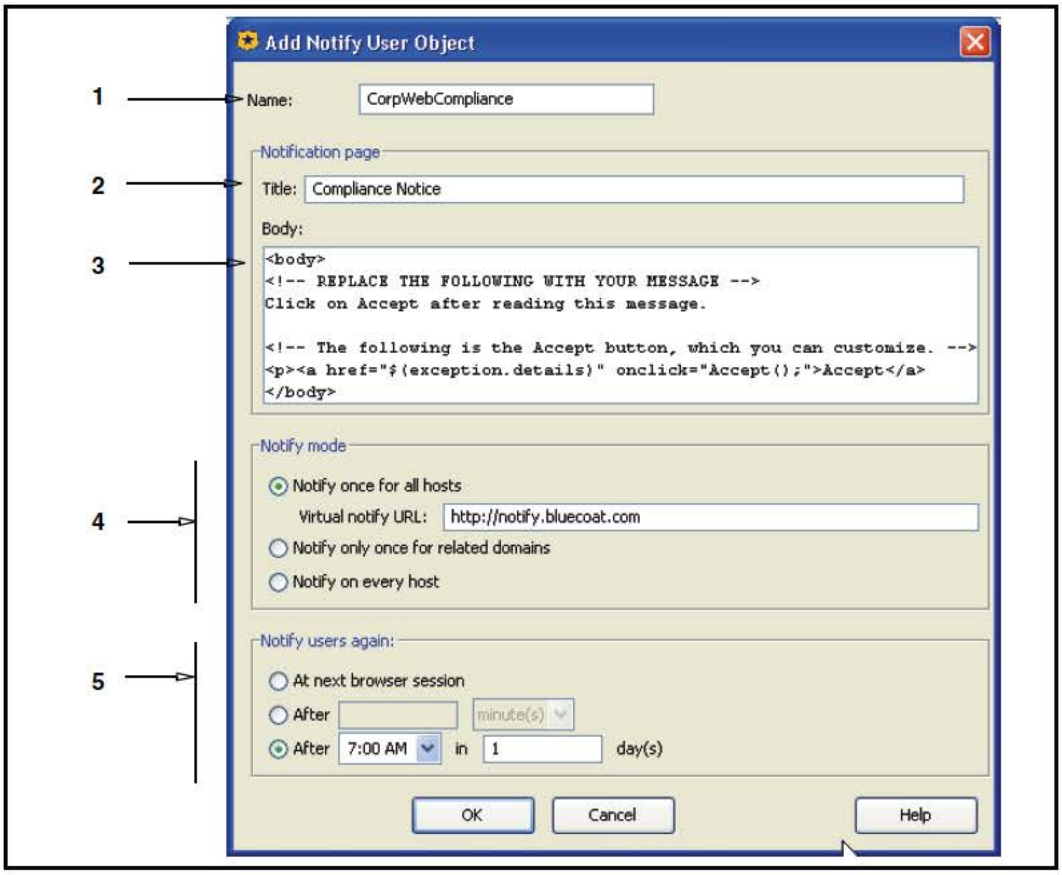
- The page title, notification message, and the **Accept** button.
- The conditions that cause a notification to be displayed again. By default, the notification is displayed each time a user begins a new Web browsing session (reboots, logs out, or closes all Web browser windows). You can configure re-notification to occur for each new visited host or Web site, or after a time interval.

Note: The **Accept** button click action is logged if HTTP access logging is enabled. A URL is logged that contains the string: `accepted-`*`NotifyName`*, where `NotifyName` is the name of the **Notify User** object.

This feature is designed to provide the following functionality:

- Web-use compliance: A compliance page is a customized notification page displayed on a user's Web browser when attempting to access the Internet. This page ensures employees read and understand the company's Acceptable Use Policy before Internet use is granted. Typically, a compliance notification is displayed each time a browser is opened, but you can configure a time condition to display the page at specific intervals or times of the day, week, or month.
- Coach users: A coaching page displays when a user visits a Web site that is blocked by content filtering policy. This page explains why the site is blocked, the consequences of un-authorized access, and a link to the site if business purposes warrants access. A coaching page is configured to display each time a user visits a new Web page that is barred by content filtering policy; however, you can also configure this page to appear at different time intervals.

To configure HTML notification:



1. In the **Name** field, enter a name for the object or leave as is to accept the default.
2. In the **Title** field, enter a name that is the title of the page (text only; no HTML is allowed).
3. In the **Body** field, compose a block of HTML that displays the message to the user. You can also customize the **Accept** link or button text. The HTML body must contain an **Accept** button or link. The default is:

```
<body><a href="$(exception.details)" onclick="Accept();">Accept</
a></body>
```

You can also use a button image (the image resides on an external Web server, as in the following example:

```
<body><a href="$(exception.details)" onclick="Accept();">
<img src="http://server.com/images/accept.png"> </a> </body>
```

If you use an HTML editor to compose code, you can paste it into the VPM; however, only copy the HTML from the `<body>` tag to the `</body>` tag.

4. Under **Notify mode**, select an option that determines notification when visiting a new Web site:

   - **Notify once for all hosts**—The notification page is displayed only once; this is used for configuring compliance pages. This option uses a Virtual Notify URL. If you must change the URL from the default value, please read the limitation section following this procedure.

     Note: This option might cause users to experience some noticeable Web browsing slowness.

   - **Notify only once for related domains**—The notify page reappears each time the user visits a new Web site; this is used for configuring coaching pages.

     Note: This option interferes with some Web advertising banners. In some cases, the notification page appears inside the banner. In other cases, banner ads are disabled by javascript errors. To fix these problems, do not serve notification pages for URLs that belong to the **Web Advertising**, **Advertising**, or **Web Ads** category. The actual name of this category varies with the content filtering vendor, and some vendors do not have an equivalent.

   - **Notify on every host**—The notify page reappears each time the user visits a new Web host. Blue Coat recommends that only highly experienced administrators employ this option. In addition to breaking banner ads, as described above in the previous option, this option, on some Internet Web sites, might cause Javascript errors that impair the functionality of the site.

5. Under **Notify users again**, select an option that specifies when the notification expires and re-notification is required:

   - **At next browser session**— The notification page does not reappear until the next browser session. When a user reboots, logs out, or closes all Web browser windows, this ends the browser session.
   - **After** (time interval)—Notification reoccurs after the defined elapsed time (minutes or hours); this is useful for coaching.
   - **After** (specific time)—Notification reoccurs at a specific time of day. You can specify an interval of days; this is useful for compliance.

     Note: The time is referenced from the local workstation. If a compliance page is configured, verify the workstations and SG appliance clocks are synchronized.

The above example creates a Notify Object with a custom message, set to display once a day after 7 AM.

Interactivities and Workarounds

If you must change the default Virtual Notify URL, consider the following:

- The Virtual Notify URL consists of an HTTP domain name or IP address (`http://`); a port number is optional.
- Do *not* use a host name that is explicitly defined as a *trusted site* on Internet Explorer 6 for Windows XP, Service Pack 2. Furthermore, only use domain names that contain dots. If you use domain names that do not contain dots, the HTTP redirects generated by the notification action causes Internet Explorer to display false warning messages each time the user is redirected from an untrusted site to a trusted site, or the other way around.
- For transparent proxy deployments, the domain name *must* be DNS-resolvable to an IP address that is in the range of destination IP addresses that are routed to the SG appliance.

Policy Interactions

This action generates CPL that might interfere with other policy or cause undesired behavior. Enhancements will occur in future SGOS releases. For this release, consider the following guidelines:

- Do not create VPM policy that modifies the `Cookie` request header.
- Do not create VPM policy that modifies the `Set-Cookie` and `P3P` response headers.
- Notification pages exist in the browser history. Therefore, if you click **Accept** and are taken to the requested page, then click the back button, you get the notification page again.
- If you have a chain of SG appliances, with different notification pages configured on each appliance in the chain, then each notification page *must* have a different object name.

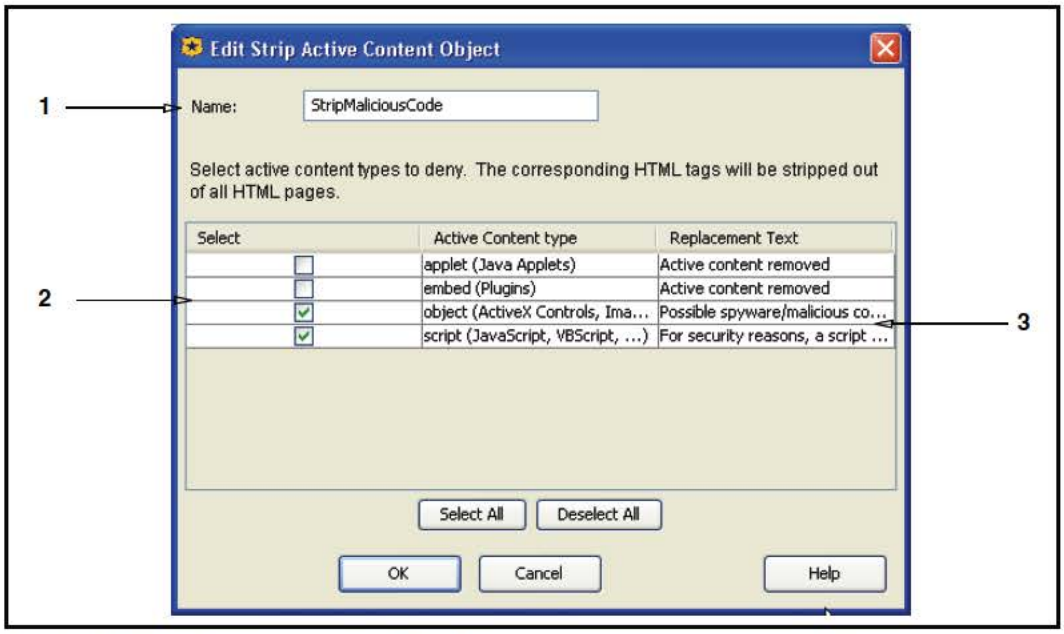
Strip Active Content

Strips HTTP tags from specified active content HTML pages. For each item you select for removal, you can also create a customized message that is displayed to the user.

Note: Pages served over an HTTPS tunneled connection are encrypted, so the content cannot be modified.

See Chapter 4: Advanced Policy, Section B: "Stripping or Replacing Active Content" on page 172 for detailed information about the different types of active content.

To create a Strip Active Content object:



1. In the **Name** field, enter name for the object or leave as is to accept the default.
2. Select **the active content to be stripped.**
3. The default message in the **Replacement Text** column is **Active Content Removed**. To replace the default message, double click the field, enter a message, and press Enter. See the examples in the screenshot, **Java applets have been removed**.

Exempting the SG Appliance

Stripping active content might interfere with Web applications deployed on your intranet. For example, if you create a policy rule that removes Java applets, and the destination defined in the rule contains an IP address of a SG appliance functioning as a proxy, the policy rule actually disables the Management Console because the Console itself is comprised of Java applets.

To prevent this, for each SG appliance functioning as a proxy, create a rule that exempts the IP address of the SG appliance from the stripping action.

1. Click **Add Rule**.
2. Click **Move Up**; the rule to exempt the SG appliance must precede the rule that strips active content.
3. In the **Destination** field, enter the SG appliance IP address.
4. With the IP address entered, right-click it in the **Destination** field and select **Negate** from the drop-down list.
5. In the **Action** field, enter the **Remove Active Contents, Java Apps** action.

Figure 3-10. Exempting a SG appliance IP Address

HTTP Compression Level

Allows you to set the level of compression to low, medium, or high. When configuring, consider that a higher compression level consumes more CPU resource.

Note: If you enable HTTP Compression using the VPM but do not specify the HTTP Compression Level using VPM policy, then by default the level is **Low**.

To specify an HTTP compression level:

1. Select a compression level option:
   - **Low**—Equivalent to compression level 1.
   - **Medium**—Equivalent to compression level 6.
   - **High**—Equivalent to compression level 9.
2. Click **OK**.

The object is automatically named as **Compression Level Low**, **Medium**, or **High**.

Set Client HTTP Compression

Specifies the behavior when the client wants the content in a different compression form than is in the cache.

To specify compression actions:

1. In the **Name** field, enter name for the object or leave as is to accept the default.
2. This object has two instructions:
   - A client requests compressed content, but only uncompressed content is available. Select to either compress the content before serving it, or serve uncompressed content.
   - A client requests uncompressed content, but only compressed content is available. Select to either uncompress the content before serving it, or serve compressed content.

   The default is to compress or decompress content, respectively, before serving it.
3. Click **OK**.

For recommended compression configurations, refer to *Volume 2: Proxies and Proxy Services*.

Set Server HTTP Compression

Enables or disables HTTP compression.

To specify compression options:

1. In the **Name** field, enter name for the object or leave as is to accept the default.
2. Select a compression option:
   - **Disable HTTP compression**—The default. Objects are not compressed.
   - **Use client HTTP compression options**—Default to the type of content requested by the client.
   - **Always request HTTP compression**—Force clients to always request compressed content.
3. Click **OK**.

For recommended compression configurations, refer to *Volume 2: Proxies and Proxy Services*.

Manage Bandwidth

Allows you to manage bandwidth for all protocols or specific protocols, on both inbound and outbound traffic.

To create a manage bandwidth object:

1. In the **Name** field, enter name for the object or leave as is to accept the default.
2. Select to limit bandwidth on the: **Client side** or **Server side**.
   - **Client side**—Traffic flowing between a client and the SG appliance.
   - **Server side**—Traffic flowing between a server and the SG appliance.
3. Select to limit bandwidth for: **Inbound** or **Outbound** traffic.
   - **Inbound**—Network packets flowing into the SG appliance. Inbound traffic mainly consists of packets originating at the origin content server (OCS) and sent to the SG appliance to load a Web object and packets originating at the client and sent to the SG appliance for Web requests.
   - **Outbound**—Network packets flowing out of the SG appliance. Outbound traffic mainly consists of packets sent to the client in response to a Web request and packets sent to an OCS or other service (such as a virus scanner) to request a service.
4. Select a **Bandwidth Class** from the drop-down list.
5. Click **OK**; click **Save Changes**.

For complete information about Bandwidth Management, refer to *Volume 5: Advanced Networking*.

ADN Server Optimization

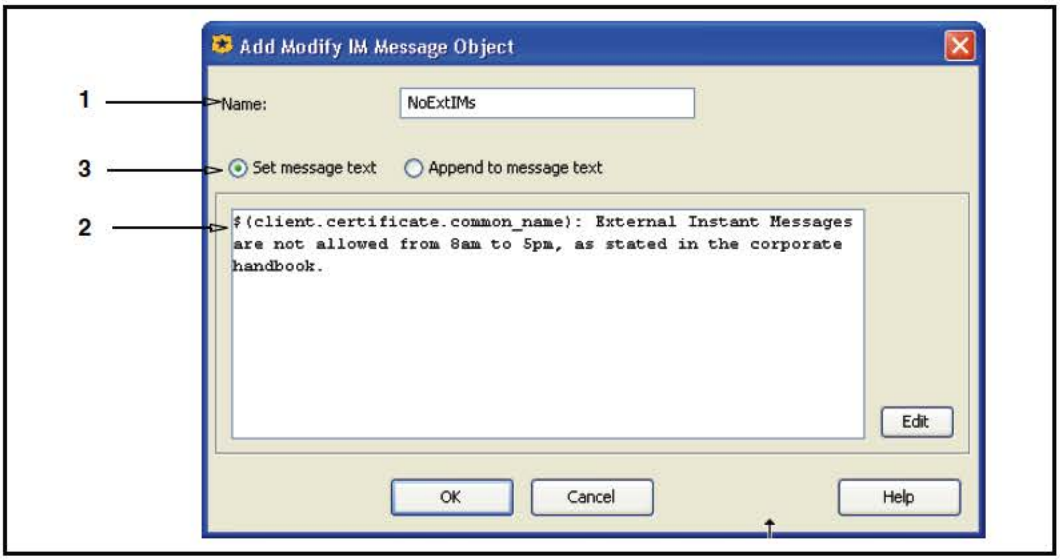
Specifies whether byte caching is employed on either (branch or core) or both sides of an Application Delivery Network connection (specified IP addresses in the rule). Byte caching reduces WAN latency.

- **Optimize traffic in both directions**: Apply Byte Caching to traffic coming and leaving the server.
- **Optimize only inbound traffic**: Apply Byte Caching only to traffic coming into the server.
- **Optimize only outbound traffic**: Apply Byte Caching only to traffic leaving the server.
- **Do not optimize traffic**: Do not allow Byte Caching on specified connections.

Modify IM Message

In IM clients, replaces or appends the given text that is displayed to IM messages in clients that are logged in through the SG appliance. For example, use with Time Object to inform users that Instant Messages sent outside the corporate network are not allowed during business hours.

To create an IM message modification object:



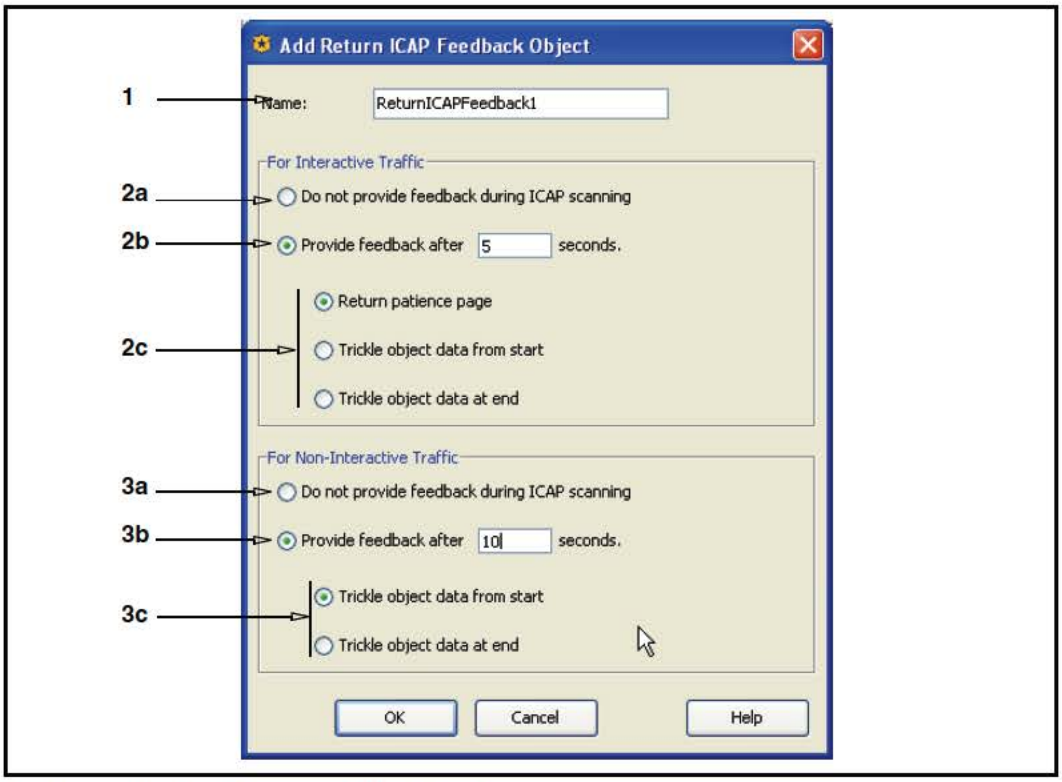
1. In the **Name** field, enter a name for the object, or accept the default.
2. In the text field, enter a message that appears on an IM client if the rule applies.
3. Select one of the following:
   - **Set message text**—Replaces the text displayed to the IM client. See the example in the screenshot.
   - **Append to message text**—The specified text is added to the IM message.

Volume 3: Web Communication Proxies provides more information about regulating IM through the SG appliance, as well as VPM examples.

Return ICAP Feedback

Specifies to display a patience page to the client or employ data trickling if ICAP scanning exceeds the given time duration.

To return ICAP feedback:



1. Name the object or accept the default.
2. Select interactive traffic (Web browser based requests) options:
   a. **Do not provide feedback...**: Users do not receive feedback for longer ICAP scans.
   b. **Provide feedback after <value> seconds**: Specifies how far into the scan to wait before providing feedback (patience page or data trickling) to the client.
      - The range for the patience page method is 5 to 65535.
      - The range for the trickling methods is 0 to 65535.
   c. Select a feedback method:
      - **Return patience page**: The SG appliance displays a (customizable) page on the Web browser client, informing the user a content scan is in progress.
      - **Trickle object data from start**: The more secure method because most of the object data does not reach the client, pending the result of the content scan. However, users might become impatient, close the request, and reinitiate the connection.
      - **Trickle object data at end**: The lesser secure method because the client receives most of the object data, pending the result of the content scan. This method provides the better user experience because they perceive the connection as almost complete.
3. Select non-interactive traffic (non-Web browser based clients, such as flash players or automatic updaters) options. See descriptions in Step 2.

4. Click **OK**.

Enter a time value (in seconds) that the SG appliance waits for content to be serviced from the origin content server before displaying the page that instructs users an ICAP scan is in progress.

Note: Patience pages display regardless of any pop up blocking policy that is in effect.

Patience page management and limitations are described in *Volume 7: Managing Content*.

Set Dynamic Categorization

Dynamic categorization extends the process of categorizing a URL. Traditional content filtering involves searching of massive URL pattern databases, which are published by vendors and downloaded to the SG appliance at specified intervals. As new content constantly reaches the Web, the limitation is that it cannot be filtered until its existence is discovered, added, and uploaded. Dynamic categorization enhances content filtering by scanning a new Web page, attempting to determine its contents, and categorizing accordingly in real time.

When an un-categorized page is first encountered, the SG appliance calls an external service with a categorization request. Once the content is scanned, a category is assigned (a majority of the time).

For related information, refer to the Content Filtering chapter in *Volume 7: Managing Content*.

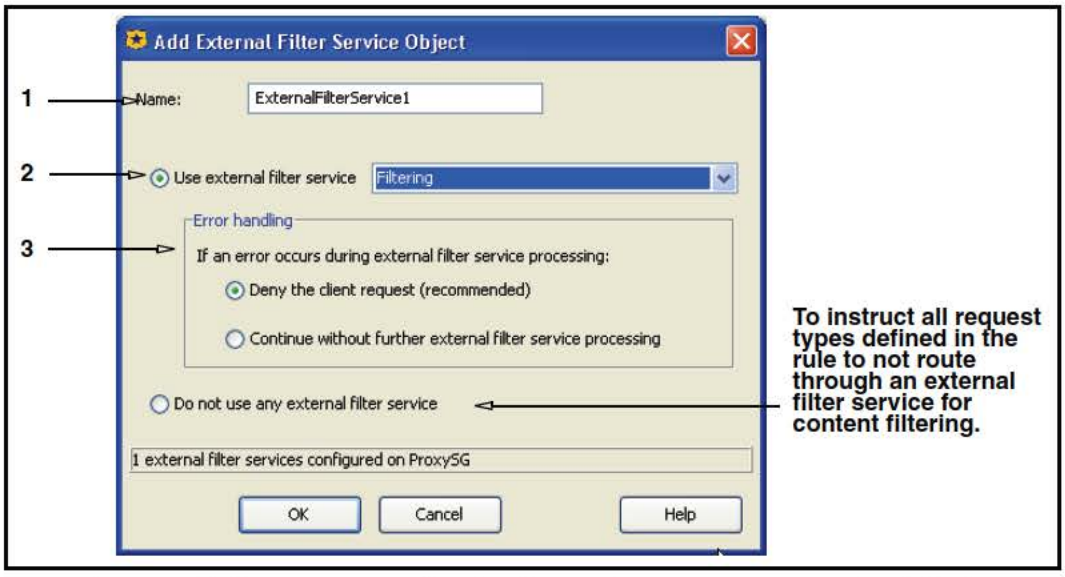
To configure dynamic categorization:

1. Select a mode:
   - **Do not categorize dynamically**—The loaded database is consulted for category information. URLs not in the database show up as category **none**.
   - **Categorize dynamically in the background**—Objects not categorized by the database are dynamically categorized as time permits. Proxy requests are not blocked while DRTR is consulted. Objects not found in the database appear as category **pending**, indicating that DRTR was requested, but the object was served before the DRTR response was available.
   - **Categorize dynamically in realtime**—The default. Objects not categorized by the database are dynamically categorized on first access. If this entails consulting the DRTR service, the proxy request is blocked until DRTR responds.
   - **Use dynamic categorizing setting from configuration**—Default to the SG appliance configuration (**Content Filtering>**Blue Coat**>Dynamic Categorization**).
2. Click **OK**.

Set External Filter Service

Specifies which installed content filtering service or service group a content request is subjected to or bypasses, and specifies what occurs if a communication error occurs between the SG appliance and the external service.

To determine external filter service request behavior:

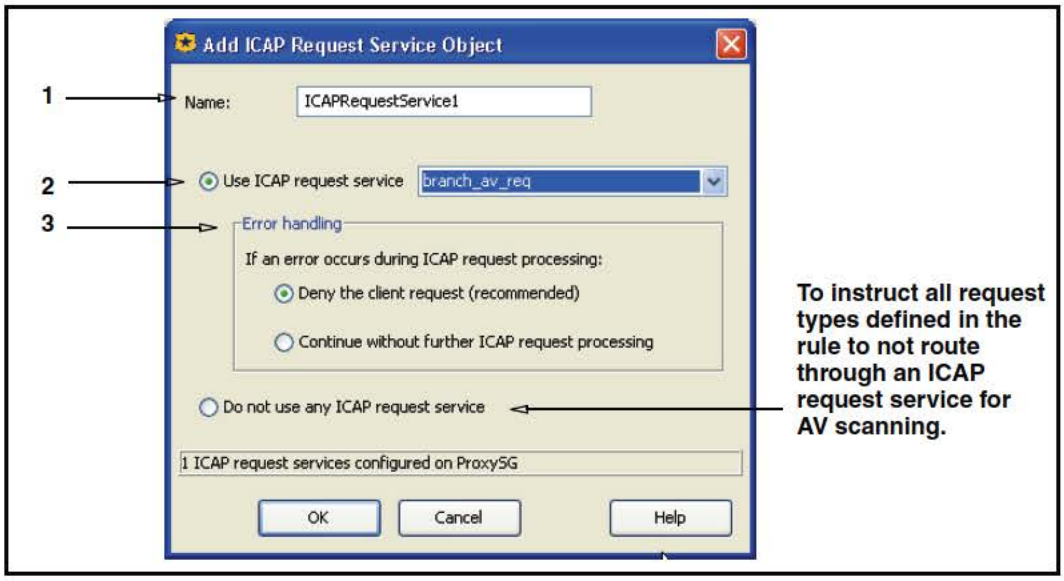


1. In the **Name** field, enter a name for the object or leave as is to accept the default.
2. To instruct all requests defined in the rule to route to a specific external filter service, select **Use External Filter Service**; from the drop-down list, select the external filter service or service group (which must already exist on the SG appliance; **Configuration > External Services**).
3. In the **Error handling** field, select one of the following option:
   - To deny all requests if a communication error occurs, select **Deny the client request**.
   - To allow requests to go through without content filtering, select **Continue without further external service processing**.
4. Click **OK**.

Set ICAP Request Service

Specifies which installed ICAP service or service group a content request routes to or bypasses, and specifies what occurs if a communication error occurs between the SG appliance and the ICAP server.

To determine ICAP request behavior:



1. In the **Name** field, enter a name for the object or leave as is to accept the default.
2. To instruct all request or response types defined in the rule to route to a specific ICAP service, select **Use ICAP Request Service**; from the drop-down list, select the ICAP request modification service or service group (which must already exist on the SG appliance; **Configuration > External Services > ICAP**).
3. In the **Error handling** field, select one of the following option:
   - To deny all requests or responses if a communication error occurs, select **Deny the client request**. This is the default and recommended by Blue Coat.
   - To allow requests or responses to go through without ICAP scanning, select **Continue without further ICAP request processing**. Be advised that this presents a content integrity risk.

Note: When the ICAP service is restored, these objects are scanned and served from the cache if they are requested again.

Set ICAP Response Service

Identical to "Set ICAP Request Service" on page 111, but applies to other protocol responses, such as HTTP and FTP. Requires an ICAP response modification service created on the SG appliance (**Configuration > External Services > ICAP**).

Set FTP Connection

For an outgoing request over FTP, specifies whether the FTP connection should be made immediately or deferred, if possible. The benefit of deferring connections is that requests for previously cached content can be served without contacting the origin server, which reduces the FTP load on that server.

Set SOCKS Acceleration

Specifies whether or not accelerate SOCKS requests, and defines the transport method.

To set SOCKS acceleration:

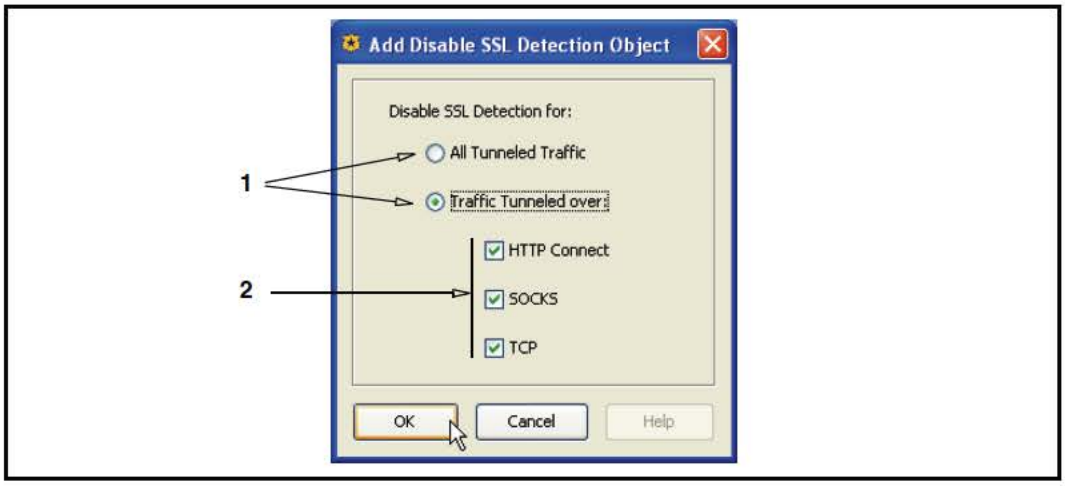
1. In the **Name** field, enter a name for the object or leave as is to accept the default.
2. Select one of the following:
   - **Automatically**—Accelerates SOCKS requests automatically, based on the destination port receiving the connection.
   - **Do Not Accelerate**—Never accelerate SOCKS requests matched by this rule.
   - **Accelerate via [HTTP | AOL IM | MSN IM | Yahoo IM]**—Specifies the type of acceleration applied to requests matched by this rule.
3. Click **OK**.

Disable SSL Detection

Important: This object is only required to preserve user-selectable SSL detection options that existed in SGOS 4.2.x but are not available in SGOS 5.2.x.

SGOS 4.2.x allowed you to select whether SSL was detected over HTTP, SOCKS, and/or TCP. These are options are not user-selectable in SGSO 5.2.x, but you can use this object to preserve the previous behavior.

To preserve 4.2.x SSL detection behavior:



1. Select one of the following:
   - If in SGOS 4.2.x you configured all proxies to not detect SSL, select **All Tunneled Traffic** and proceed to step 3.
   - If in SGOS 4.2.x you configured SSL detection for one or two proxies, select **Traffic Tunneled Over** and proceed to step 2.
2. Select one or more proxies.
3. Click **OK**.

For more information about this feature, refer to the *Blue Coat SGOS Upgrade/Downgrade Guide*.

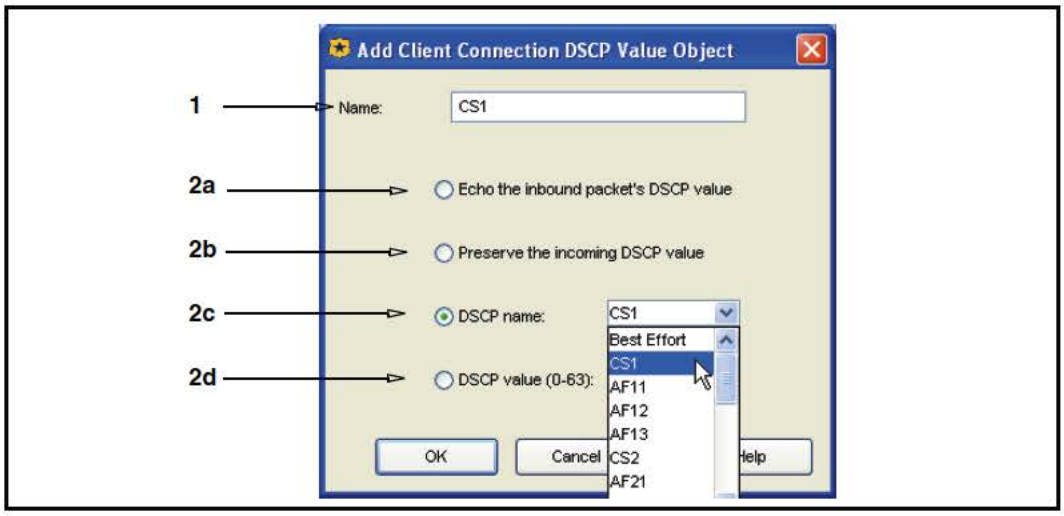
Set Streaming Max Bitrate

Specifies the maximum bitrate, in kilobits per second, of requested streaming media. If a request exceeds this rule, the request is denied.

Set Client Connection DSCP Value

Sets the outgoing differentiated service code point (DCSP) value or action for primary client connections (from the server) matching the DSCP value(s) in the **Source** column.

To set the server to client DSCP value or action:



1. In the **Name** field, enter a name for the object or leave as is to accept the default. This example sets the DSCP value to **CS1** (IP Precedence 1).
2. Selection an action:
   a. **Echo the inbound packet's DSCP value**: Use the same outbound (point of reference, the SG appliance) packet DSCP value as the inbound value.
   b. **Preserve the incoming DSCP value**: Track the inbound (from the client) DSCP bits on the *primary* server connection and use that same value when sending packets to outbound to the server. This is valuable for protocols that have multiple client/server connections. For example, FTP control and data connections. The values remain independent for each connection.
   c. **DSCP name**: Instead of the incoming DSCP, use the DSCP value selected from the drop-down list.
   d. **DSCP value**: Instead of the incoming DSCP value, use this non-categorized DSCP value (range is **0** to **63**).
3. Click **OK**.

For conceptual information about configuring the SG appliance to manipulate traffic based on type of service, refer to Chapter 4: Advanced Policy, "Managing QoS and Differential Services" on page 194.

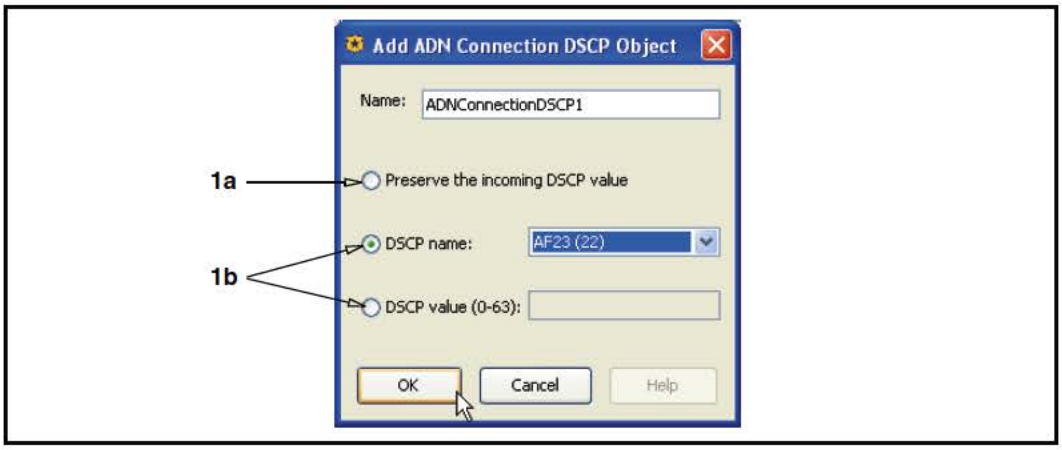
Set Server Connection DSCP Value

This object is identical to "Set Client Connection DSCP Value" on page 114, but applies to using the DSCP values or bits from client connections to server connections.

Set ADN Connection DSCP

This object specifies DSCP settings for Application Delivery Network (ADN) tunnel connections, which allows you more granular control to regulate WAN traffic. For example, you might not want the DSCP values for packets sent from the OCS and downstream tunnel packets to have the same value.

To specify an ADN connection DSCP value:



1. Select one of the following options:
   a. **Preserving the incoming DSCP value**: This is the default behavior if no other policy is specified. The ADN proxies (branch and concentrators) preserve the inbound packet DSCP values:
      - The client inbound packet and upstream tunnel packet DSCP values are the same.
      - The server inbound packet to the concentrator and downstream tunnel packet DSCP values are the same.
   b. From the **DSCP name** drop-down list, select one of the standard DSCP values. The behavior is as follows:
      - The DSCP value of the upstream tunnel packets is the selected value until it is reset by an intermediary device.
      - The DSCP value of a downstream packet is the selected value until it is reset by an intermediary device, even if the intermediary device modifies DSCP values of upstream tunnel packets.

      Alternately, if your network uses a numerical DSCP value system, select **DSCP value (0-63)** and enter a value.

> **Note:** For more information about DSCP values, see Chapter 4: Advanced Policy, Section F: "Managing QoS and Differential Services".

Click **OK**.

Set Authorization Refresh Time

Realms that support authorization and authentication separately use the authorization refresh time value to manage the load on the authorization server. These realms include: Local, LDAP, Windows SSO, Novell SSO, Certificate, XML and Policy Substitution. They determine authorization data (group membership, attribute values) separately from authentication, allowing the time the authorization data is trusted to be increased or decreased.

For realms that must authenticate the user to determine authorization data, the authorization data is updated only when the user credentials are verified with the authentication server.

Set Credential Refresh Time

The credential refresh time value determines how long a cache username and password is trusted. After that time has expired, new transactions that require credential authentication result in a request to the authentication server. A password different than the cached password also results in a request to the authentication server.

This value can is only valid for realms that cache the username and password on the proxy and realms that use Basic username and password credentials: LDAP, RADIUS, XML, IWA (with Basic credentials), SiteMinder, and COREid.

Set Surrogate Refresh Time

Specifies how long surrogate credentials are trusted in a particular realm.

Send DNS/RDNS Response Code

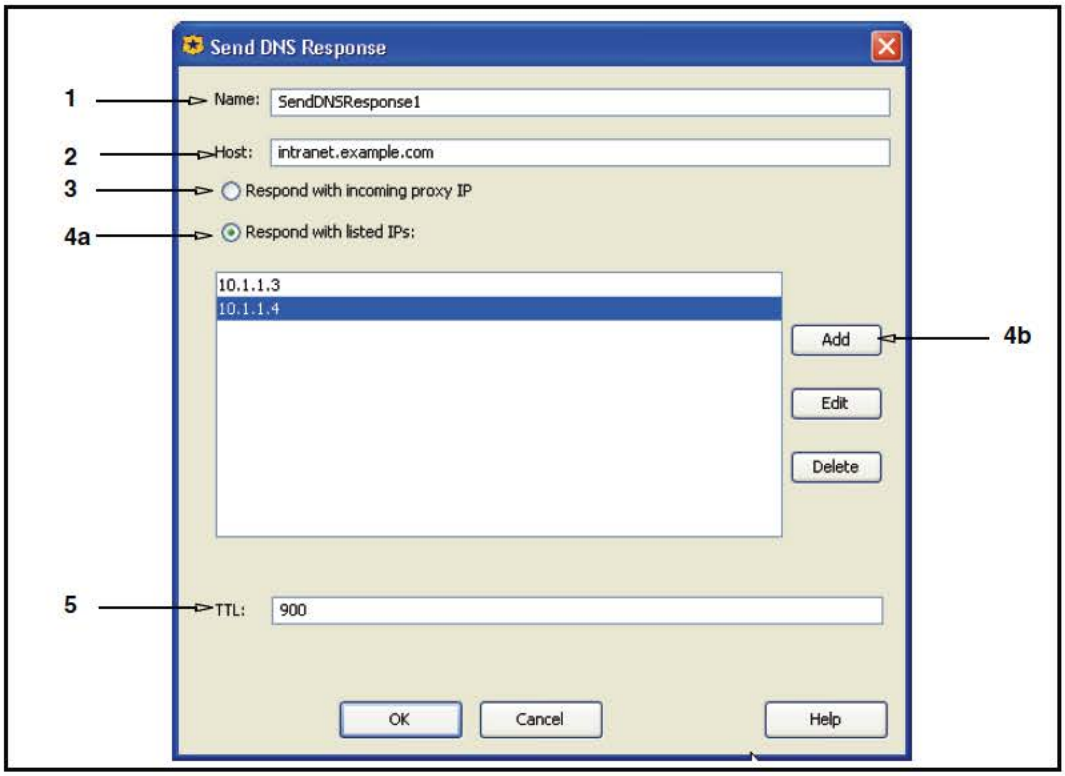
Specifies to send out the default response code or a selectable error response code. Perform one of the following:

- Select **Send Default DNS Response**; optionally, enter a TTL (time to live) value.
- Select **Send Error Response Code** and select a code from the drop-down list.

Send DNS Response

Specifies which IP address to return for a specified host.

To set a DNS response:



1. In the **Name** field, enter a name for the object or leave as is to accept the default.
2. In the **Host** field, enter a host name that is returned.
3. To respond with the IP address of the proxy that is forwarding the request, select **Respond with proxy IP**.
4. To respond with one or more IP addresses:
   a. Select **Respond with listed IPs**.
   b. Click **Add**. The Add DNS Response IP dialog appears.
   c. Enter an IP address and click **Add**.
   d. Repeat as required; click **Close**.
5. (Optional) In the **TTL** field, enter a time-to-live value (how long the response is cached).
6. Click **OK**.

Send Reverse DNS Response

Specifies which host to return for a reverse DNS response. Optional: define a time-to-live value.

Do Not Cache

This is a static object. Specifies that objects are never cached.

Force Cache

This is a static object. Specifies that (cacheable) objects are always cached. Objects that are not cacheable (for example, RealMedia file types) and supported in pass-through mode only are not cached.

Use Default Caching

This is a static object. Overrides the **Do Not Cache** and **Force Cache** actions and instructs the SG appliance to use its default determination of whether or not to cache the content.

Mark/Do Not Mark As Advertisement

These are static objects. Specifies content to be identified as an advertisement. The SG appliance still fetches content from the cache (if present); however, just after serving to the client, the content is re-fetched from the ad server so that hit counters are updated.

Enable/Disable Pipelining

These are static objects. Enables or disables the SG appliance pipelining feature, which, when enabled, examines Web pages for embedded objects and requests them from the origin server in anticipation of a client request.

Set TTL

Specifies the time-to-live (TTL) an object is stored in the SG appliance. In the **Name** field, enter a name for the object (or leave as is to accept the default); in the **TTL** field, enter the amount of time in seconds.

Send Direct

This is a static object. Overrides forwarding host, SOCKS gateway, or ICP configurations and instructs the SG appliance to request the content directly from the origin server.

Integrate/Do Not Integrate New Hosts

This is a static object. Used in server accelerator deployments. When enabled, the corresponding host that is accessed is added to the list of hosts for which the SG appliance performs health checks. If that host name resolves to multiple IP addresses that correspond to different servers, the SG appliance fetches content from the available servers and ignores the servers that fail the health check.

Allow Content From Origin Server

This is a static object. Allows request to access content from an origin server if the content is not cached.

Serve Content Only From Cache

This is a static object. Requests to access content that is not cached are denied. If the content is cached, the content is served.

Select SOCKS Gateway

Specifies which SOCKS gateway, if any, to use; defines behavior if communication between the SOCKS gateway and the SG appliance is down.

- To instruct the rule to connect directly without routing through a SOCKS service, select **Do not use SOCKS gateway**.
- To instruct the rule to connect through a SOCKS gateway, select **Use SOCKS Gateway** and select an installed SOCKS service from the drop-down list.

  In the **If no SOCKS gateway is available** field, select **Deny the request** or **Connect directly**, which allows requests to bypass the SOCKS service.

Select Forwarding

Specifies which forwarding host or group, if any, to use; defines behavior if communication between the forwarding and the SG appliance is down.

- To instruct the rule to connect directly without redirecting to a forwarding host or group, select **Do not forward**.
- To instruct the rule to redirect to a forwarding host, select **Use Forwarding** and select an installed forwarding host from the drop-down list.

  In the **If no forwarding is available** field, select **Deny the request (fail closed)** or **Connect directly (fail open)**, which allows requests to bypass the forwarding host.
- To instruct the rule to forward using the ICP configuration, select **Forward using ICP**.

Server Byte Caching

Specifies whether byte caching is employed on either (branch or core) or both sides of an Application Delivery Network connection (specified IP addresses in the rule). Byte caching reduces WAN latency.

- **Optimize traffic in both directions**: Apply Byte Caching to traffic coming and leaving the server.
- **Optimize only inbound traffic**: Apply Byte Caching only to traffic coming into the server.
- **Optimize only outbound traffic**: Apply Byte Caching only to traffic leaving the server.
- **Do not optimize traffic**: Do not allow Byte Caching on specified connections.

Set IM Transport

Specifies the transport method used for IM traffic.

- **Auto**—Connects using the transport method used by the client.
- **HTTP**—Tunnels the IM requests over HTTP.
- **Native**—Connects using the native transport used by the service.

Set Streaming Transport

Specifies which streaming transport method the rule uses.

- **Auto**—Connects using the transport method used by the client.
- **HTTP**—Streaming over HTTP.
- **TCP**—Streaming over TCP.

Authentication Charset

The VPM allows you enter non-ASCII in many objects, such user and group names and text for the "Notify User" on page 101 object. This object allows you set the character set to use in conjunction with localized policy. From the drop-down list, select a character set and click **OK**.

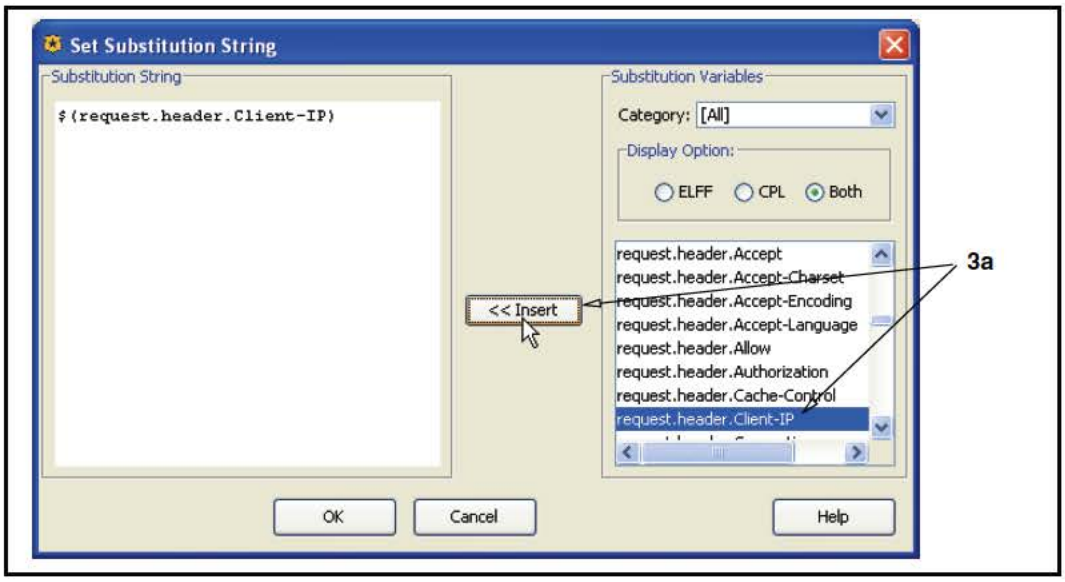
Set IP Address For Authentication

Some Application Delivery Network (ADN) configurations in proxy chain deployments mask the source IP address of the request. Policy to set the IP address for authentication is required so that Windows Single Sign On (SSO), Novell SSO, and policy substitution realms can authenticate users.

For more information, see *Volume 4: Securing the Blue Coat SG Appliance* for more information about this type of authentication.

To set an IP address for authentication:

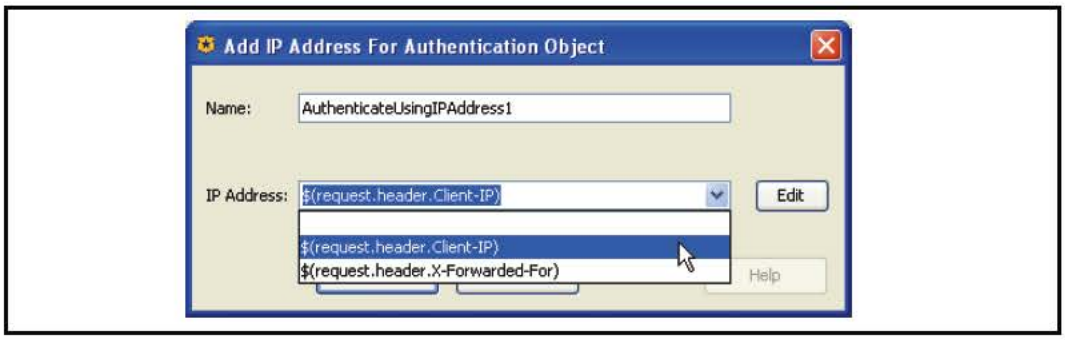
1. Define a name for the object or accept the default.
2. Click **Edit** to display the Set Substitution dialog.



3. Define the substitution strings:

a. Select one or more strings and click **Insert**. For example, your branch user headers contain the `request.header.ClientIP` HTTP header.

b. Click **OK**.



4. From the **IP Address** drop-down list, select a substitution string. For example: the **$(request.header.Client-IP)**: sets the address for authentication to the address received from the HTTP `Client-IP` header.

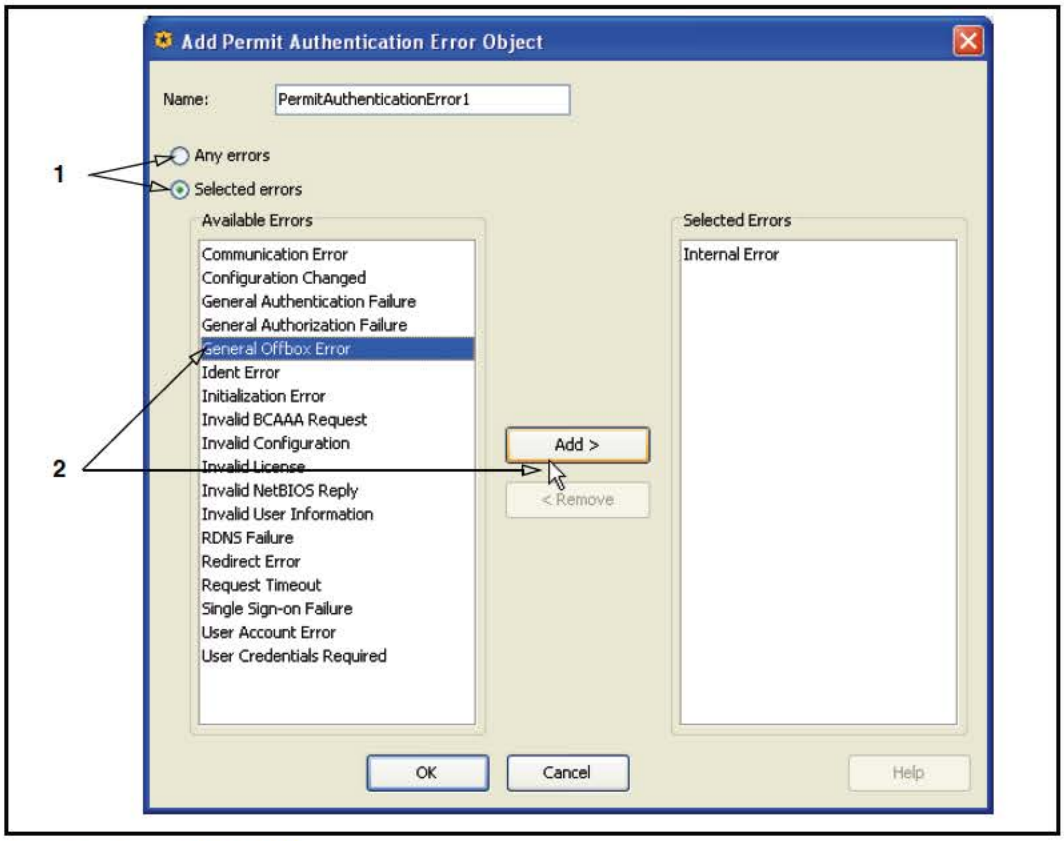
5. Click **OK**.

Permit Authentication Error

After an authentication failure occurs, the authentication error is checked against the list of errors that policy specifies as permitted:

- If the error is not on the list, the transaction terminates.
- If the error is on the list, the transaction is allowed to proceed; however, the user is unauthenticated. Because the transaction is not considered authenticated, the `authenticated=yes` policy condition evaluates to false and the user has no username, group information, or surrogate credentials. Policy that uses the user, group, domain, or attribute conditions does not match.

To permit an authentication error:



1. Select one of the following:
   - **Any errors**: Allows any type of authentication error.
   - **Selected errors**: Only allowed if the error matches the selected errors.
2. If you selected **Selected errors**:
   a. Select one or more error types (use Control + Left-click to highlight multiple errors).
   b. Click **Add** to move the errors to the **Selected** field.
   c. Name the object or accept the default name.
3. Click **OK**.

Permit Authorization Error

After an authorization failure occurs, the authorization error is checked against the list of errors that policy specifies as permitted.

- If the error is not on the list, the transaction is terminated.
- If the error is on the list, the transaction is allowed to proceed and the user is marked as not having authorization data.
- If a user is successfully authenticated but does not have authorization data, the `authenticated=yes` condition evaluates to true and the user has valid authentication credentials.

- The `user.authorization_error=any` evaluates to true if user authorization failed and the user object contains username and domain information, but not group or attribute information. As a result, policy using user or domain actions still match, but policy using group or attribute conditions do not.

Combined Action Objects

Allows you to combine an action object that invokes multiple actions. See "Using Combined Objects" on page 128.

Action Column/Policy Layer Matrix

The following matrix lists all of the **Action** column objects and indicates which policy layer they apply to.

Object	Admin Auth	Admin Acc	DNS Acc	SOCKS Auth	SSL Int	SSL Acc	Web Auth	Web Acc	Web Cont	Fwding
Allow						x		x		
Deny (static)	x	x					x	x		
Allow Read-Only Access		x								
Allow Read-Write Access		x								
Do Not Authenticate	x			x			x			
Authenticate	x			x			x			
Force Authenticate	x			x			x			
Bypass Cache								x		
Do Not Bypass Cache								x		
Check Authorization								x	x	
Do Not Check Authorization								x	x	
Always Verify								x	x	
Use Default Verification								x	x	
Block Up Ads								x		
Do Not Block PopUp Ads								x		
Force IWA For Server Auth								x		
Do Not Force IWA For Server Auth								x		
Require Client Certificate						x				
Do Not Require Client Certificate						x				
Reflect IM Messages								x		
Do Not Reflect IM Messages								x		
Block IM Encryption								x		

Section C: Detailed Object Column Reference

Object	Admin Auth	Admin Acc	DNS Acc	SOCKS Auth	SSL Int	SSL Acc	Web Auth	Web Acc	Web Cont	Fwding
Do Not Block IM Encryption								x		
Trust Destination IP								x		
Not Trust Destination IP										x
Deny						x		x		
Return Exception						x		x		
Return Redirect								x		
Set Client Certificate Validation						x				
Set Server Certificate Validation						x				
Set HTTPS Intercept					x					
Set HTTPS Intercept on Exception					x					
Send IM Alert								x		
Modify Access Logging								x	x	
Override Access Log Field								x	x	
Rewrite Host								x		
Reflect IP			x					x		
Suppress Header								x		
Control Request Header								x		
Control Response Header								x		
Notify User								x		
Strip Active Content								x		
Set Client HTTP Compression								x		
ADN Server Optimization										x
Set Server HTTP Compression								x		
Modify IM Message								x		
Return ICAP Feedback								x		
Set Dynamic Categorization									x	
Set External Filter Service								x		
Set ICAP Request Service								x	x	
Set ICAP Response Service									x	
Use Default Caching									x	
Set FTP Connection								x		

Section C: Detailed Object Column Reference

Object	Admin Auth	Admin Acc	DNS Acc	SOCKS Auth	SSL Int	SSL Acc	Web Auth	Web Acc	Web Cont	Fwding
Set SOCKS Acceleration								x		
Set Streaming Max Bitrate								x		
Client Connection DSCP Value			x					x		
Server Connection DSCP Value			x					x	x	x
Send DNS/RDNS Response Code			x							
Send DNS Response			x							
Send Reverse DNS Response			x							
Do Not Cache									x	
Force Cache									x	
Mark As Advertisement									x	
Do Not Mark as Advertisement									x	
Enable Pipelining									x	
Disable Pipelining									x	
Set TTL									x	
Send Direct										x
Integrate New Hosts										x
Do Not Integrate New Hosts										x
Allow Content From Origin Server										x
Serve Content Only From Cache										x
Select SOCKS Gateway										x
Select Forwarding										x
Reflect IP										x
Set IM Transport										x
Set Streaming Transport										x
Authentication Charset							x			
Combined Objects			x		x	x		x	x	x

Track Object Column Reference

A *track* object defines the parameters for tracking and tracing traffic. All policy layers contain the same trace objects, but tracking parameters are layer-specific.

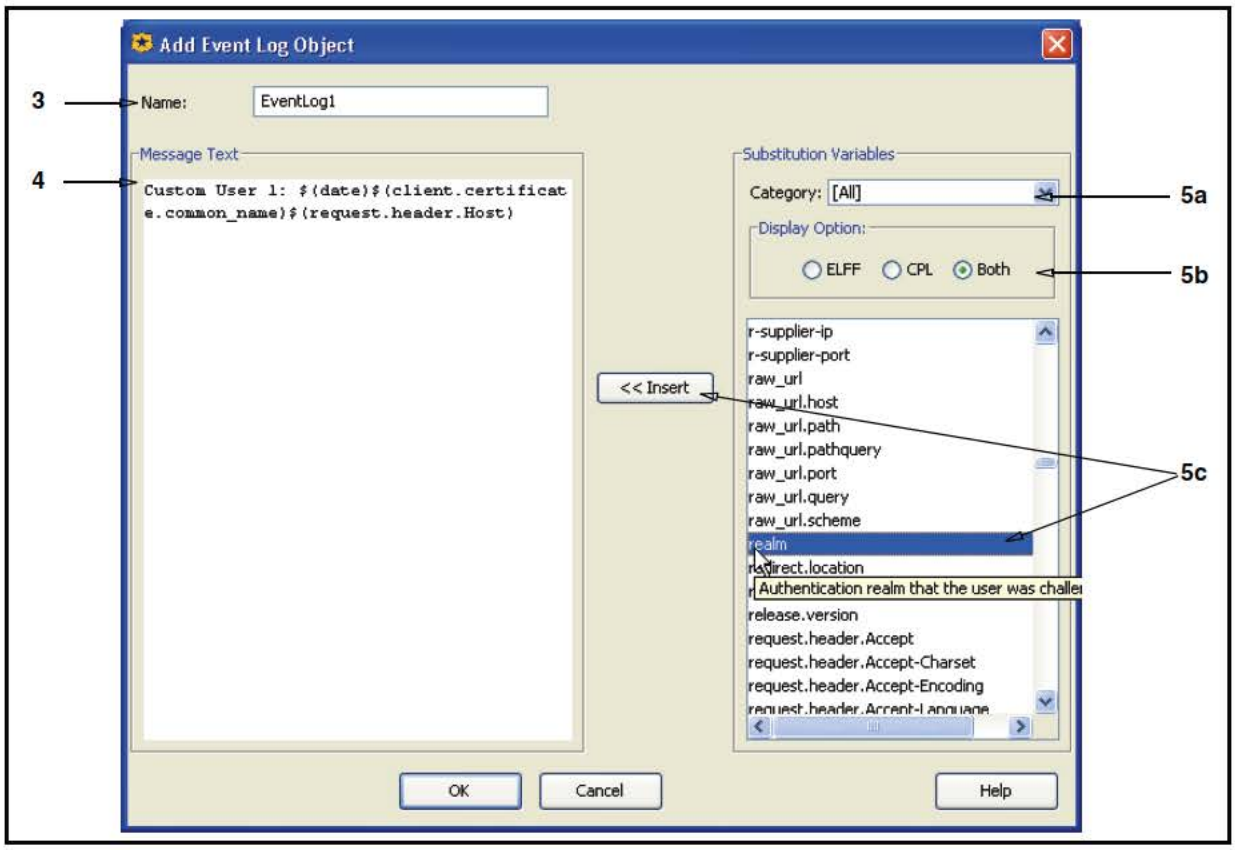
Note: Because of character limitations required by the generated CPL, only alphanumeric, underscore, and dash characters can be used to define an action object name.

Event Log, E-mail, and SNMP

You can customize the event log, E-mail notification, and SNMP with triggers. These triggers are the same for all three object types.

To customize an Event Log, E-mail, or SNMP object:

1. Right-click the **Tracking** cell in a policy layer and select **Set**; the Set Track Object dialog appears.
2. Click **New** and select **Event Log**, **Email**, or **SNMP**; the appropriate add object dialog appears.



3. In the **Name** field, enter a name for this object or leave as is to accept the default.

Note: The e-mail object also contains a **Subject** field.

4. In the **Message Text** field, enter a customized message that appears with each entry.

5. Optional: Add substitution variables. The substitution variables instruct the SG appliance to append specific information to the tracking object. The variables are categorized alphabetically, according to prefix.

   Note: Some variables do not have prefixes.

   In the **Substitution Variables** field:

   a. From the **Category** drop-down list, select a category to narrow the view to a subset of variables.
   b. The Display Option options allow you to further aggregate the variables by **ELFF** (Extended Log File Format) or **CPL** (Content Policy Language).
   c. Select a variable and click **Insert**. Rolling the mouse over a variable displays a brief description of the variable. Repeat as required.

Tracing Objects

This object specifies rule and Web traffic tracing.

Click **Trace Level** and select one of the following trace options:

- **No Tracing**—The default.
- **Request Tracing**—Generates trace output for the current request. The trace output contains request parameters (such as URL and client address), the final values of property settings, and descriptions of all actions taken.
- **Rule and Request**—Generates trace output that displays each rule that was executed
- **Verbose Tracing**—Generates the same output as **Rule and Request**, but also lists which rules were skipped because one or more of their conditions were false, and displays the specific condition in the rule that was false.

Furthermore, a trace destination can be entered that specifies the destination for any trace produced by the current transaction. To specify a destination path, select **Trace File** and enter a path in the field. For example, `abc.html`.

If a trace destination is configured in multiple layers, the actual trace destination value displayed is the one specified in the last layer that had a rule evaluated (which has a destination property configured). Consider the following multiple Web Access Layer example, demonstrated by the generated CPL:

```
<Proxy>
  url.domain=aol.com trace.request(yes) trace.rules(all)
  trace.destination("aol_tracing.html")
  url.domain=msn.com trace.request(yes)
  trace.rules(all)trace.destination("msn_tracing.html")
<Proxy>
  client.address=10.10.10.1 trace.request(yes) trace.rules(all)
```

The resulting actions are:

- Requests to the `aol.com` domain are logged to `aol_tracing.html`.
- Requests to the `msn.com` domain are logged to `msn_tracing.html`.
- Requests from the client with the IP of `10.10.10.1` are logged to the default location of `default.html`.

Note: After using a trace to troubleshoot, remove the trace to save log space.

The **Trace File** option can be used in conjunction or separately from the **Trace Level** option.

The default path of the trace file is accessible through one of the following URLs.

If the Management Console secure mode is enabled (the default on a new or upgraded system):

`https://SG_appliance_IP_address:8082/Policy/Trace/default_trace.html`

If the Management Console is deployed in non-secure mode:

`http://SG_appliance_address:8081/Policy/Trace/default_trace.html`

Combined Track Object

Allows you to combine track objects into one. See "Using Combined Objects" on page 128.

Track Objects/Policy Layer Matrix

The following matrix lists all of the **Track and** column objects and indicates which policy layer they apply to.

Object	Admin Auth	Admin Acc	DNS Acc	SOCKS Auth	SSL Int	SSL Acc	Web Auth	Web Acc	Web Cont	Fwding
Event Log		x	x		x	x		x	x	
Email Log		x	x		x	x		x	x	
SNMP Objects		x	x		x	x		x	x	
Trace	x	x	x	x	x	x	x	x	x	x
Combined Objects		x	x		x	x		x	x	

Comment Object Reference

The Comment object allows you to write any text to aid in labeling the policy layer. The text in this field does not impact the policy.

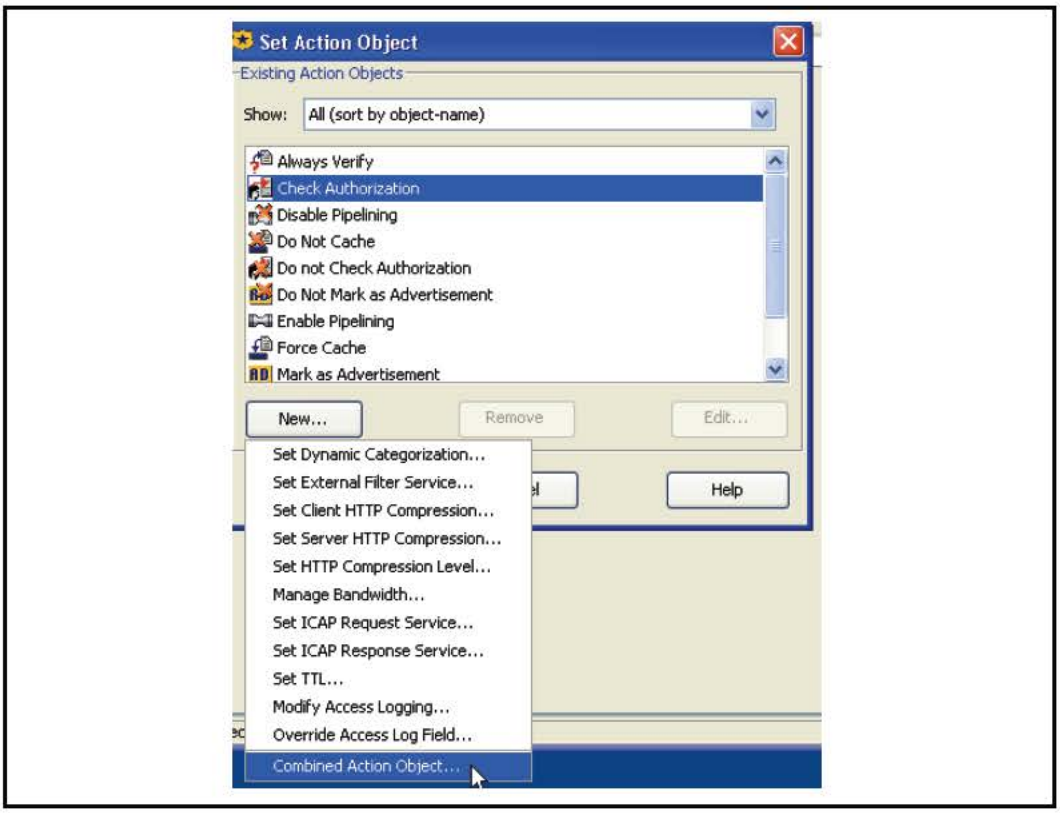
Using Combined Objects

As previously discussed, you select one object for as many object types as required for a given rule. Most object types also have the option of using a combined object. This feature allows you to select multiple objects for a given type, thus creating more complex tools. There are two uses for combined conditions: lists and multiple object types. Also consider the **Negate** option, which exempts the objects in the list.

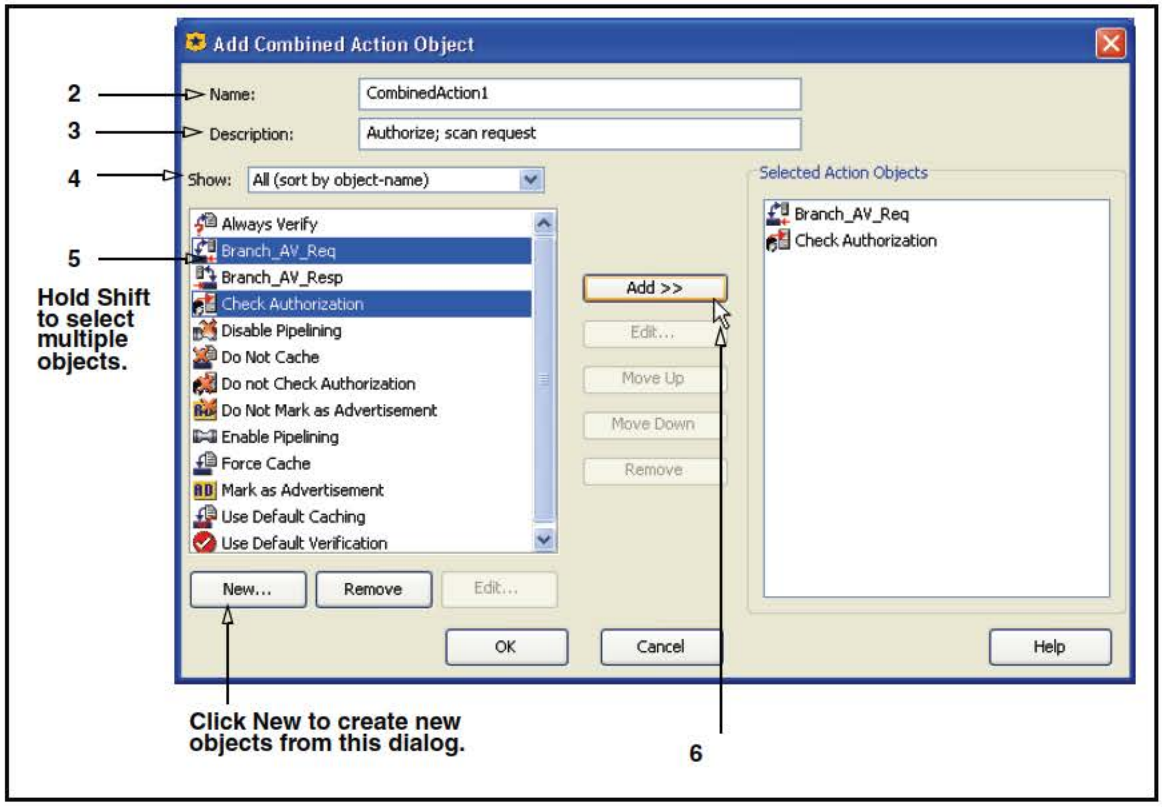
Example One

Consider the following example. You want a Web Content policy layer that as an action forces authorization *and* sends the response to an ICAP service for content scanning.

Section C: Detailed Object Column Reference



1. In the Set Action Object dialog, select **New > Combined Action Object**.

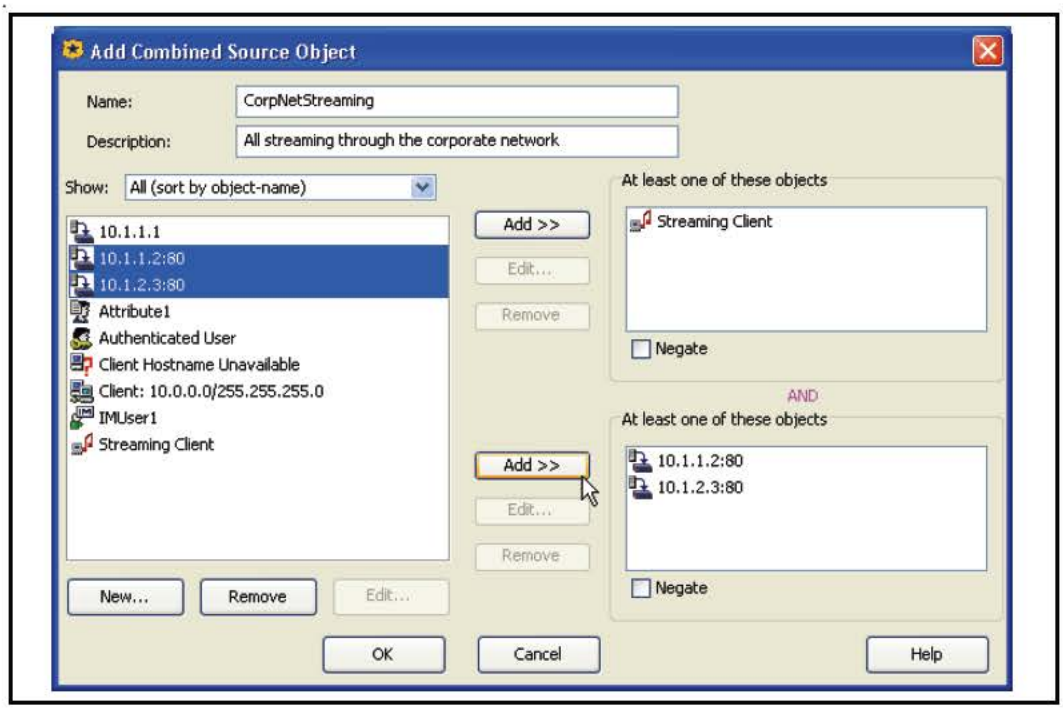


2. In the **Name** field, enter a name for this object or leave as is to accept the default.
3. In the Description field, enter brief text that explains the intent of this object (for reference).
4. The Show drop-down list allows you narrow the scope of the displayed objects.
5. Hold the Shift key and select **Check Authorization** and **Branch_AV_Req**.
6. Click **Add**. The selected objects appear in the **Selected Action Objects** field.
7. Click **OK**. The **CombinedAction1** object appears as a separate, selectable object.
8. Select **CombinedAction1**; click **OK**. The object is now part of the rule.

   Based on the other parameters specified in the rule, all requests are forced to an upstream server for authorization and the Web responses are subject to content scanning through the ICAP service.

Example Two

In the following example, the rule searches for one of the **Proxy IP Address/Port** objects and one of the streaming client user agents.



Note

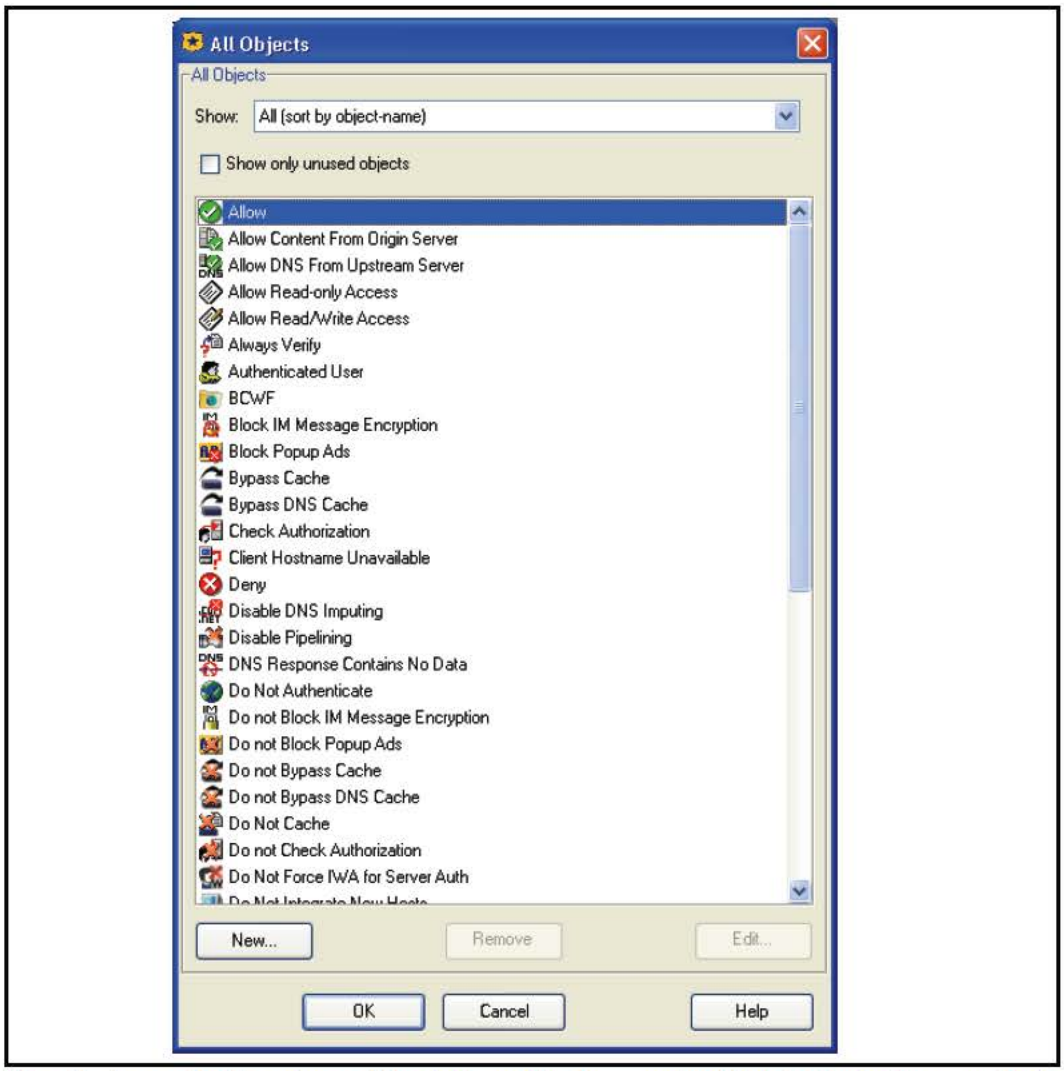
The VPM displays various warning messages if you attempt to add objects that creates an invalid combined object. However, it is possible to add a combined object to another combined object, even if doing so presents duplication of simple object definitions without receiving validation warnings. For example, the contents of a child combined object might have already been included either within the parent combined object directly, or indirectly within other child combined objects. This is allowable because of the complexity some combined objects and policies can achieve.

Centralized Object Viewing and Managing

This section describes how to use the All Objects dialog to view and manage every VPM object.

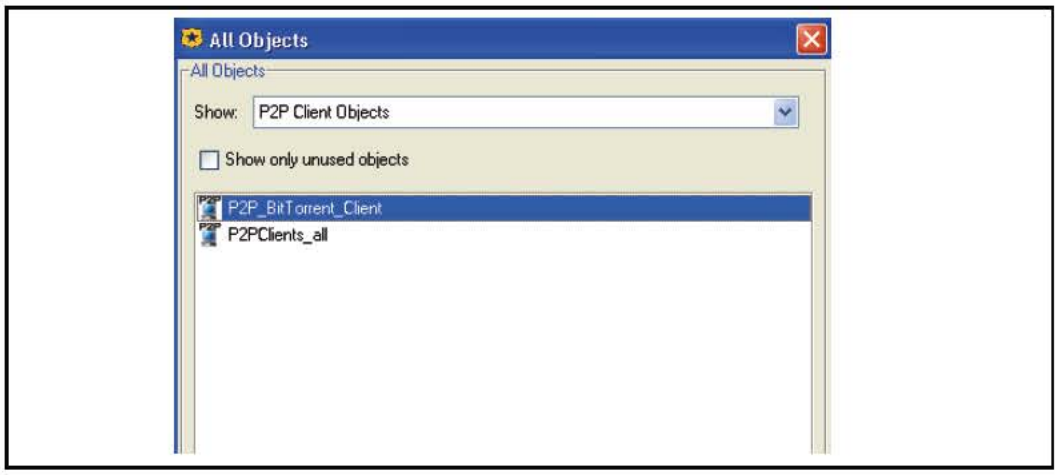
Viewing Objects

The All Objects feature allows you view a list of all objects—both static and user-defined—that currently exist across all layers and columns. To view all configured VPM objects, in the Menu Bar select **View > All Objects**.



The objects are displayed according to the policy layer order (click **Policy** in the menu bar) and the column order (as presented in "Policy Layer and Rule Object Reference" on page 38). To narrow the scope of the displayed objects, select from the **Show** drop-down list at the top:

- **All (sort by object name)**: Displays all objects in alphabetical order.
- **All (sort by object type)**: Groups object types together.
- You can select to display only the static (predefined) objects for the **Source**, **Destination**, **Service**, and **Action** columns.
- You can select to display or any one object type. For example, you want to only view the user-defined **P2P Client** objects. Scroll down and select **P2P Client Objects**.



View Unused Objects

Selecting **Show only unused objects** displays all static and user-defined objects that are not currently used in any policy layer.

Managing Objects

This section describes how to manage objects within the All Objects dialog.

Creating Objects

The All Objects dialog also allows you to create objects. Once an object is created, it appears in the list. When creating or editing policy layers, the objects are available to add to rules.

To create an object:

1. Select **New**. The available columns and relevant objects are displayed in a cascade style.
2. Select **Column > Object**. The Add dialog for that object appears.
3. Define the object as required
4. Click **OK**.

Note: When creating Combined Objects, not all objects that appear in the left column are valid for more than one policy layer type. For example, the **IM User** object is only valid in the **Web Access Layer > Source** column. If you attempt to add an object that is not valid, a dialog appears with that information.

Editing Objects

Any user-defined object can be modified. Highlight the object and click **Edit**. After editing the object, re-install the policy to apply the modified object in every policy layer it exists in.

Deleting Objects

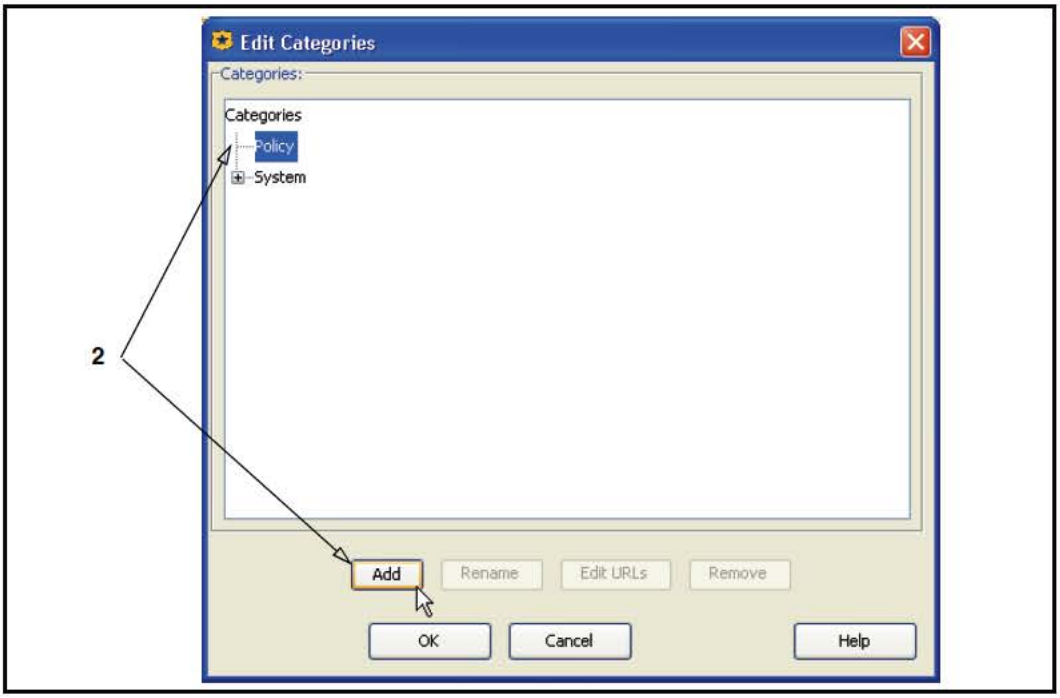
You cannot delete an object that is currently part of an installed policy or combined object. Before removing an object, you can use the **View>Object Occurrences** feature to identify which policy layers contain the object.

Creating Categories

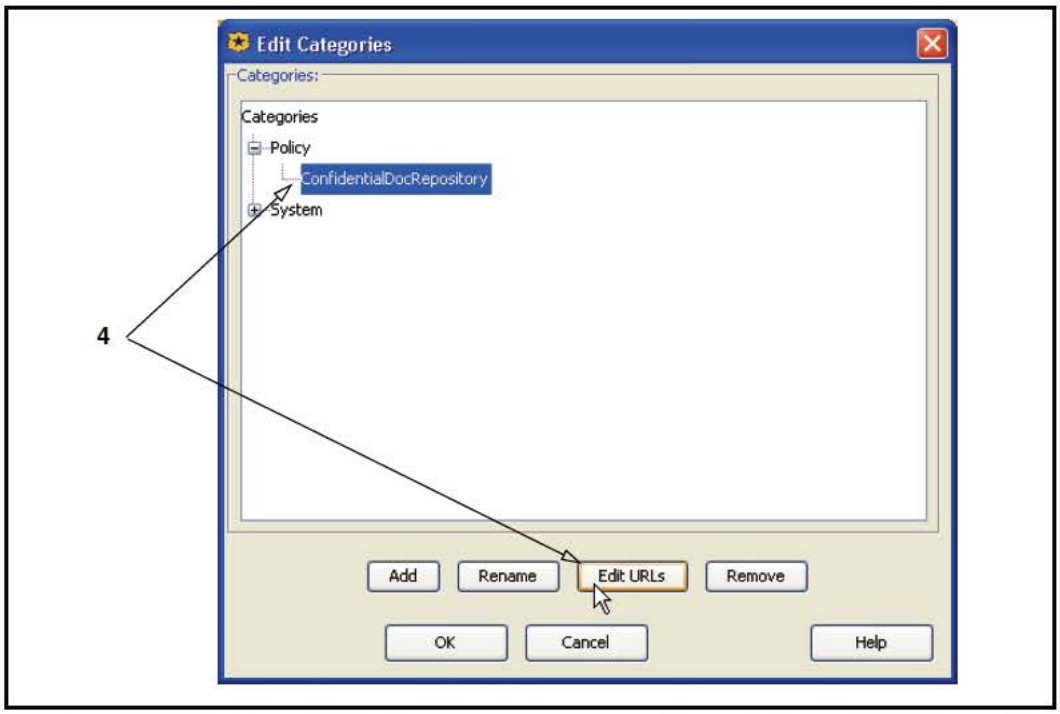
This feature allows you create the content filter URL categories that can be used in the **Category** object. The **Destination** column in the **DNS Access**, **Web Access**, **Web Authentication**, and **Web Content** policy layers contain the **Category** object. Similarly, categories created in the **Category** object (see "Request URL Category" on page 68) appear in this dialog and can be edited.

Create a category

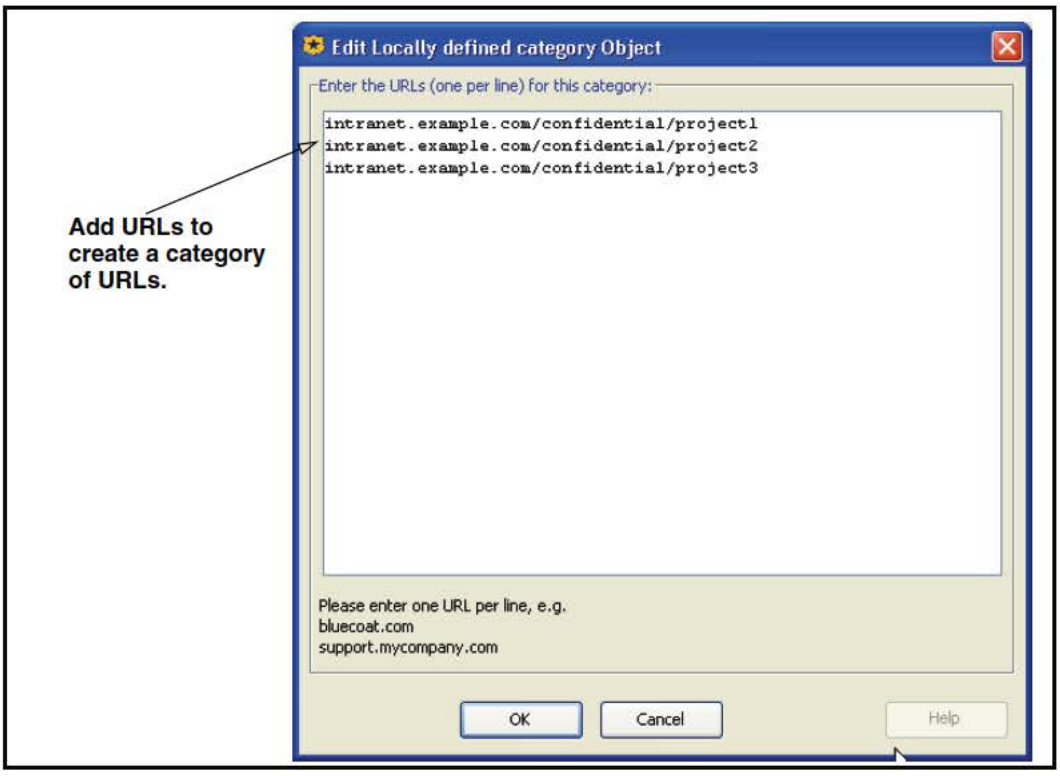
1. In VPM, select **Configuration > Edit Categories**. The Edit Categories dialog appears.



2. Select **Policy**; click **Add**. The Object Name dialog appears.
3. Name the category and click **OK**.



4. Drop the **Policy** list and select the created category; click **Edit URLs**. The Edit Locally Defined Category Object dialog appears.



5. Enter URLs appropriate for the content filter category you are creating; click **OK**.
6. Click **OK** in the Edit Categories dialog to complete the category creation.

Note: If other administrators have access to the SG appliance through other workstations and are creating categories either through VPM or with `inline` commands, consider that newly-created or edited categories are not synchronized until the policy is installed. When the policy is installed with VPM, the categories are refreshed. If too many categories are created at the same time and confusion occurs, select the **File > Revert to Existing Policy on** SG **Appliance** option to restore the policy to the previous state and reconfigure categories.

Refreshing Policy

In between occurrences when either VPM is closed and reopened or **Install Policies** is invoked, VPM does not recognize changes to VPM-managed policy that were made on the SG appliance through another method. For example:

- Another administrator opens a separate VPM to make changes.
- Another administrator edits the local or central policy file through the serial console.
- Another administrator makes edits the local or central policy file.
- A new content filter database is downloaded automatically and the new update contains category changes.
- A new content filter database is downloaded manually by an administrator.

Restricting DNS Lookups

This section discusses DNS lookup restrictions and describes how to create a list.

About DNS Lookup Restriction

The DNS lookup restriction list is a list of domain names that apply globally, regardless of policy layer definitions. Once a domain name is added to the list, DNS lookup requests do not occur for that domain name while policy is evaluated. For more detailed information about using DNS lookups, refer to *Volume 10: Blue Coat SG Appliance Content Policy Language Guide*.

Creating the DNS Lookup Restriction List

The list is created from the VPM Menu bar.

To create the DNS lookup restriction list:

1. Select **Configuration > Set DNS Lookup Restrictions**; the Set DNS lookup restrictions dialog appears.

   The default is **None**; no domain names are restricted.
2. To restrict every domain name, select **All**.
3. To add specific domain names, perform the following steps.

a. Select **Listed Host Patterns**. This enables the **Host Patterns** field.
b. Click **Add**; the Add Host Pattern dialog appears.
c. Enter a domain name; click **OK**.
d. Repeat to add other domain names.
e. Click **OK**.

Restricting Reverse DNS Lookups

This section discusses reverse DNS lookup restrictions and describes how to create a list.

About Reverse DNS Lookup Restriction

The Reverse DNS lookup restriction list is a list of subnets that apply globally, regardless of policy layer definitions. Once a subnet is added to the list, the SG appliance will not perform a reverse lookup of addresses on that subnet during policy evaluation. For more detailed information about using reverse DNS lookups, refer to *Volume 10: Blue Coat SG Appliance Content Policy Language Guide*.

Creating the Reverse DNS Lookup Restriction List

The list is created from the VPM Menu bar. This prevents the SG appliance from performing reverse DNS lookups of addresses in the list while evaluating policy.

To create the reverse DNS lookup restriction list:

1. Select **Configuration > Set Reverse DNS Lookup Restrictions**; the Set Reverse DNS lookup restrictions dialog appears.

   The default is **None**; no subnets are restricted.
2. To restrict every subnet, select **All**.
3. To add specific subnets, perform the following steps.
   a. Select **Listed Subnets**.

      This enables the **Subnets** field.
   b. Click **Add**; the Add Subnet dialog appears.
   c. Enter a subnet; click **OK**.
   d. Repeat to add other subnets.
   e. Click **OK**.

Setting the Group Log Order

This section discusses the group log order and describes how to create a list.

About the Group Log Order

The Group Log Order object allows you to establish the order group data appears in the access logs. For more detailed information about using group log ordering, refer to *Volume 10: Blue Coat SG Appliance Content Policy Language Guide*.

Creating the Group Log Order List

The list is created from the VPM Menu bar.

To create the group log order list:

1. Select **Configuration > Set Group Log Order**; the Set Group Log Order dialog appears.
2. Click **Add**; the Add Group Object dialog appears.
3. In the **Group Name** field, enter the name of a group.

   The group must be already configured on the SG appliance.
4. From the **Authentication Realm** drop-down list, select a realm.
5. Click **OK**.
6. Repeat as required to add more groups.
7. To order the list, select a group and click **Move Up** or **Move Down** until you achieve the desired order.
8. Click **OK**.

Section D: Managing Policy Layers, Rules, and Files

This section contains the following topics:

- "How Policy Layers, Rules, and Files Interact" —Describes the importance of rule order policy layer order.
- "Managing Policy" —Describes how to save and install policies on the SG appliance.
- "Installing VPM-Created Policy Files" —Describes how to propagate a policy file created on one SG appliance to another.
- "Viewing the Policy/Created CPL" —Describes how to view the underlying CPL that is created with VPM.

How Policy Layers, Rules, and Files Interact

The following critical points discuss the behaviors and priorities of policy rules, layers, and files:

- Rules in different policy layers of the same type work together, and the order of policy layers is important.
- The order of policy layers of different types is important.
- The order of rules in a policy layer is important.
- Policy created in VPM is saved in a file on the SG appliance; the state of the VPM user interface is also stored as an XML file on the SG appliance.

Note: These files are stored *only* if the policy is installed without any errors.

- How the appliance evaluates those rules in relation to policy layers that exist in the central and local policy files is important. For more information, see Chapter 2: "Managing Policy Files" on page 15.

How VPM Layers Relate to CPL Layers

VPM generates CPL in various layers, but the concept of layers presented in VPM is slightly different. VPM provides policy layers for special purposes. For example, Web Authentication and Web Authorization, which both generate CPL `<Proxy>` layers. This minimizes timing conflicts by restricting the choices of conditions and properties to those compatible timing requirements. The following table summarizes how to use VPM layers and which CPL layers result.

Table 3-4. VPM-Generated CPL Layers

Policy Purpose	VPM Layer	CPL Layer
Establish Administrator identities.	Admin Authentication	`<Admin>`
Control Administrator access.	Admin Authorization	`<Admin>`
Control DNS access.	DNS Access	`<DNS>`
Establish SOCKS user identities.	SOCKS Authentication	`<Proxy>`

Table 3-4. VPM-Generated CPL Layers (Continued)

Policy Purpose	VPM Layer	CPL Layer
Allow HTTPS interception.	SSL Intercept	<SSL-Intercept>
Control HTTPS traffic.	SSL Access	<SSL>
Establish user identities.	Web Authentication	<Proxy>
Control user access.	Web Access	<Proxy>
Control content independent of users.	Web Content	<Cache>
Control forwarding.	Forwarding	<Forward>

Note: VPM currently does not support the <Exception> layer.

Ordering Rules in a Policy Layer

The SG appliance evaluates the rules in the order in which they are listed in a policy layer. When it finds a rule that applies to the situation, it skips the remaining rules in the policy layer and goes on to the next policy layer.

Consider the following simple example. Assume that a company has a policy that prohibits everyone from accessing the Web. This is a policy that is easy to create with a Web Access layer rule.

There are, however, likely to be exceptions to such a broad policy. For example, you require the manager of the purchasing department to be able to access the Web sites of suppliers. Members of the sales department need to access their customer Web sites. Creating Web Access rules for both these situations is also simple. But if you put all these rules in a single policy layer, then the rule prohibiting access to everyone must be ordered last, or the other two rules are not applied.

Principle Design Rule:

Always go from the specific to the general.

Using Policy Layers of the Same Type

Because the SG appliance skips the remaining rules in a policy layer as soon as it finds one that meets the condition, multiple policy layers and a combination of rules might be required to accomplish a task.

Consider the following example. A company does not want to prohibit its employees from accessing the Web, but it does not want them to abuse the privilege. To this end, the company wants employees who access the Web to authenticate when they do so; that is, enter a username and password. So the company creates a Web Authentication policy layer with a rule that says: "If anyone from anywhere in the company sends a request to a URL on the Web, authenticate the client before granting access."

The company also allows members of the group Sales to access various sports Web sites only during non-work hours. Given the Web Authentication rule above, these people must authenticate when they do this. But the company feels that it is not important for people going to these sites after hours to authenticate. So the company creates the following Web Access policy-layer rule:

- Grant Sales personnel access to sports Web sites from 5:00 PM to midnight.

  But there are additional issues. Some members of the sales department spend a lot of time watching game highlights on video clips, and this takes up a lot of bandwidth. At the same time, a lot of customers access the company Web site in the evening (during non-work hours), so internal bandwidth should remain manageable. The company, therefore, limits the bandwidth available to the people in the Sales department with a Web Access layer rule that is identical to the one above in all respects except for the action:

- Grant Sales personnel access to sports Web sites from 5:00 PM to midnight, but limit the maximum streaming bitrate to 300 kilobits per second.

For both these rules to work, they need to be in separate policy layers. If they were in the same policy layer, the rule listed second would never be applied.

Ordering Policy Layers

The order of policy layers is also important. The SG appliance evaluates policy layers in the order in which they are listed in VPM. When the SG appliance is going through policy layers, it does not execute a given rule as soon as it finds that it meets the specific situation. Rather, it compiles a list of all the rules that meet the condition; when it has gone through all the policy layers, it evaluates the list, resolves any apparent conflicts, and then executes the required actions. If there is a conflict between rules in different policy layers, the matching rule in the policy layer evaluated last takes precedence.

In the above example, there are two Web Access policy layers: one contains a rule stating that Sales personnel can access certain Web sites without authenticating, and the other states that when they do access these Web sites, limit the available bandwidth. The order of these policy layers is irrelevant. The order is irrelevant because there is no conflict between the rules in the layers.

The following is an example in which the order of policy layers does matter. Assume all URL requests from members of the purchasing department are directed to a single proxy server. To discourage employees from surfing the Web excessively during business hours, a company creates a Web Authentication Policy rule that says: "Whenever a client request comes in to the proxy server, prompt the client to authenticate."

Members of the purchasing department, however, need to access specific Web sites for business reasons, and the company does not want to require authentication every time they do this. So they create a Web Access policy rule that says: "If any member of the purchasing department sends a request to a specific URL contained in a combined-object list, allow access."

The policy layer with the first rule needs to come first in evaluation order; it is then overridden by the second rule in a subsequent policy layer.

Principle Policy Layer Design Rule

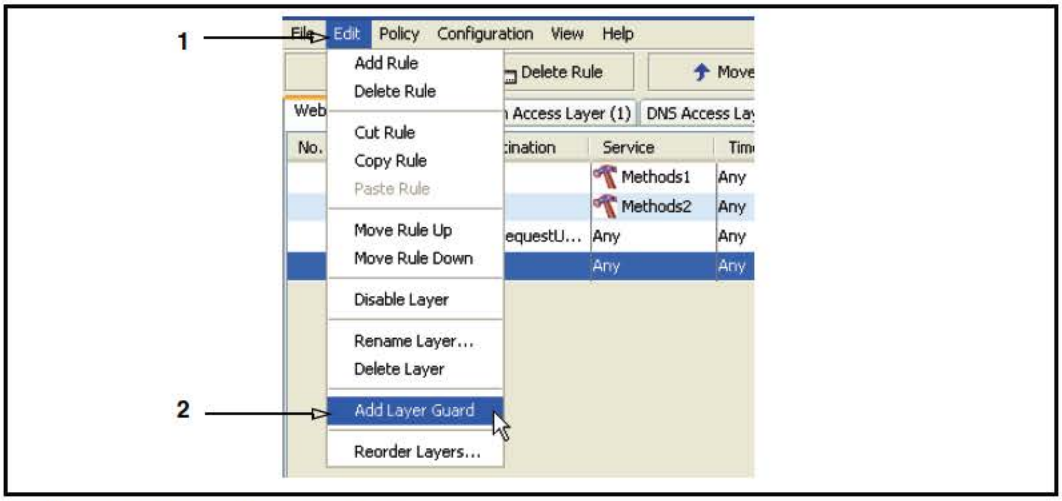
Always go from the general to the specific; that is, establish a general rule in an early policy layer, then write exception rules in later policy layers.

About the Layer Guard Rule

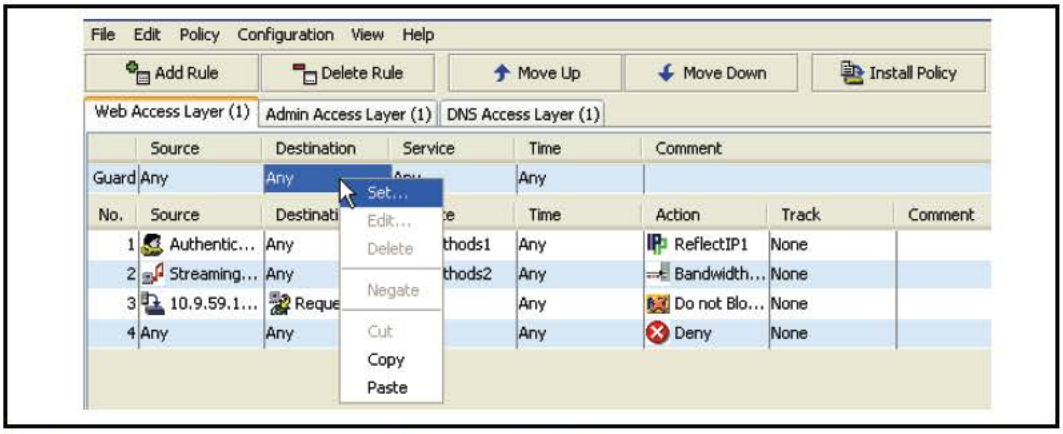
The VPM layer guard feature allows you to set a condition by which the whole layer is evaluated or not. This saves system resources, especially if you have layers with large numbers of rules. When added, the layer guard is a single rule table that appears above the selected layer. The layer guard rule contains all of the columns available in the layer except for the **Action** and **Track** columns. These columns are not required because the rule itself does not invoke an action other than allowing or not allowing policy evaluation for the entire layer. All of the objects valid in the available columns are selectable and configurable in the layer guard rule, just as they are in the layer.

You cannot add a layer guard rule until you have created other policy layer rules.

To add a layer guard:



1. From the Menu Bar, click **Edit**.
2. Select **Add Layer Guard**. The layer guard rule displays above the evaluation rules.



3. Right-click any column in the layer guard rule; select **Set**.

4. Define an object or objects just as you would in a policy layer. These objects determine if the rest of the rules in the layer are evaluated. For example, if you specify a Destination IP address in the layer guard rule, the other rules in the layer are evaluated only when the SG appliance detects the a transaction destined for that IP address.

Note: If you create and install a "Notify User" object, the following layer guard CPL is automatically added to the Web Access, Web Content, and SSL Access policy layers: `"condition=!__is_notify_internal"`. This is required for compatibility does not require any user interaction or tasks.

Disabling or Deleting a Layer Guard Rule

By default, a layer guard rule is enabled. You can disable (which retains the rule) or delete the rule from the VPM. Right-click **Guard** and make a selection.

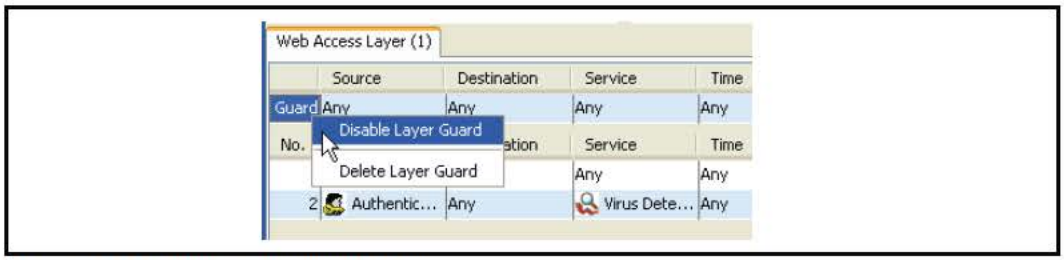


Figure 3-11. Disabling a layer guard rule.

Note: Alternately, right-click the layer tab to add, disable, or remove a layer guard rule.

Installing Policies

As you add policy layers and rules, your work is saved in a file on the SG appliance. However, policies only take effect after you install the policies and the generated XML has been validated. The SG appliance then compiles the policies into CPL format and saves the resulting policies in the `vpm.cpl` file. This overwrites any policies previously created using VPM. The appliance saves VPM-generated policies in a single file and loads it all at once. You do not need to load policies separately, as is the case with the local or central policy files.

To install policies:

- Select **File > Install Policies**, or
- Click **Install Policies** on the Rule bar.

  The VPM validates the generated XML for any issues, such as missing layers. If the validation passes, the CPL is generated and the policies are loaded.

  If the XML fails the validation, a dialog appears allowing you to:

  - Revert to the policy currently installed on the SG appliance, or
  - Continue to edit the policy and attempt another installation.

  Furthermore, the failed XML file is written to your hard disk; view this file to troubleshoot the failed XML. The default location for this file is:

  C:\Documents and Settings\user.name\bluecoat\vpm_err.xml

Notes

The **Category** and **Notify User** objects and the **DNS Lookup Restrictions**, **Reverse DNS Lookup Restrictions**, and **Group Log Order** configuration objects generate CPL, regardless if they are or are not included in rules. These specific objects and features allow users to edit categories and lists that might or might not be used in current policies.

Managing Policy

This section describes how to manage VPM policy.

Refreshing Policy

In between occurrences when either VPM is closed and reopened or Install Policies is invoked, VPM does not recognize changes to VPM-managed policy that were made on the SG appliance through another method. For example:

- Another administrator opens a separate VPM to make changes.
- Another administrator edits the local or central policy file through the serial console.
- Another administrator makes edits the local or central policy file.
- A new content filter database is downloaded automatically and the new update contains category changes.
- A new content filter database is downloaded manually by an administrator.

Reverting to a Previous Policy

If after creating new policies or editing an existing policy you decide to abandon the process and continue with the existing policy installed on the SG appliance, you can revert to that version. All current changes are deleted (VPM provides a verification prompt).

To revert to an existing installed policy:

Select **File > Revert to Existing Policy on** SG **Appliance**.

Changing Policies

You can change, edit, delete, add to, and otherwise manage policies created in VPM at any time by returning to VPM and working with policy layers and rules just as you did when creating them.

Managing Policy Layers

This section describes how to perform edits of policy layers.

Renaming a Policy Layer

The VPM allows you to rename policy layers and disable and re-enable layers.

To rename a policy layer:

1. Right-click the tab of the policy layer and select **Rename**. The Rename New Layer dialog appears.
2. Rename the layer and click **OK**.

Disabling a Policy Layer

Disabling policy layers allows you to remove a subset of the employed policy without losing the rules and the effort put forth to create them. Once disabled, the policy in that layers is ignored. You can re-enable a disabled layer at any time.

To disable or enable a policy layer:

Right-click the tab of the policy layer and select **Disable Layer**. The layer name text turns red and the layer rules are greyed-out.

To re-enable a layer, repeat this step and select **Enable Layer**.

Deleting a Policy Layer

You can completely remove a policy layer.

Important: Once deleted, a layer cannot be recovered.

To delete a policy layer:

1. Right-click the tab of the policy layer to be deleted.
2. Select **Delete Policy** from the drop-down list.

Note: All of the above procedures can be accomplished from the **Menu Bar>Edit** drop-down list.

Managing Policy Rules

Occasionally, you might need to temporarily disable rules in a policy layer; for example, when troubleshooting compiles errors and warnings. This might help confirm that the SG appliance can successfully compile the remaining policy. After disabling a rule, you can edit the objects and re-enable the rule.

To disable or enable a rule:

1. Click the appropriate policy layer tab.
2. Right-click in the **No.** column.
3. Click **Disable Rule** on the **shortcut** menu. The policy editor changes the rule text color to red.
4. To enable the rule, repeat step 3. After you enable a disabled rule, the policy editor changes the rule text color to black.

Installing VPM-Created Policy Files

Policies created with VPM are saved on the specific SG appliance on which they are created. SGOS automatically creates the following files when saving VPM-created policies:

```
config_policy_source.xml
config_policy_source.txt
```

You can install VPM policies that were created on another SG appliance. This requires the following steps:

1. Copy the two VPM files, to be shared, to a Web server from the SG appliance on which they reside.
2. Use the Management Console or CLI to load VPM files on another SG appliance.

To copy VPM files from a SG appliance to a Web server:

1. Select **Statistics > Advanced**.
2. Scroll down and click **Policy**.

   The page jumps down to the Policy files links.

Figure 3-12. Policy Files in Custom URLs

3. Right-click the **Show VPM CPL policy** link.
4. In the Save As dialog, enter the full path to a directory on the Web server before the file name and click **OK**.

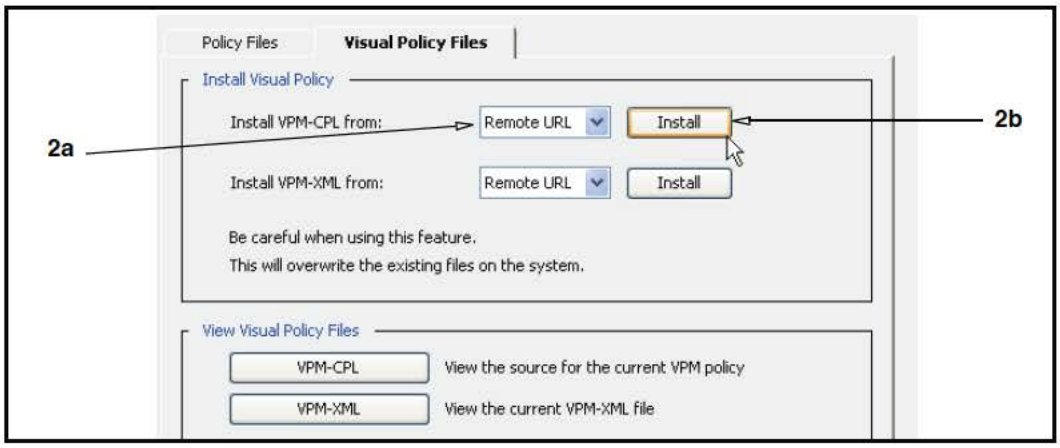
Important: The Save As dialog offers the appropriate default file name (`config_policy_source.xml` or `config_policy_source.txt`). You can change the names, including the extension. This can be helpful if an enterprise is using various sets of shared VPM files. You could rename files to indicate the SG appliance on which they were created, for example, or for a department that has a set of VPM-specific policies, used perhaps in multiple locations (**sales_vpm.cpl** and **sales_vpm.xml**).

---

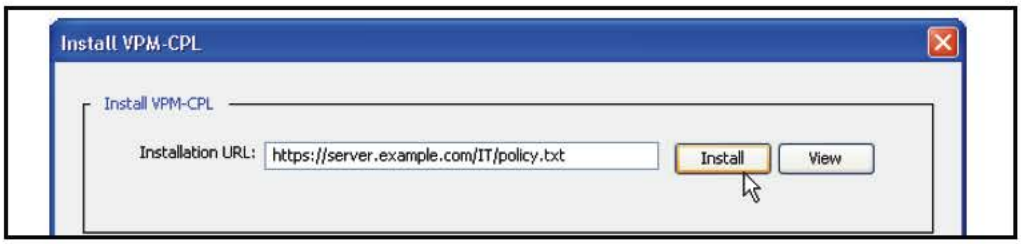
5. Repeat the previous step for the second VPM file.

To load VPM files to a SG appliance:

1. Select **Configuration > Policy > Policy Files > Visual Policy Files**.



2. In the **Install Visual Policy** field:
   a. Select **Remote URL** from the **Install VPM-CPL from** drop-down list.
   b. **Click Install**. The Install VPM-CPL dialog appears.



   c. In the **Installation URL** field, enter the URL to the VPM CPL file copied to the Web server (this is the file with the default `.txt` extension) and click **Install**.
   d. Repeat Steps a through c to enter the URL to the second VPM XML file copied to the Web server (this is the file with the default `.xml` extension) and click **Install**.
3. Click **Apply**.

Notes

- If VPM files already exist on the SG appliance, the URLs to those files display in the two file fields. To replace them, delete the URLs and type new ones. Installing new files overwrites any that are already present.
- To review VPM-generated policies before installing them, enter the URL to the CPL file on the Web server and click **View**.
- Regardless of whether you are installing new VPM files, you can review the CPL or XML files of the policies currently on the SG appliance. Click **VPM-CPL** and **VPM-XML** in the **View Visual Policy Files** box at the bottom of the dialog.
- Never edit either of the VPM files directly. Change the files only by working with the policies in VPM and saving changes there.

To load VPM files to a SG appliance:

The two commands in the first step load one of the VPM policy files; the commands in the second step load the other policy file. In each case, *`url`* is the complete path, including file name, to the appropriate file on the Web server.

1. At the `config` command prompt, enter the following commands:

   ```
   SGOS#(config) policy vpm-cpl-path url
   SGOS#(config) load policy vpm-cpl
   ```

2. At the `config` command prompt, enter the following commands:

   ```
   SGOS#(config) policy vpm-xml-path url
   SGOS#(config) load policy vpm-xml
   ```

Viewing the Policy/Created CPL

View the CPL generated by installing VPM-created policy from VPM or the Management Console.

To view the generated CPL through the VPM:

Select **View > Generated CPL**.

To view the VPM policy file:

Select **View > Current** SG **Appliance VPM Policy Files**.

Important: Do *not* edit or alter VPM-generated files by opening the VPM policy file and working in the generated CPL. To edit, change, or add to VPM policies, edit the policy layers and re-install the policy.

Section E: Tutorials

This section contains the following topics:

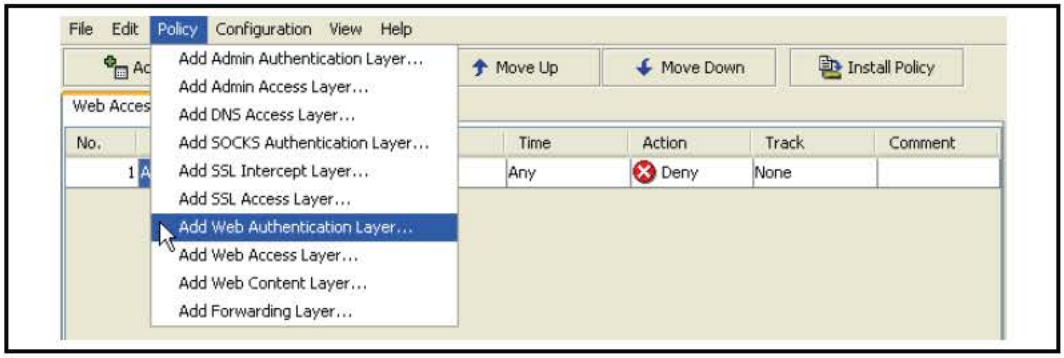
Tutorial—Creating a Web Authentication Policy

This section is a tutorial that demonstrates how to create policies and rules for Web authentication.

Use Web Authentication policies to specify whether the individual making a request is prompted to authenticate by entering a username and password. In this example, a company uses a PAC file to configure most employee browsers to connect to a specific IP address on the SG appliance. It wants these users to authenticate when their browsers send a request to the proxy.

To create a policy layer:

1. Start the VPM from the Management Console: **Configuration > Policy > Visual Policy Manager.**



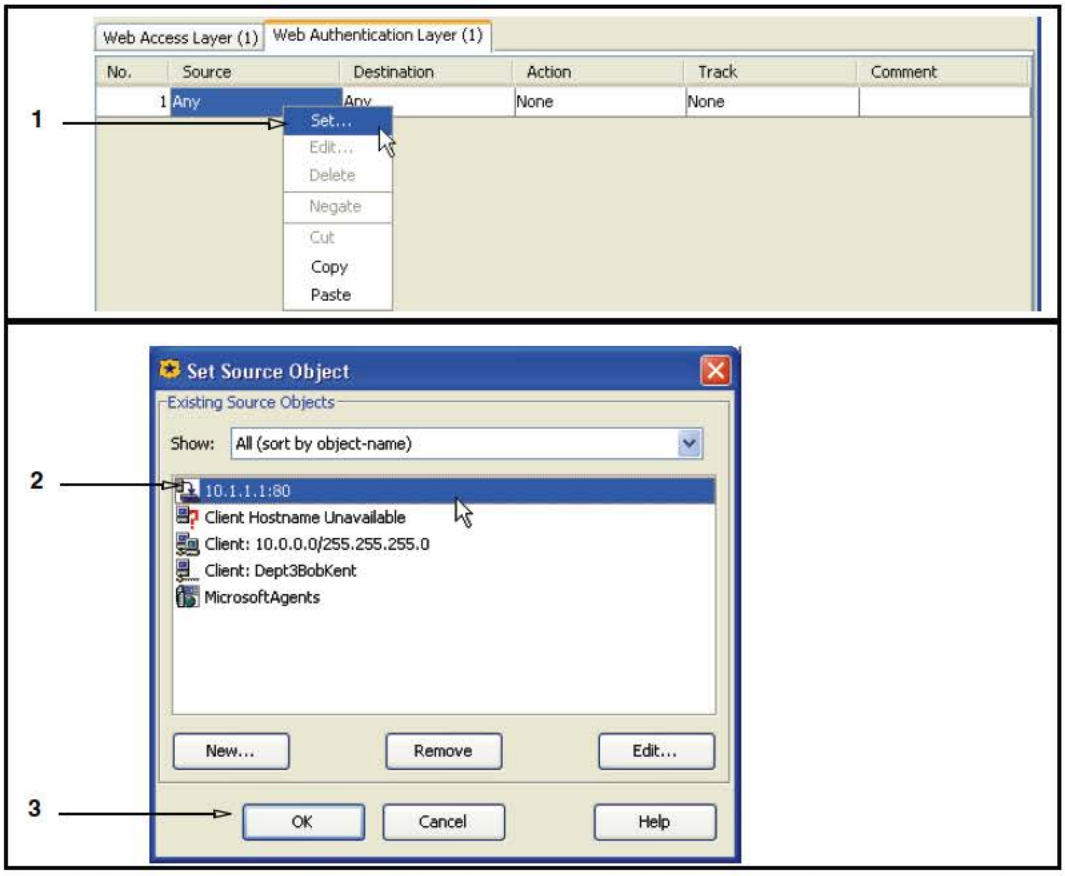
2. Select **Policy > Add Web Authentication Layer.**

3. A dialog displays offering a default name for the layer, consisting of the layer type and a number. Rename the layer or accept the default and click **OK**.

   The VPM creates the new layer tab and adds a blank rule.

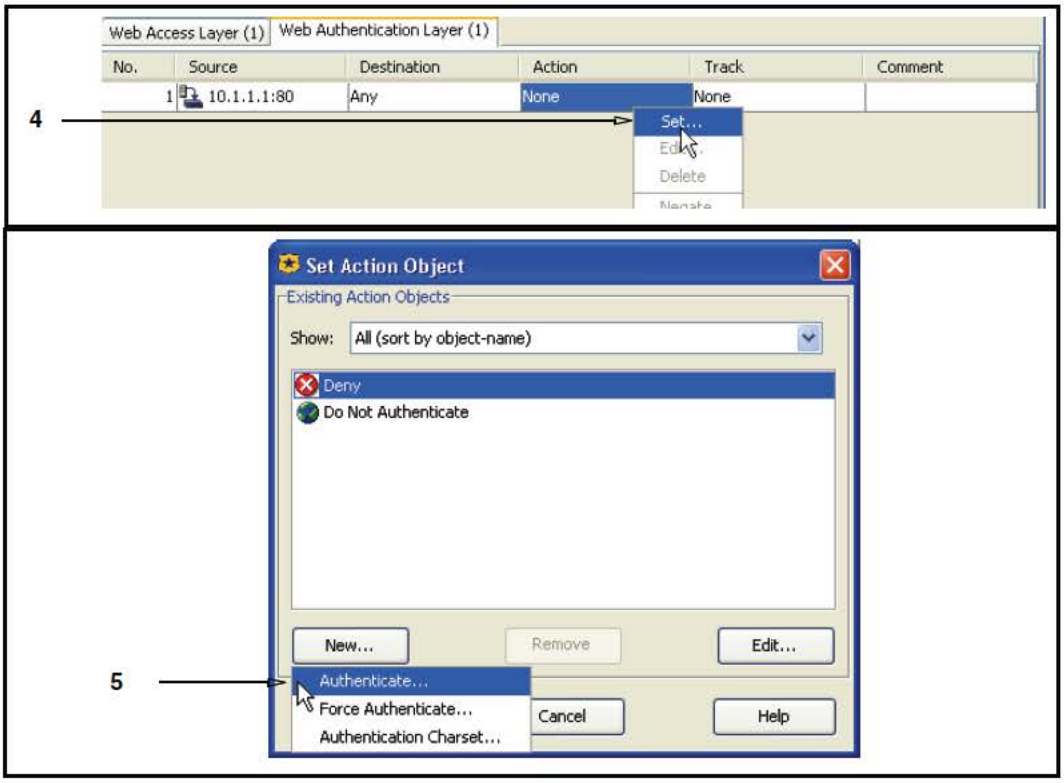
Example 1: Create an Authentication Rule

By default, the unmodified rule applies to everyone whose browsers connect to a specific IP address.



1. Right-click the **Source** cell to drop the menu; select **Set** to open the Set Source Object dialog.
2. Select a proxy IP address or port; if necessary, click **New** to create a new one. This example selects the IP address on the SG appliance where the PAC file sends most employee browsers.
3. Click **OK** to enter the IP address in the **Source** cell.

Section E: Tutorials



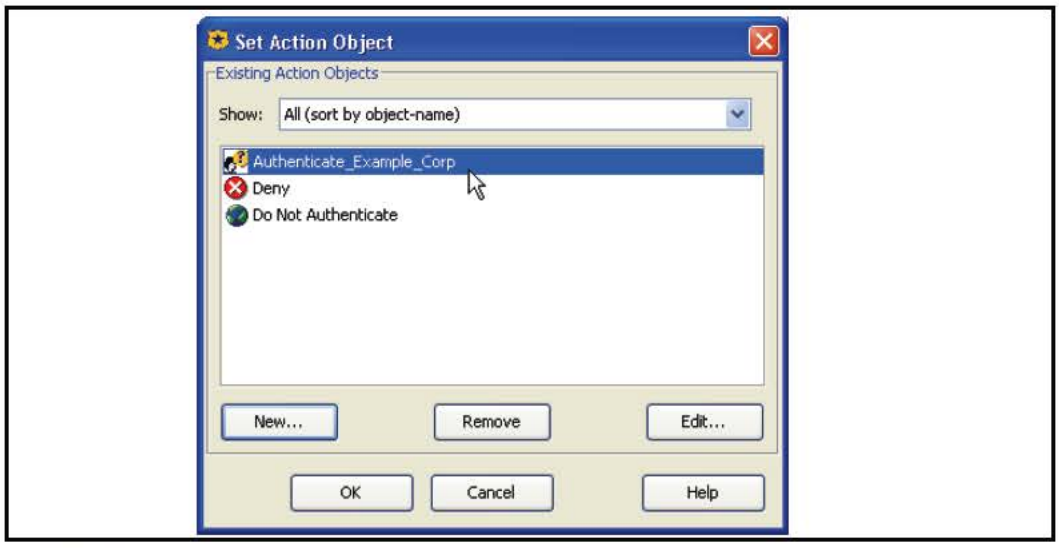
4. Create an authentication Action object. Right-click the **Action** cell to drop the menu and select **Set**; the Set Action Object dialog displays.

5. The only objects available are the pre-existing static objects, so you must create a new Authenticate object. Click **New** and select **Authenticate**. The Add Authenticate Object window displays.

6. For this example, the following fields are:

- **Name**—Every configurable object has a name. The default name **Authenticate1**; change to **Authenticate_Example_Corp**, which is how it is listed in the Add Object window.
- **Realm**—Specifies an LDAP realm.
- **Mode**—Specifies the authentication realm mode is **Origin IP**.

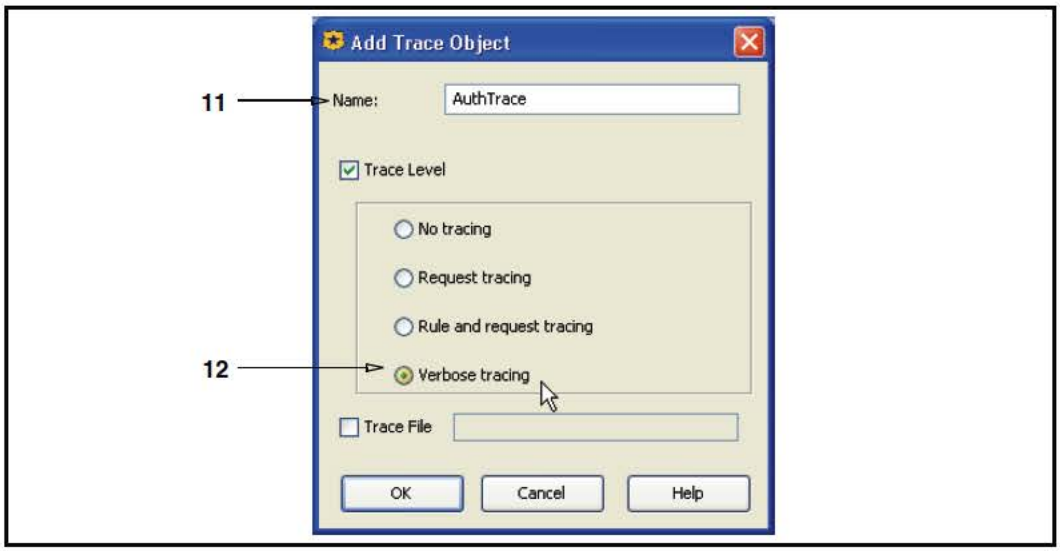
7. Click **OK** to close the Add Action Object window, with the new Authenticate object in the list.



8. Click **OK**.

Figure 3-13. Completed Action Object

9. Create a Trace object to log all authentication activity. Right-click the **Track** cell to drop the menu and select **Set**; the Set Track Object dialog appears.
10. You must create a new Trace object. Click **New** and select **Trace**; the Add Trace Object appears.



11. In the **Name** field, enter **AuthTrace**.
12. Click **Trace Level** and **Verbose** to enable verbose tracing, which lists the rules that were skipped because one or more of their conditions were false and displays the specific condition in the rule that was false.
13. Click **OK**.
14. Click **OK** again to add the object. The rule is complete.

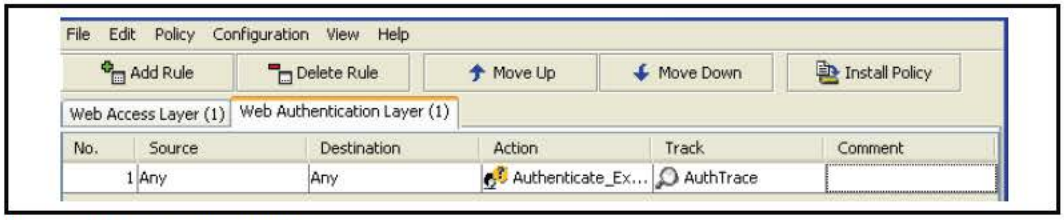
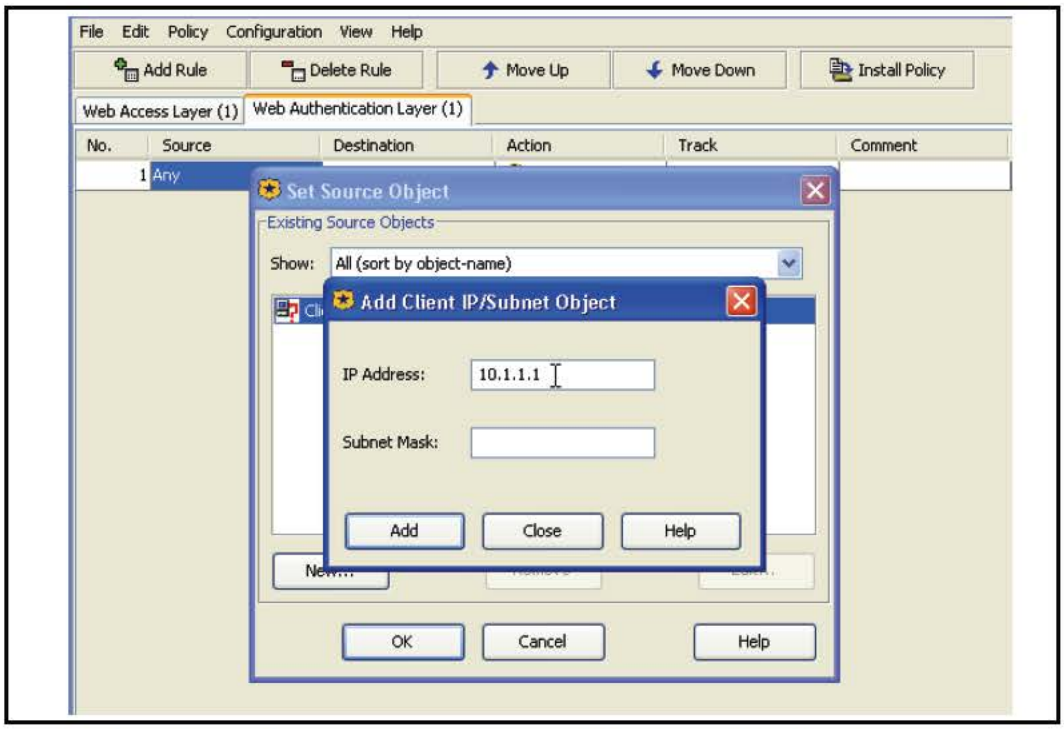


Figure 3-14. Completed Rule

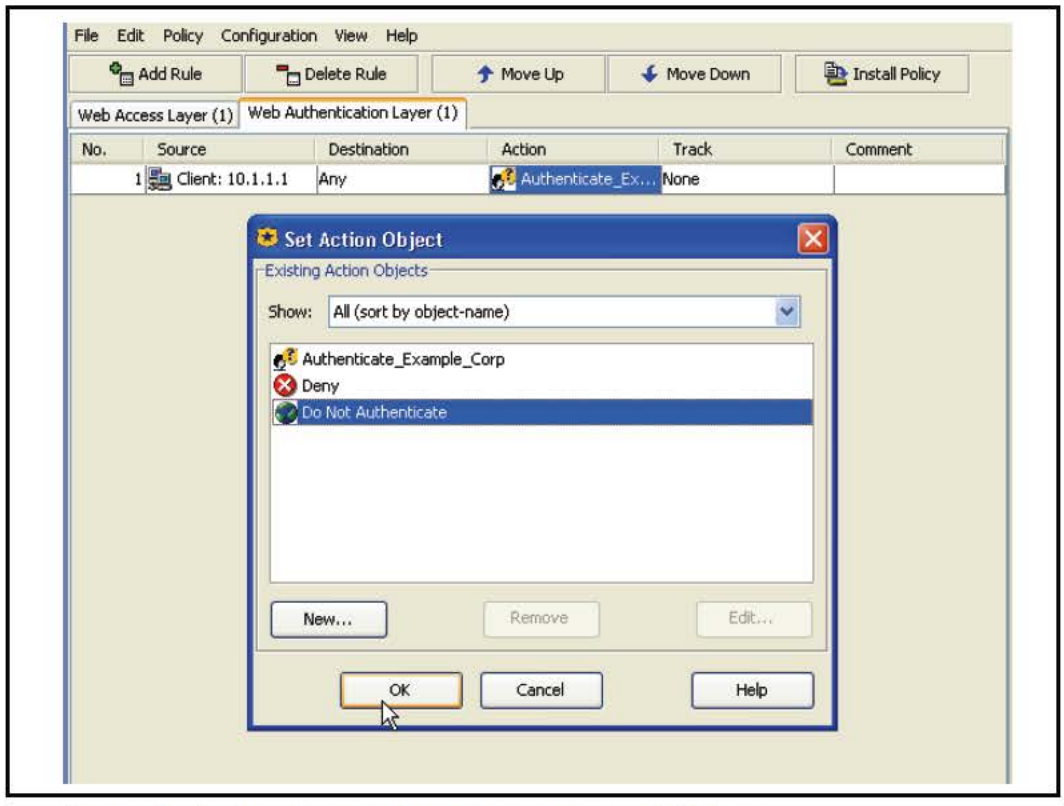
Example 2: Exempt Specific Users from Authentication

Certain individuals and groups are exempt from the above restriction. Individuals in the purchasing department are required to access the Web often so they can order online from supplier Web sites, and the company does not want them to authenticate.

1. Click **Add Rule** to add a new rule to this policy layer.



2. People in the purchasing group use the same PAC file and thus their browsers are directed to the same IP address: `10.1.1.1`.

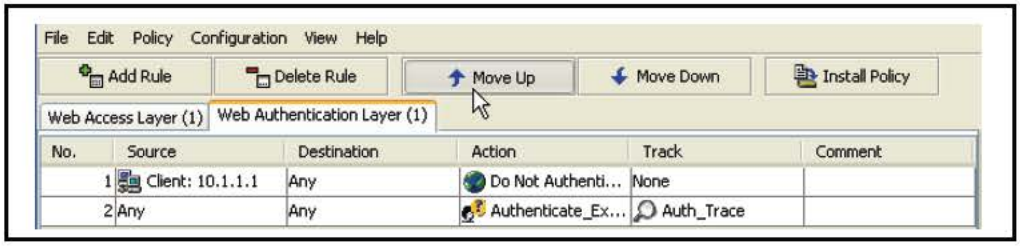


3. Change the Action object to **Do Not Authenticate** and click **OK**.

The new rule in the policy layer accepts the default Action Object to not authenticate and does not require a Trace Object.

Figure 3-15. Updated second rule.

However, a problem exists. The second rule cannot be evaluated because the first rule affects everyone who goes through the proxy. The rules need to be reversed.



4. Select the second rule and click **Move Up** to reorder the rules.
5. Click **Install Policy**.

Tutorial—Creating a Web Access Policy

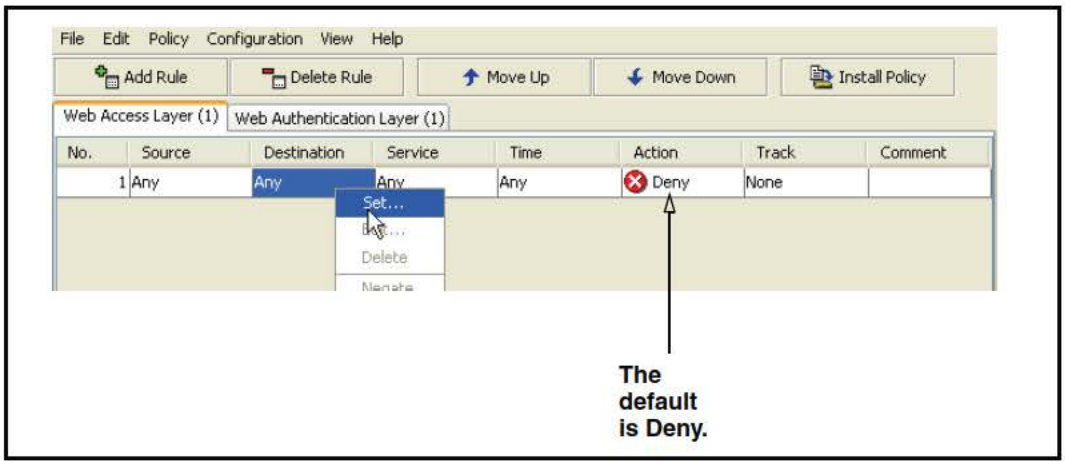
This section is a tutorial that demonstrates how to create policies and rules for Web access.

Use SG appliance policies to define end-user access to Web resources. For more information about Web access policies, refer to *Volume 7: Managing Content*. This section provides examples.

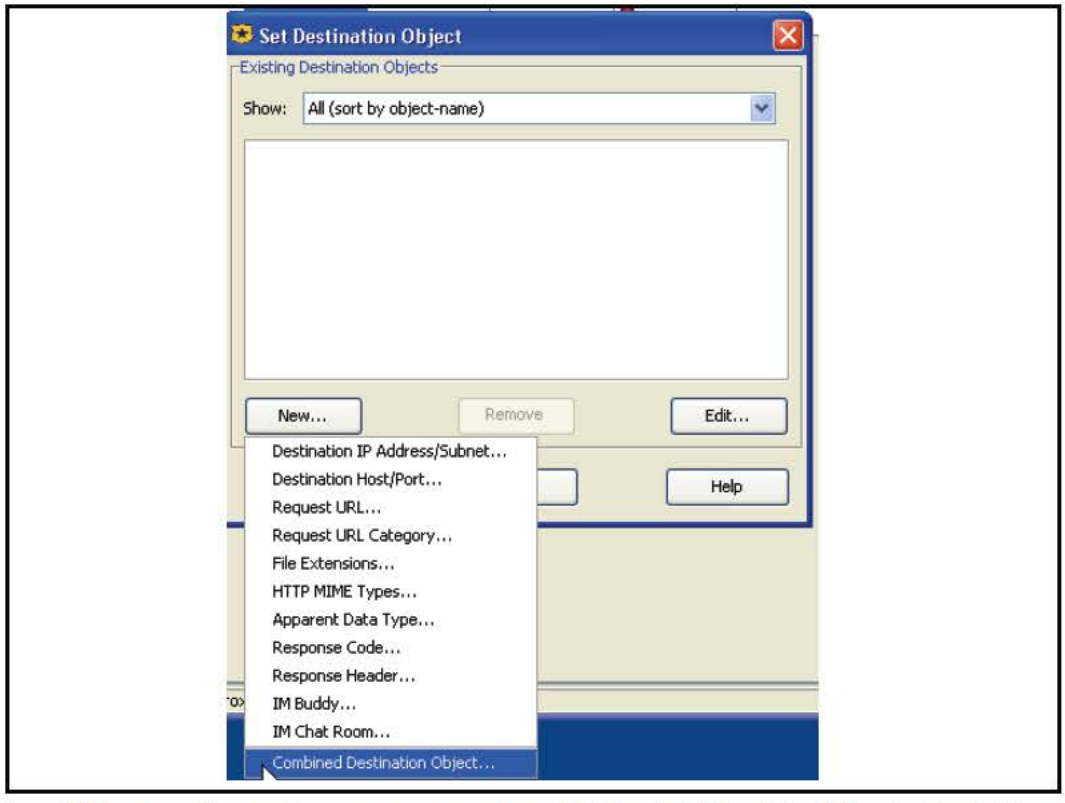
Example 1: Restrict Access to Specific Websites

This example demonstrates a simple rule that denies everyone access to specific job searching Web sites. This rule requires you to configure only one rule option; it uses the defaults for all other options.

1. Start the policy editor and select **Policy > Add Web Access Layer**. The VPM displays a tab with the name of the new policy; beneath that is a new rule-specific row.

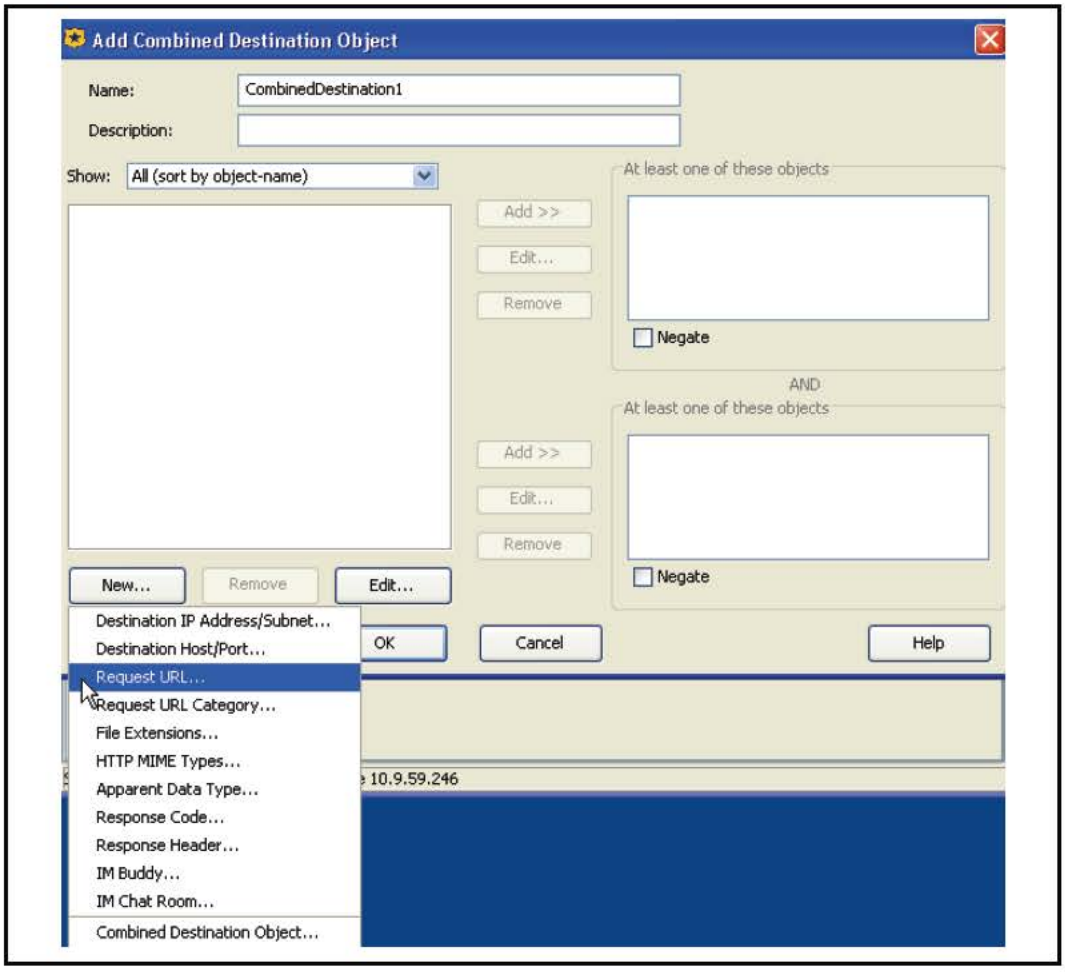


2. Right-click **Destination** and select **Set**; the Set Destination Object dialog appears.

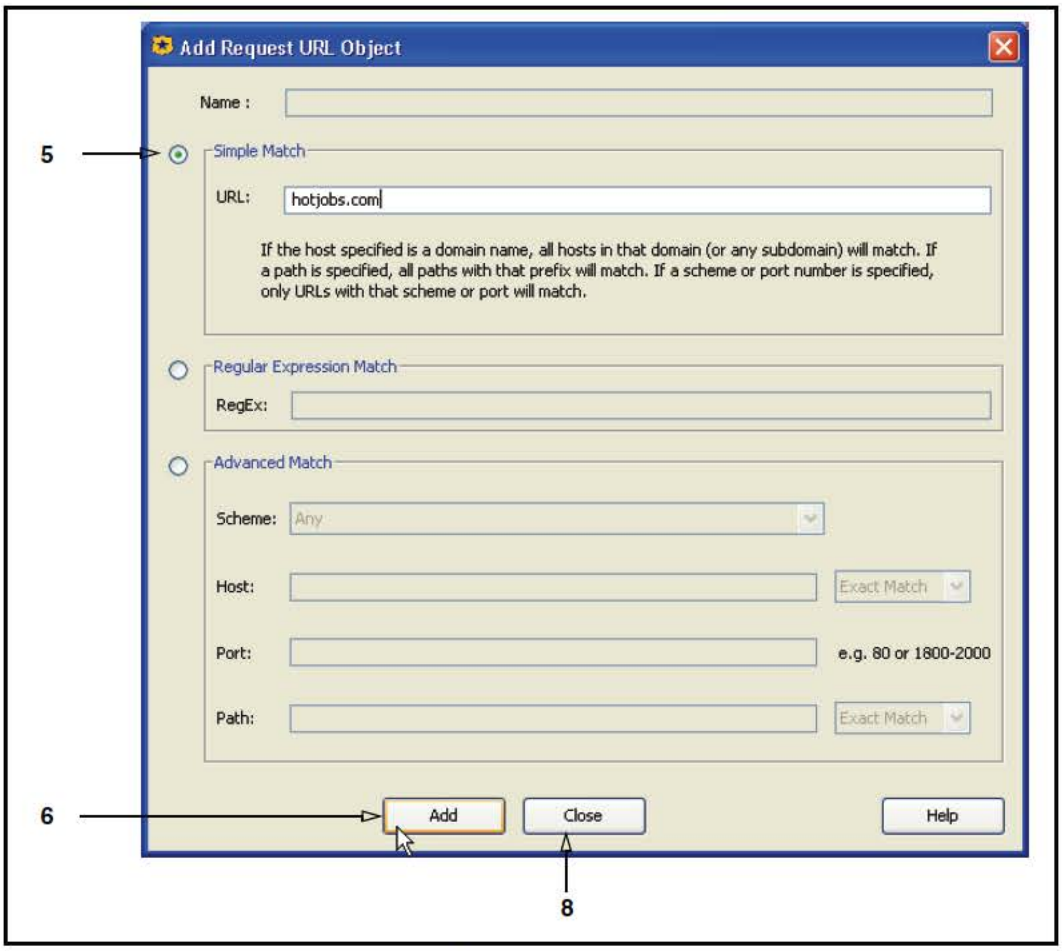


3. Click **New**; select **Combined Destination Object**. The Add Combined Destination Object dialog appears.

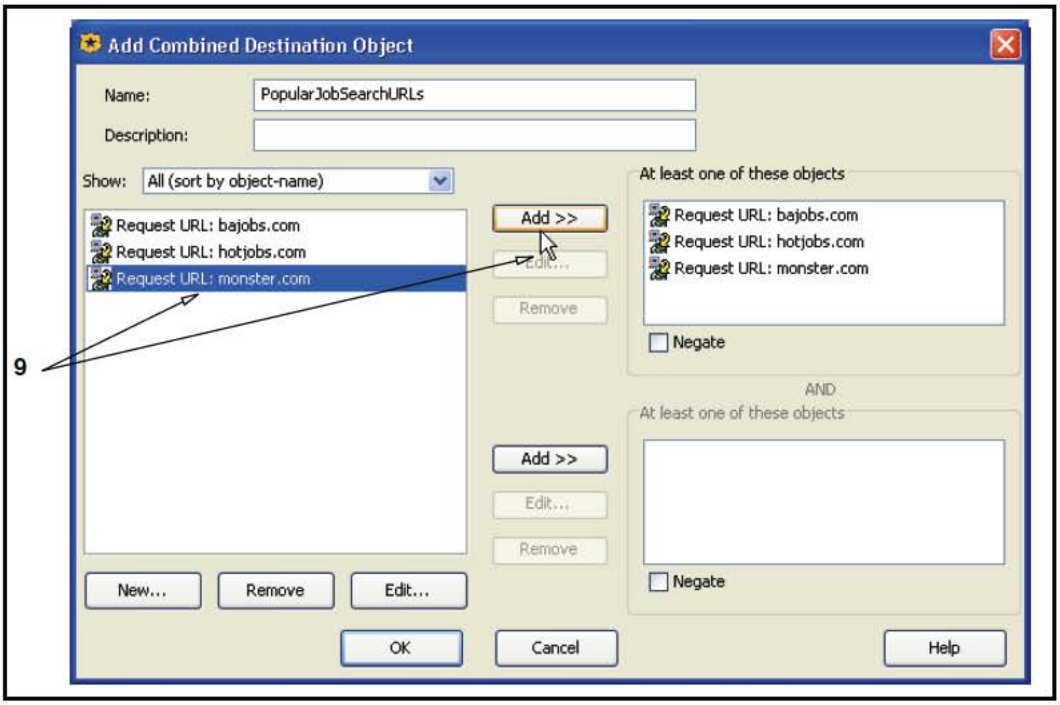
Section E: Tutorials



4. Select **New > Request URL**.



5. Click **Simple Match;** in the URL field, enter **hotjobs.com**.
6. Click **Add**.
7. Repeat Step 5, adding **monster.com** and **bajobs.com**.
8. Click **Close**.



9. Select each newly added URL and click the first **Add** button.
10. Click **OK**. The Set Destination Object now contains the individual URL objects and the combined object.
11. Select the **JobSearchURLs** combined object and click **OK**. The object is now part of the rule.

Figure 3-16. Completed Rule

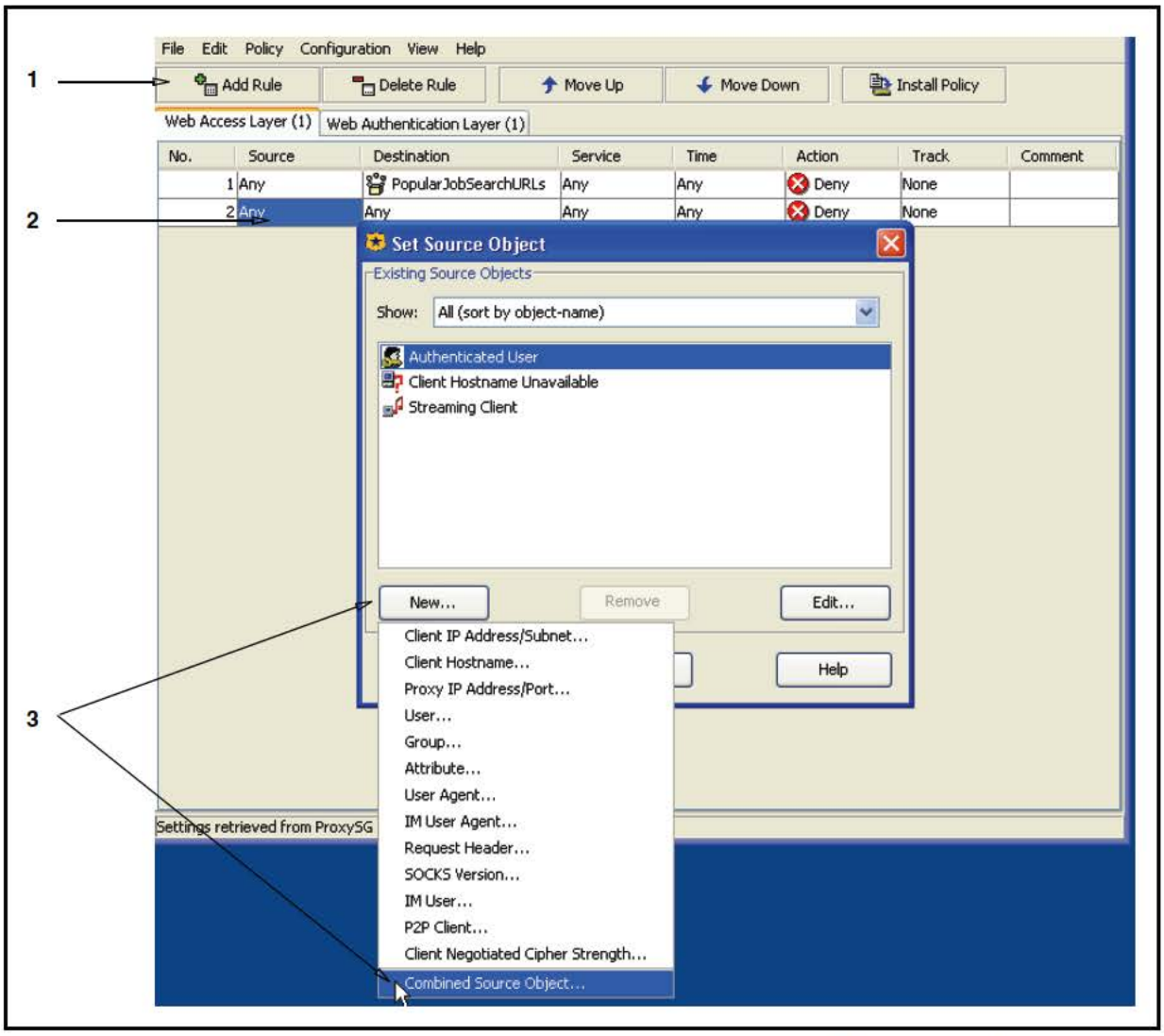
As the default action is deny, the rule is complete. No one can access these Web sites.

12. To activate the rule, click **Install Policies**.

Example 2: Allow Specific Users to Access Specific Websites

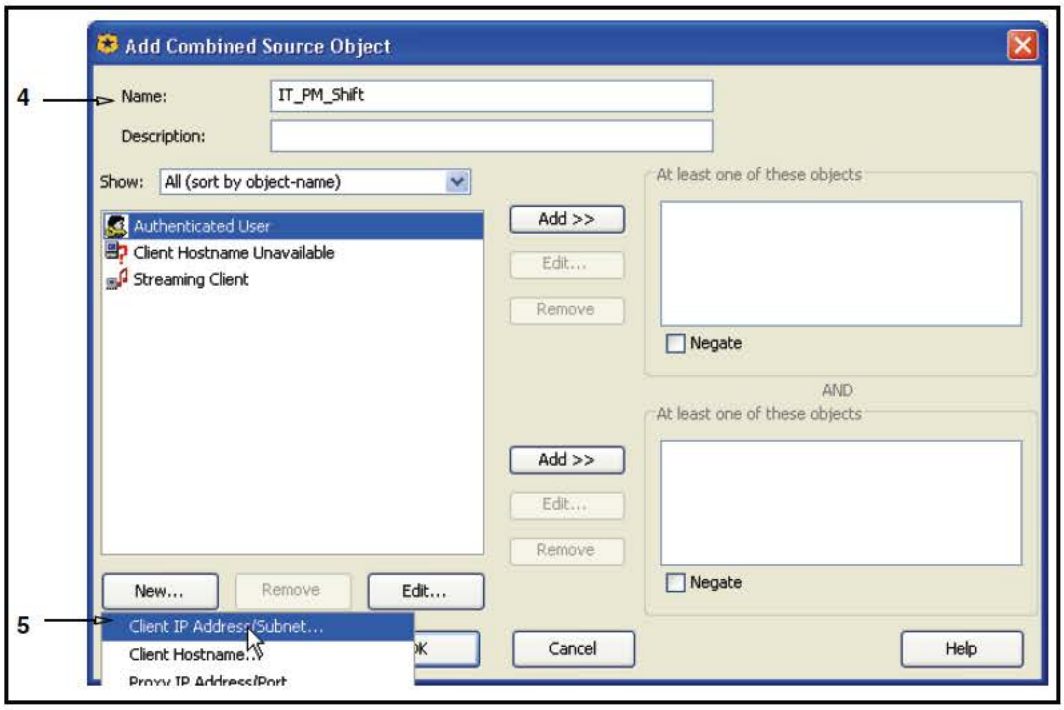
The after-hours IT shift is comprised of part-time college interns who are on call to handle small problems, but are not involved in major projects. Therefore, you allow them to browse certain sports and entertainment Web sites when all is quiet; access is allowed from two workstations and you still want to track their browsing activity.

To configure the Source object:

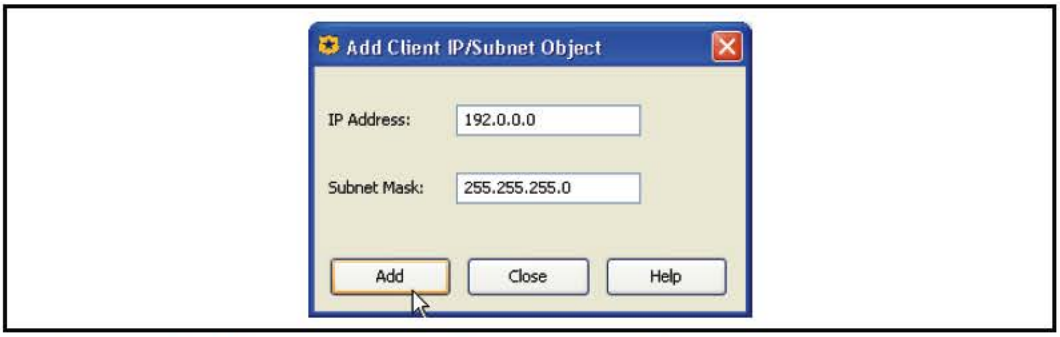


1. Add a new rule to the policy and position the pointer in the **Source** cell.
2. Right-click the **Source** cell and select **Set** to display the Add Source Object dialog.
3. Click **New** and select **Combined Source Object**; the Add Combined Source Object appears.

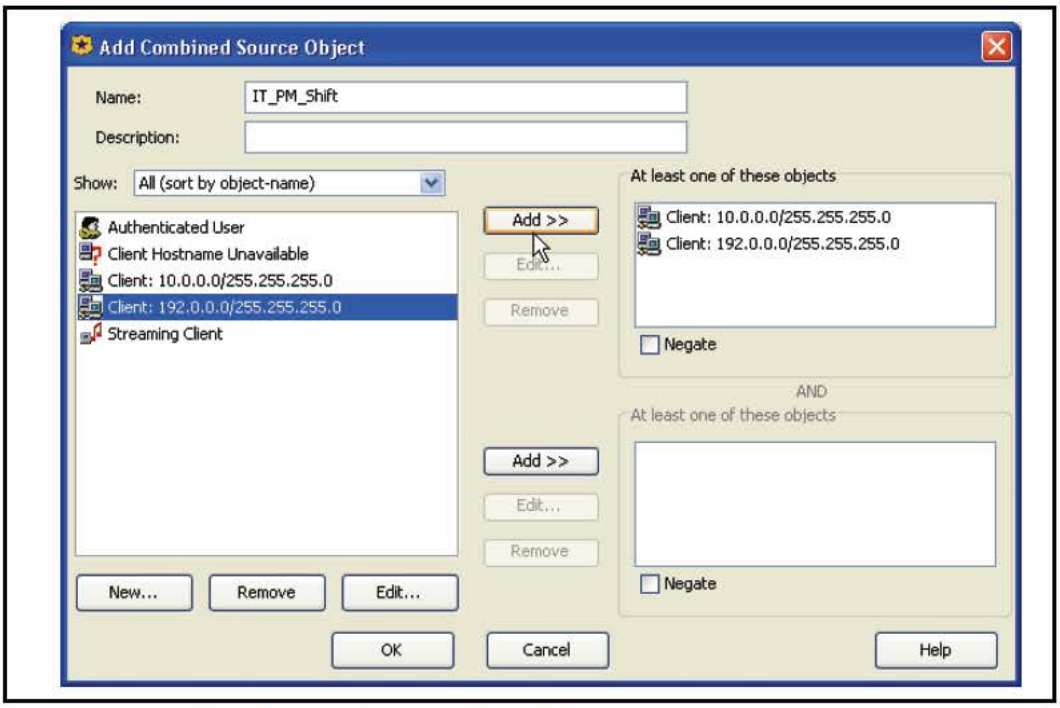
Section E: Tutorials



4. Name the object **IT_PM_Shift**.
5. Under the selectable list of objects, click **New** and select **Client IP Address/Subnet**; the Add Client IP Address/Subnet Object dialog appears.

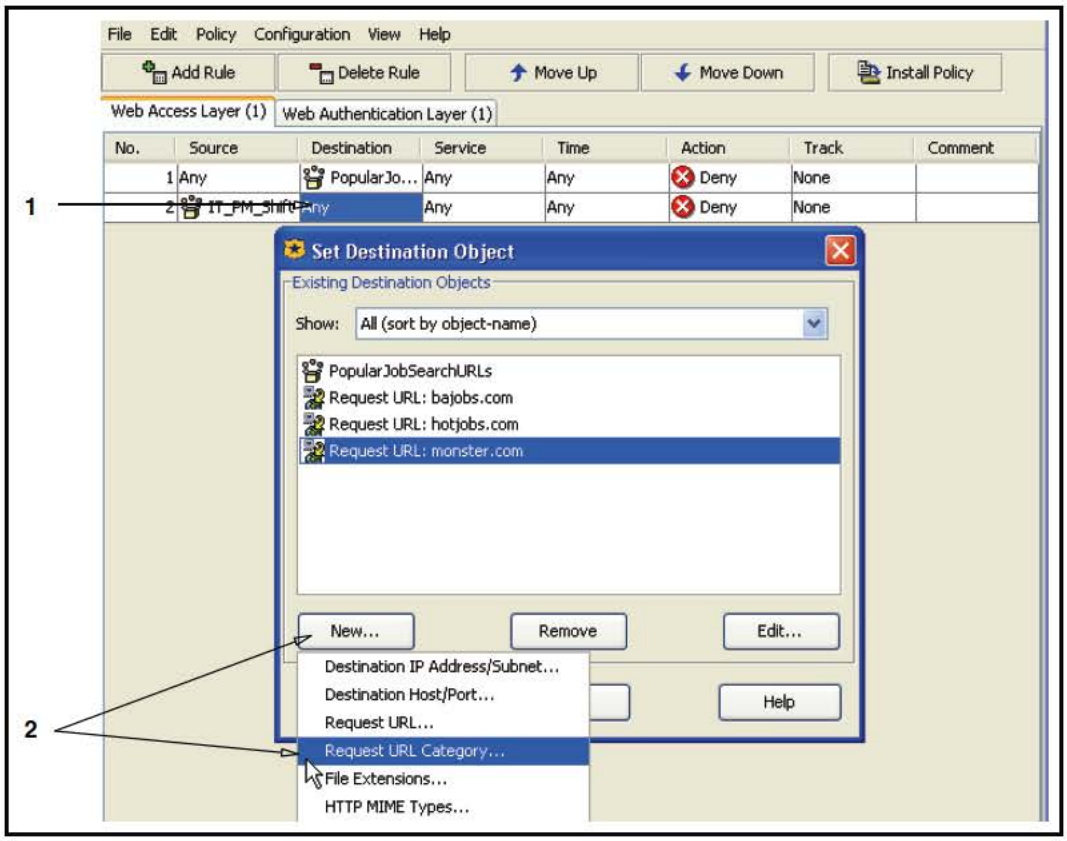


6. Enter the IP address of the first workstation and click **Add**; repeat for the second; click **Close**.

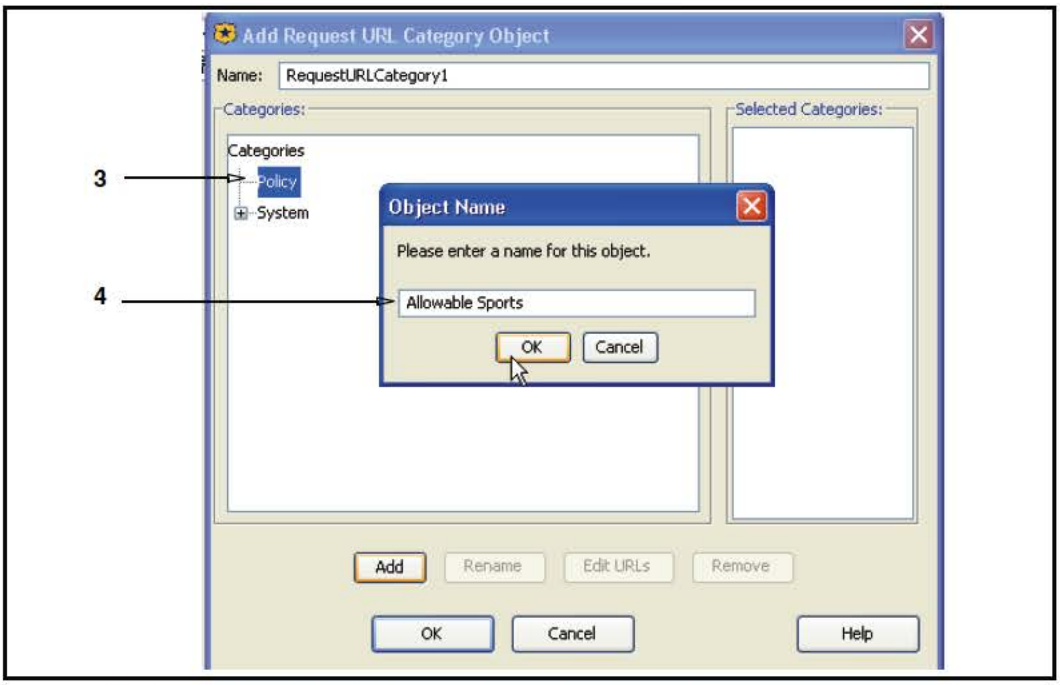


7. Select each IP address and click the first **Add**.
8. Click **OK**; click **OK** again to add the Source object to the rule.

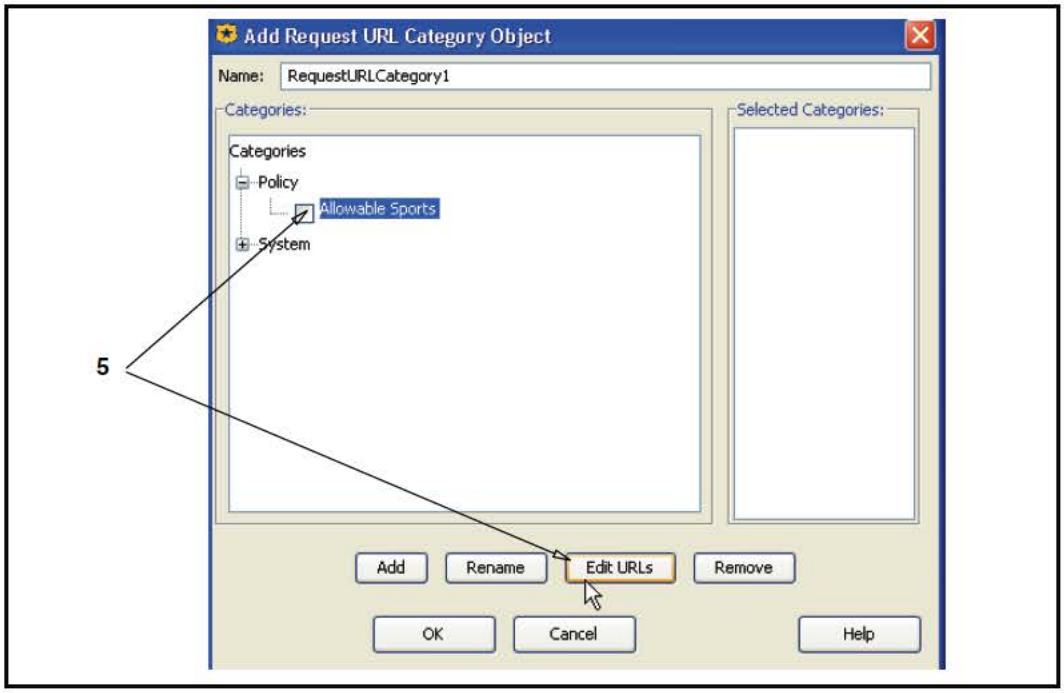
To configure the Destination object:



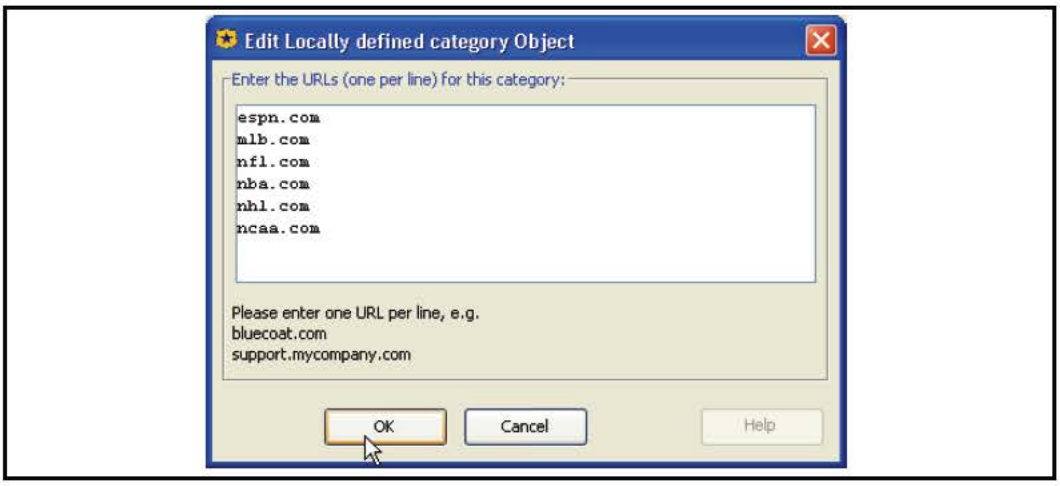
1. Right-click the **Destination** field and select **Set**; the Set Destination Object dialog appears.
2. Click **New** and select **Request URL Category**; the Add Request Category Object dialog appears.



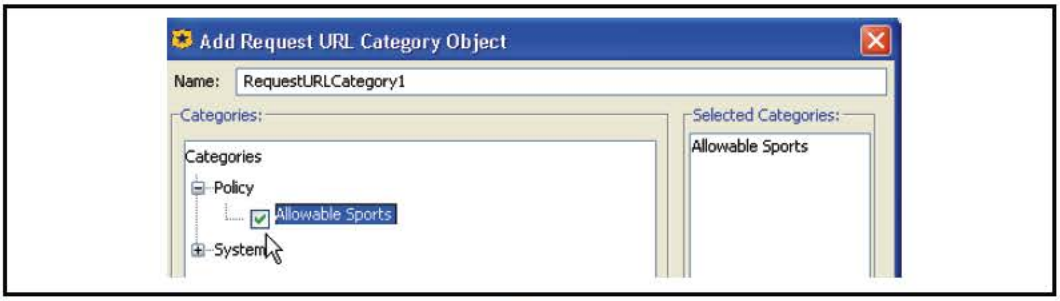
3. Select **Policy** and click **Add**; the Enter Name for New Category dialog appears.
4. Name the object **Allowable_Sports** and click **OK**.



5. Select Sports URLs. Click **Edit URLs**. The Edit Locally Defined Category Object dialog appears.



6. Enter the URLs for the allowable sports Web sites and click OK.



7. Under **Policy**, select **Allowable_Sports**; click **OK**.
8. Repeat Steps 3 through 7, creating a category called **Allowable_Entertainment** with the URLs **ew.com**, **rollingstone.com**, and **variety.com**.
9. Name the object **Allowable PM IT Websites**. Click **OK** twice to add the object to the rule.

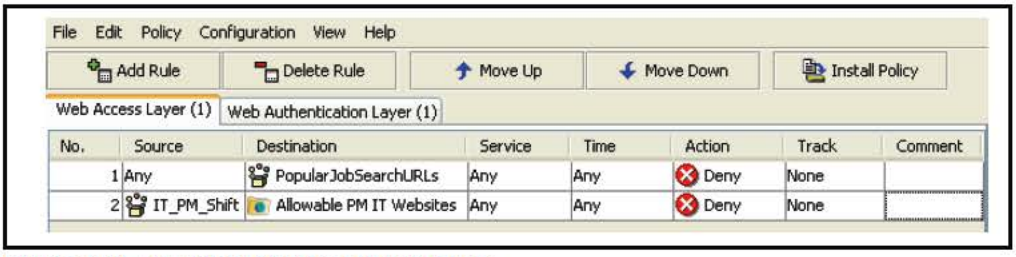
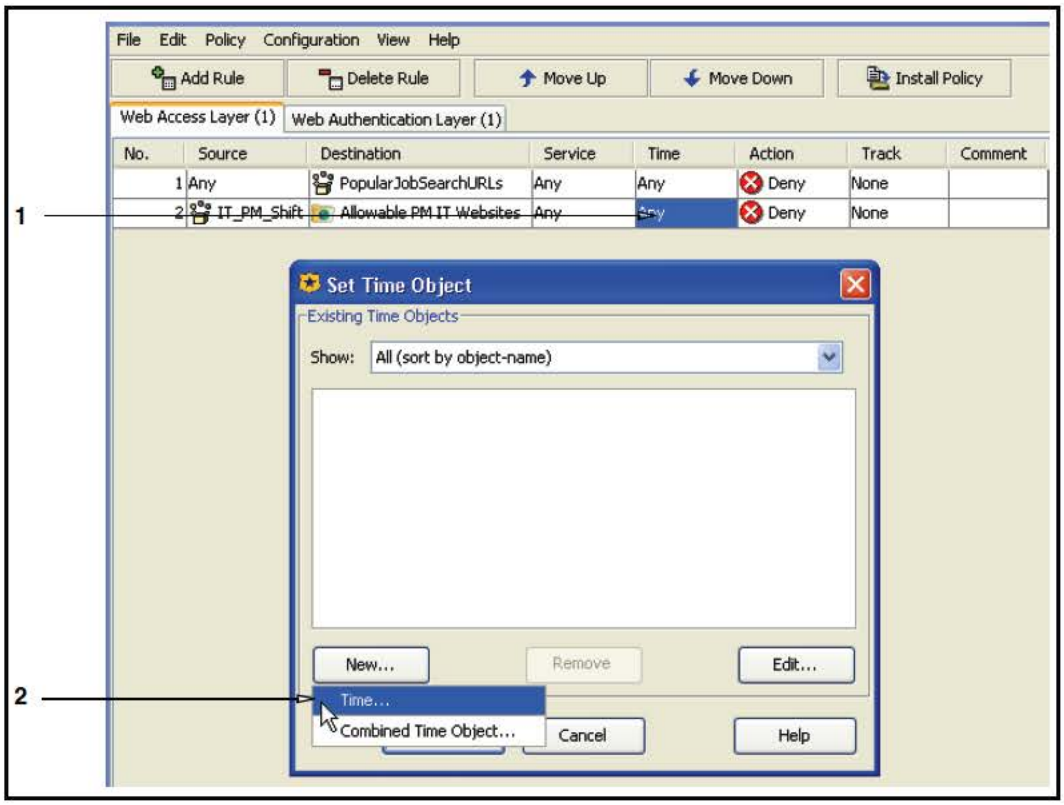


Figure 3-17. Completed Second Rule in Layer

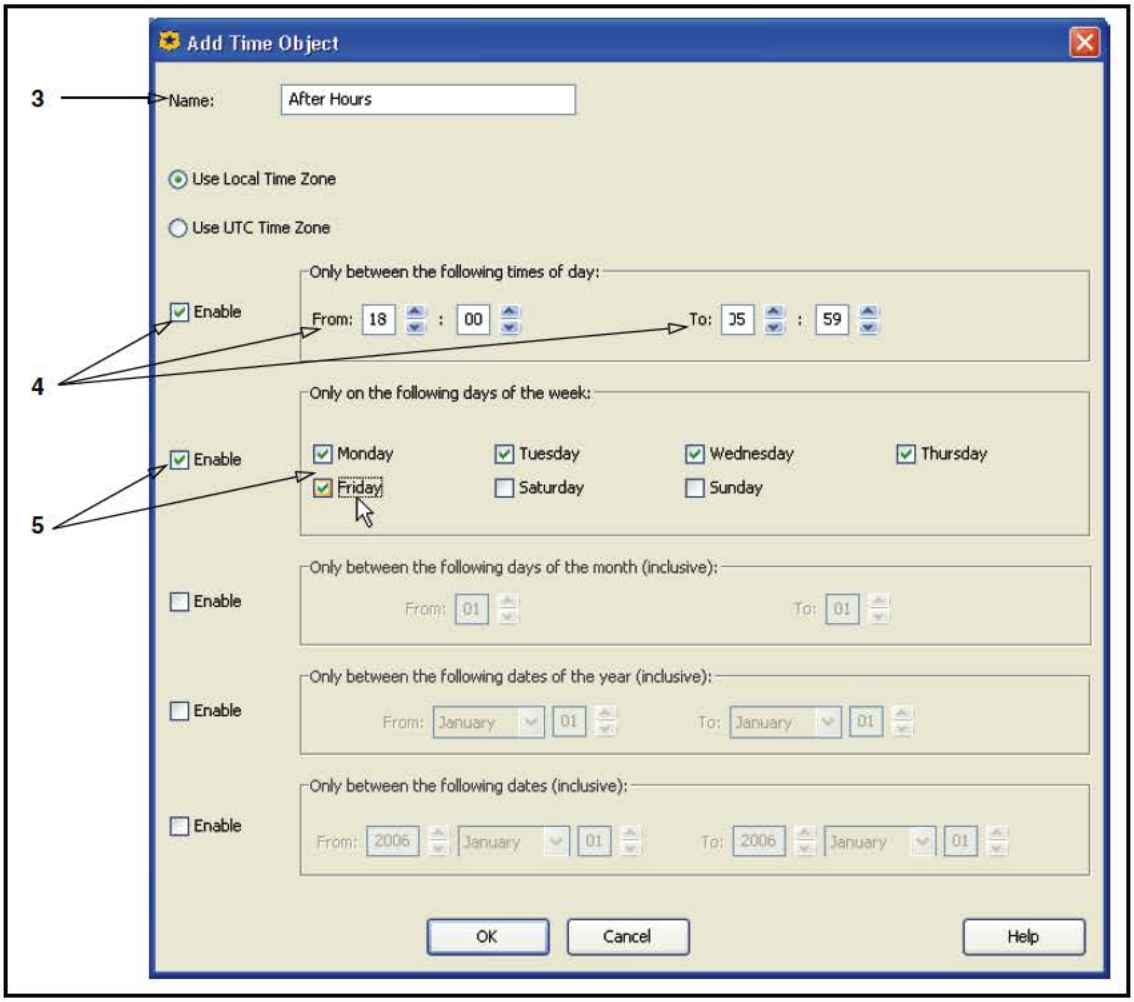
To configure the Time object:

This example allows the specified users to access the sports and entertainment Web sites after business hours.



1. In the second rule, right-click the **Time** field and select Set; the Set Time Object dialog appears.
2. Click **New** and select **Time Object**; the Add Time Object dialog appears.

Section E: Tutorials

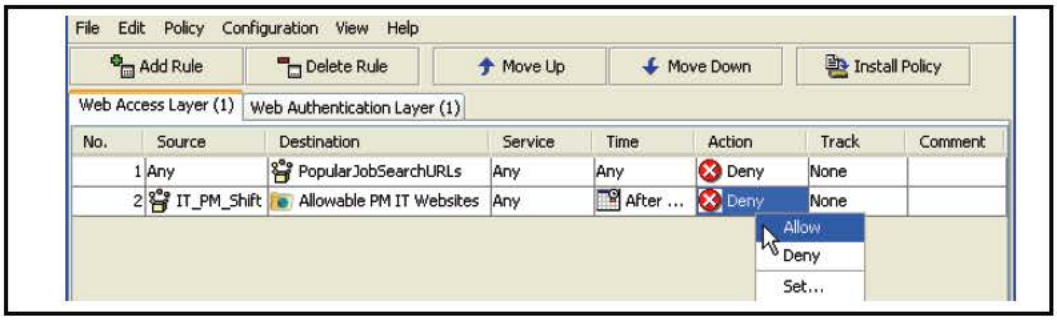


3. Name the object **After Hours**.
4. In the **Specific Time of Day Restriction** field, select **Enable** and set the time from **18:00** to **05:59**.

   This defines after hours as 6:00 PM to 6:00 AM.
5. In the **Specific Weekday Restriction** field, select **Enable** and select **Monday, Tuesday, Wednesday, Thursday,** and **Friday**.

   This defines the days of the week to which this rule applies.
6. Click **OK** twice to add the Time Object to the rule.

To configure the Action object:



1. In the second rule, right-click **Action** and select **Allow**.
2. Click **Install Policy**.

Chapter 4: Advanced Policy Tasks

This chapter provides conceptual and procedural information about the SG appliance advanced policy features. While many SG appliance features have a policy component, some features have no configuration component outside policy. Configuring advanced policy is accomplished by defining rules in the Visual Policy Manager (VPM) or by composing Content Policy Language (CPL). While some examples are provided in this chapter, references to the relevant VPM chapter component are included in each section.

This chapter contains the following topics:

Excluding exceptions, you *must* use policy to implement these capabilities. (For exceptions, you can create a list outside of policy to install on the system.)

Section A: Blocking Pop Up Windows

This section describes the Blue Coat solution for blocking unwanted pop up windows.

About Pop Up Blocking

The SG appliance allows you to block pop up windows, which are usually in the form of unsolicited advertisements. Pop up windows are blocked by inserting Javascript code into each HTML Web page. Every time the Web page tries to open a new window, the code attempts to determine if the window is a result of user click. The window is allowed to open if the SG appliance determines a user clicked a button or link; otherwise, the window does not open.

Interactivity Notes

Because of the dynamic nature of the Web, blocking pop up windows is not a perfect solution. Consider the following caveats before configuring this feature:

- Windows that contain desired or useful information cannot be distinguished from undesired content, such as advertisements.
- If the Web browser caches a page that spawns pop up windows before the blocking policy was installed, pop up ads continue to be served from that page regardless of current policy.
- Animated ads contained within Web pages are not blocked. Commonly seen in scrolling or drop-down form, these are not true pop up windows but are contained within the page. Users who see these ads might believe that pop up window blocking is not implemented.
- Pop up windows that are delivered through HTTPS are not blocked.
- Although the SG appliance request headers instruct a Web server not to use compression, it is possible (though not likely) for a Web server to be configured to send compressed responses anyway. The pop up blocking feature does not work on compressed HTML pages.

Recommendations

- To compensate for limiting factors, administrators and users can override pop up blocking:
  - Administrators—Use the VPM to create policy rules that exempt pop up blocking for specific Web sites and IP address ranges. For example, Blue Coat recommends disabling pop up blocking for your Intranet, which commonly resides on a IP address range.

Figure 4-1. Disabling pop up blocking on the corporate site.

- Users—When a pop up window is blocked, a message is displayed in the status bar:

  ```
  blocked popup window -- use CTRL Refresh to see all popups.
  ```

  While pressing the Control key, click the Web browser **Refresh** button; the page is reloaded with pop up blocking disabled for that action.

- Create a separate Web Access policy layer for pop up blocking actions. This alleviates interference with Web applications deployed on your Intranet that require pop up windows.
- To prevent a cached Web page from spawning pop up windows, clear the browser cache, then reload the page without holding down the `CTRL` key.

Blocking pop up windows is accomplished through the Visual Policy Manager. See "Block/Do Not Block PopUp Ads" on page 90 for information about how to create blocking actions in a policy layers.

Section B: Stripping or Replacing Active Content

This section describes the Blue Coat solution for stripping or replacing unwanted active content.

About Active Content

Scripts activated within Web pages can pose a security concern. The SG appliance policy can be configured to supplement standard virus scanning of Web content by detecting and removing the HTML tags that launch active content such as Java applets or scripts. In addition, the removed content can be replaced with predefined material, a process referred to as *active content transformation*.

When the SG appliance is configured to perform active content transformation, Web pages requested by a client are scanned before they are served and any specified tags and the content they define are either removed or replaced. Because the transformed content is not cached, the transformation process is based on a variety of conditions, including time of day, client identity, or URL.

Note: Pages served over an HTTPS tunneled connection are encrypted, so the content cannot be modified.

The following tags and related content can be removed or replaced:

- `<APPLET>`—Java applets, as defined by HTML `<applet>` elements.
- `<EMBED>`—Embedded multimedia objects displayed using Netscape Navigator plug-ins as defined by HTML `<embed>` elements.
- `<OBJECT>`—Embedded multimedia objects displayed using Internet Explorer ActiveX controls and other multimedia elements, as defined by HTML `<object>` elements
- `<SCRIPT>`—Embedded Javascript and VBScript programs, whether these are represented as HTML `<script>` elements, Javascript entities, Javascript URLs, or event handler attributes. The `<noscript>` tag is *not* affected by this features.

Stripping active content is accomplished through the Visual Policy Manager or by composing CPL.

- See "Strip Active Content" on page 104 for information about how to create a **Strip Active Content** action object in a Web Access policy layer.
- Refer to *Volume 10: Blue Coat SG Appliance Content Policy Language Guide*.

About Active Content Types

The following sections provide more detail about the types of active content that can be removed or replaced.

Script Tags

Scripts are generally placed between the start and end tags `<SCRIPT>` and `</SCRIPT>`. The type of script used is defined by the `LANGUAGE` attribute; for example, `<SCRIPT LANGUAGE="JavaScript 1.0">`). When the `LANGUAGE` attribute is undefined, the browser assumes JavaScript.

When `transform active_content` is configured to remove scripts, the basic operation is to remove all content between and including `<SCRIPT>` and `</SCRIPT>`, regardless of the language type, and substitute any defined replacement text. A notable exception occurs when a script is defined in the header portion of the HTML document (defined by the `<HEAD>` tag). In this case, the script is simply removed. This is because images, objects, and text are not allowed in the header of an HTML document. If the end script tag `</SCRIPT>` is missing from the document (the end of the document is defined as either up to the `</BODY>` or `</HTML>` tag, or the last character of the document), then all content from the start `<SCRIPT>` tag to the end of the document is removed.

JavaScript Entities

JavaScript entities have the following format: `&{javascript code}` and are found anywhere in the value part of an attribute (that is, `<IMG SRC="&{images.logo};"`). You can define more than one entity in the value portion of the attribute. When `transform active_content` is configured to remove scripts, all JavaScript entities attribute/value pairs are removed. No replacement text is put in its place.

JavaScript Strings

JavaScript strings have the following format: `javascript: javascript code` and are found anywhere in the value part of an attribute, though usually only one of them can be defined in an attribute. Most modern browsers support JavaScript strings. When `transform active_content` is configured to remove scripts, all JavaScript string attribute/value pairs are removed. No replacement text is put in its place.

JavaScript Events

JavaScript events are attributes that start with the keyword `on`. For example, `<A HREF="index.html onMouseOver="javascript code">`. The HTML 4.01 specification defines 21 different JavaScript events:

```
onBlur, onChange, onClick, onDblClick, onDragDrop, onFocus, onKeyDown,
onKeyPress, onKeyUp, onLoad, onMouseDown, onMouseMove, onMouseOut,
onMouseOver, onMouseUp, onMove, onReset, OnResize, onSelect, onSubmit,
onUnload
```

Both Microsoft Internet Explorer and Netscape have defined variations on these events as well as many new events. To catch all JavaScript events, the active content transformer identifies any attribute beginning with the keyword `on`, not including `on` itself. For example, the attribute `onDonner` in the tag `<A HREF="index.html" onDONNER="DONNER">` is removed even though `onDonner` does not exist as a valid JavaScript event in the browser. In this case, the transformed file would show `<A HREF="index.html">`.

Embed Tags

HTML `<EMBED>` tags are not required to have an `</EMBED>` end tag. Many Web browsers do, however, support the `<EMBED> </EMBED>` tag pair. The text between the tags is supposed to be rendered by the browsers when there is no support for the embed tag, or if the MIME-type of the embed object is not supported. Thus, when `transform active_content` is configured to transform embed tags, only the `<EMBED>` tag is removed and replaced with any replacement text. Any occurrence of the end tag `</EMBED>` is simply removed, leaving the text between the beginning and end tags intact.

Object Tags

Objects tags have a start `<OBJECT>` and end `</OBJECT>` tag pair, and the attributes `CODETYPE` and `TYPE` determine the type of object. The text between the tags is supposed to be rendered by the browsers when the object tag is not supported, so when `transform active_content` is configured to transform object tags, only the `<OBJECT>` and `</OBJECT>` tags are removed and replaced with any replacement text. The text between the tags remains. The `CODETYPE` or `TYPE` attributes do not affect the transformation. Also, if the end `</OBJECT>` tag is missing, the transformation will not be affected.

Section C: Modifying Headers

The request headers are sent when users access Web objects that contain a lot of information. This can raise a concern that such details compromise the privacy or security of the enterprise or user.

When a user clicks on a link, the Web browser sets the request's Referer header to the URL of the Web page that contained the link. (This header is not set if the URL was entered or selected from a favorites or bookmarks list.) If an internal Web page provides links to external Web sites, users clicking those links sends the URL of the internal pages, and are logged in the Web logs of those external sites. This is not usually an issue; however, if the external Web site is a competitor Web site or another site with interest in the internal details of your enterprise, this might be a concern.

For example, how you structure your intranet might suggest something about your company's current or future direction. Certain project names or codewords might show up in directory or file names. Exposing the structure of the intranet makes it easier for hackers to attack the network.

The broad solution of deleting Referer headers from all requests presents a problem because some Web sites do not serve images or other linked objects unless the Referer header is set to a referring page on that same Web site. The solution implemented by Blue Coat is to strip the Referer header only when the target Web page resides on the Internet and the referring page is on an internal host.

Suppressing headers is accomplished through the Visual Policy Manager or by composing CPL.

- See "Suppress Header" on page 99 for information about how to create a **Suppress Header** action object in a Web Access policy layer.
- Refer to *Volume 10: Blue Coat SG Appliance Content Policy Language Guide*.

Section D: Defining Exceptions

Exceptions are sent in response to certain SG appliance client requests, such as denial by policy, failure to handle the request, and authentication failure. Exceptions are returned to users based on policy rules defined by the SG appliance administrator. For example, if a client sends a request for content that is not allowed, an exception HTML page (for HTTP connections) or an exceptions string (for non-HTTP connections) is returned, informing the client that access is denied.

Two types of exceptions are used: built-in and user-defined.

Built-in Exceptions

Built-in exceptions are a set of pre-defined exceptions contained on the SG appliance. Built-in exceptions send information back to the user under operational contexts that are known to occur, such as *policy_denied* or *invalid_request*.

Built-in exceptions are always available and can also have their contents customized; however, built-in exceptions cannot be deleted, and you cannot create new built-in exceptions.

The table below lists the built-in exceptions and the context under which they are issued.

Table 4-1. Exceptions

Exception Type	Issued When
`authentication_failed`	The transaction cannot be authenticated, usually because the credentials were incorrect. `authentication_failed` is a synonym for `deny.unauthorized`.
`authentication_failed_password_expired`	The authentication server reports that the credentials provided have expired, and a new password must be obtained.
`authentication_mode_not_supported`	The configured authentication mode is not supported for the current request.
`authentication_redirect_from_virtual_host`	Transparent redirect authentication is being used. This exception redirects the transaction from the virtual authentication host to the original request.
`authentication_redirect_off_box`	The request is being redirected to an authentication service on another device.
`authentication_redirect_to_virtual_host`	Transparent redirect authentication is being used. This exception redirects the transaction to the virtual authentication host.
`authentication_success`	Transparent redirect authentication with cookies is being used. This exception redirects the transaction to the original request, but removes the authentication cookie form the request URL.

Section D: Defining Exceptions

Table 4-1. Exceptions (Continued)

Exception Type	Issued When
`authorization_failed`	The `deny.unauthorized` policy action is matched. This exception notifies the user that their currently authenticated identity is not permitted to perform the requested operation, but they might have some other credentials that would allow their request through (for example. they get an opportunity to enter new credentials).
`client_failure_limit_exceeded`	Too many requests from your ip address (`$(client.address)`) have failed.
`configuration_error`	A configuration error on the SG appliance was detected, and the requested operation could not be handled because of the configuration error. This exception is a likely indicator that the administrator of the SG appliance must intervene to resolve the problem.
`connect_method_denied`	A user attempted an CONNECT method to a non-standard port when explicitly proxied. Blue Coat does not allow CONNECT methods to non-standard ports by default because it is considered a security risk to do so.
`content_filter_denied`	A particular request is not permitted because of its content categorization.
`content_filter_unavailable`	An external content-filtering service could not be contacted, and the SG appliance is failing closed in such a situation.
`dns_server_failure`	The request could not be processed because the SG appliance was unable to communicate with the DNS server in order to resolve the destination address of the request.
`dns_unresolved_hostname`	The request could not be processed because the SG appliance was unable to resolve the hostname in the request with DNS.
`dynamic_bypass_reload`	The `dynamic_bypass` policy action is matched.
`gateway_error`	There was a network error while attempting to communicate with the upstream gateway.
`icap_communication_error`	A network error occurred while the SG appliance was attempting to communicate with an external ICAP server.
`internal_error`	The SG appliance encountered an unexpected error that resulted in the inability to handle the current transaction.
`invalid_auth_form`	The submitted authentication form is invalid. The form data must contain the username, password, and valid original request information.

Section D: Defining Exceptions

Table 4-1. Exceptions (Continued)

Exception Type	Issued When
`invalid_request`	The request received by the SG appliance was unable to handle the request because it detected that there was something fundamentally wrong with the syntax of the request.
`license_expired`	The requested operation cannot proceed because it would require the usage of an unlicensed feature.
`method_denied`	The requested operation utilizes a method that has been explicitly denied because of the service properties associated with the request.
`not_implemented`	The protocol cannot handle the requested operation because it utilizes a feature that is not currently implemented.
`notify`	Used internally by VPM. You do not need to customize the text of this exception, since in this case the entire HTML response is generated by VPM and is not taken from the exception definition.
`notify_missing_cookie`	This exception is returned when a VPM N**otify User** action is being used to notify the user, and the user has disabled cookies in the Web browser.
`policy_denied`	`policy_denied` is a synonym for `deny`.
`policy_redirect`	A `redirect` action is matched in policy.
`redirected_stored_requests_ not_supported`	This applies to forms authentication with POST requests only): The origin server returned a redirect for the request. The SG appliance is configured to not allow stored requests to be redirected.
`refresh`	A refresh (using the `HTTP Refresh: header`) is required. The refresh exception (by default) refreshes the originally requested URL (or in some cases, its post-imputed form).
`server_request_limit_ exceeded`	Too many simultaneous requests are in progress to $(url.host).
`silent_denied`	An `exception(silent_denied)` is matched in policy. This exception is pre-defined to have no body text, and is *silent* in that it results in only the status code being sent to the client.
`ssl_domain_invalid`	There was a failure contacting an upstream host through HTTPS because the certificate presented by the upstream host was either the incorrect one or invalid.
`ssl_failed`	A secure connection could not be established to an upstream host. This is typically because the upstream host is not configured to accept SSL connections.

Section D: Defining Exceptions

Table 4-1. Exceptions (Continued)

Exception Type	Issued When
`tcp_error`	A network error occurred attempting to communicate with an upstream host.
`transformation_error`	The server sends an unknown encoding and the SG appliance is configured to do content transformation.
`unsupported_encoding`	The client makes a request with an `Accept-Encoding: Identity;q=0, ...` header. Only uncompressed content is available in cache, the SG appliance is not configured to compress the content, or the compression license is expired, or the client request results in to `Accept-Encoding: Identity;q=0` because of the combination of request and configured policy.
`unsupported_protocol`	The protocol used in the request is not understood.

Most of the above exceptions can be initiated directly through the policy exception property. However, some require additional state that makes initiating them either problematic or out of context. The following are exceptions that cannot be initiated through the exception property:

- `authentication_failed`
- `authentication_failed_password_expired`
- `authentication_redirect_from_virtual_host`
- `authentication_redirect_to_virtual_host`
- `authentication_success`
- `dynamic_bypass_reload`
- `license_expired`
- `ssl_domain_invalid`
- `ssl_failed`

To view the content of a built-in exception, enter the following commands at the `(config)` prompt:

```
SGOS#(config) exceptions
SGOS#(config exceptions) show exceptions configuration_error
configuration_error exception:
all protocols:
summary text:
         SG configuration error
details text:
         Your request could not be processed because of a configuration
error: $(exception.last_error)
help text:
         The problem is most likely because of a configuration error,
$(exception.contact) and provide them with any pertinent information
from this message.
http protocol:
         code: 403
```

User-Defined Exceptions

User-defined exceptions are created and deleted by the administrator. If a user-defined exception is referenced by policy, it cannot be deleted. The default HTTP response code for user-defined exceptions is 403.

Note: For users who are explicitly proxied and use Internet Explorer to request an HTTPS URL, an exception body longer than 900 characters might be truncated. The workaround is to shorten the exception body.

An exception body less than 512 characters might cause a *page does not exist* 404 error. If this occurs, use the `exception.autopad(yes|no)` property to pad the body to more than 513 characters. For more information on the `exception.autopad` property, refer to the *Volume 10: Blue Coat SG Appliance Content Policy Language Guide*.

About Exception Definitions

Each exception definition (whether built-in or user-defined) contains the following elements:

- `Identifier`—Identifies the type of exception. Table 4-1 lists the built-in exception types. For user-defined exceptions, the identifier is the name specified upon creation.
- `Format`—Defines the appearance of the exception. For an HTTP exception response, the format is an HTML file. For other protocols, where the user agents are not able to render HTML, the format is commonly a single line.
- `Summary`—A short description of the exception that labels the exception cause. For example, the default `policy_denied` exception summary is `"Access Denied"`.
- `Details`—The default text that describes reason for displaying the exception. For example, the default `policy_denied` exception (for the HTTP protocol) detail is: Your request has been denied by system policy.
- `Help`—An informative description of common possible causes and potential solutions for users to take. For example, if you want the categorization of a URL reviewed, you can append the `$(exception.category_review_url)` and `$(exception.category_review_message)` substitutions to the `$(exception.help)` definition. You must first enable this capability through content filtering configuration. For information on enabling review categorization, refer to *Volume 7: Managing Content*.
- `Contact`—Used to configure site-specific contact information that can be substituted in all exceptions. Although it is possible to customize contact information on a per-exception basis, customizing the top-level contact information, which is used for all exceptions, is sufficient in most environments.
- `HTTP-Code`—The HTTP response code to use when the exception is issued. For example, the `policy_denied` exception by default returns the `403 Forbidden` HTTP response code.

Important: Fields other than Format must be less than 8000 characters. If they are greater than this, they are not displayed.

When defining the above fields, you can use substitution variables that are particular to the given request. Some of the above fields are also available as substitutions:

- `$(exception.id)`
- `$(exception.summary)`
- `$(exception.details)`
- `$(exception.help)`
- `$(exception.contact)`

Additionally, the `Format`, `Summary`, `Details`, `Help` and `Contact` fields can be configured specifically for HTTP, or configured commonly for all protocols.

The `Format` field, the body of the exception, is not available as a substitution. However, the Format field usually includes other substitutions. For example, the following is a simple HTML format:

```
<html>
<title>$(exception.id): $(exception.summary)</title>
<body><pre>
Request: $(method) $(url)
Details: $(exception.details)
Help: $(exception.help)
Contact: $(exception.contact)
</pre></body></html>
```

Some additionally useful substitutions related to exceptions are:

- `$(exception.last_error)`—For certain requests, the SG appliance determines additional details on why the exception was issued. This substitution includes that extra information.
- `$(exception.reason)`—This substitution is determined internally by the SG appliance when it terminates a transaction and indicates the reason that the transaction was terminated. For example, a transaction that matches a DENY rule in policy has its `$(exception.reason)` set to `"Either 'deny' or 'exception' was matched in policy"`.

About the Exceptions Hierarchy

Unlike the error pages in previous SGOS releases, exceptions are not required to have its entire contents defined. Exceptions are stored in a hierarchical model, and *parent* exceptions can provide default values for *child* exceptions. There are two parent exceptions from which other exceptions are derived: `exception.all` and `exception.user-defined.all`.

Each built-in and user-defined exception derives its default values from the `all` exception. For example, by default the built-in exceptions do not define the `format` field. Instead, they depend on the `all` exception's `format` field definition. To change the format text for all built-in and user-defined exceptions, customize the `format` field for the `all` exception.

The `user-defined.all` exception is the parent of all user-defined exceptions, but it is also a child of the `all` exception. Configuring `exception.user-defined.all` is only necessary if you want certain fields to be common for all user-defined exceptions, but not common for built-in exceptions.

The following example demonstrates using the exception inline command to configure the `$(exception.contact)` substitution for every HTTP exception:

```
#(config exceptions) inline http contact EOF
For assistance, contact <a
href="mailto:sysadmin@example.com">sysadmin</a>EOF
```

The following example configures a different `$(exception.contact)` substitution for every HTTP exception:

```
#(config exceptions) user-defined inline http contact EOF
For assistance, contact <a
href="mailto:policyadmin@example.com">policyadmin</a>EOF
```

About the Exceptions Installable List

The Exceptions Installable List uses the Structured Data Language (SDL) format. This format provides an effective method to express a hierarchy of key/value pairs. For example, the following is SDL file before customization:

```
(exception.all
  (format "This is an exception: $(exception.details)")
  (details "")
  (exception.policy_denied
  (format "")
  (details "your request has been denied by system policy")
)
```

This SDL file defines an exception called `policy_denied` that defines the `$(exception.details)` substitution as `"Your request has been denied by system policy"`. Because the exception does not define the `format` field, it inherits the format field from its parent exception (`exception.all`). When the `policy_denied` exception is issued, the resulting text is: This `is an exception: your request has been denied by system policy`.

Suppose you want to customize the `$(exception.contact)` substitution for every HTTP exception. Edit the `exception.all` component.

Note: The default HTTP format and built-in exception definitions have been removed for example purposes.

```
(exception.all
   (contact "For assistance, contact your network support team.")
   (details "")
   (format "$(exception.id): $(exception.details)")
   (help "")
   (summary "")
   (http
     (code "200")
     (contact "")
     (details "")
     (format <<EOF
<format removed>
   EOF
   )
   (help "")
   (summary "")
   )
<built-in exceptions removed>
   )
```

To add the `$(exception.contact)` information, modify the `contact` substitution under the `http` node:

```
(exception.all
   (contact "For assistance, contact your network support team.")
   (details "")
   (format "$(exception.id): $(exception.details)")
   (help "")
   (summary "")
   (http
     (code "200")
     (contact "For assistance, contact <a
      href="mailto:sysadmin@example.com">sysadmin</a>")EOF
     (details "")
     (format <<EOF
<format removed>
   EOF
      )
      (help "")
      (summary "")
<built-in exceptions removed>
      )
   )
```

Keep in mind the following conditions when modifying exception installable lists:

- Every exception installable list must begin with a definition for `exception.all`.
- In the exceptions' installable list, all definitions must be enclosed by `exception.all` and its accompanying closing parenthesis; that is,
  ```
  (exception.all
  (exception.policy_denied)
  )
  ```
- Keep the definition strings under the enclosed parentheses short, no longer than one line if possible.
- Blue Coat strongly recommends downloading the existing exceptions installable list, then modifying it.

Creating or Editing Exceptions

You can create or edit an exception with the CLI or with installable lists on the Management Console.

Note: You cannot create user-defined exceptions for Patience Pages.

To create or edit an exception:

1. At the `(config)` prompt, enter the following commands:

   ```
   SGOS#(config) exceptions
   SGOS#(config exceptions) create definition_name
   SGOS#(config exceptions) edit definition_name
   SGOS#(config exceptions user-defined.definition_name) http-code
   numeric HTTP
   response code
   SGOS#(config exceptions user-defined.definition_name) inline ?
   ```

```
  contact     Set the $(exceptions.contact) substitution
  details     Set the $(exceptions.details) substitution
  format       Set the format for this exception
  help           Set the $(exceptions.help) substitution
  http           Configure substitution fields for just HTTP exceptions
  summary     Set the $(exception.summary) substitution
SGOS#(config exceptions user-defined.definition_name) inline contact
eof
string eof
SGOS#(config exceptions user-defined.definition_name) inline details
eof
string eof
SGOS#(config exceptions user-defined.definition_name) inline format
eof
string eof
SGOS#(config exceptions user-defined.definition_name) inline help eof
string eof
SGOS#(config exceptions user-defined definition_name) inline summary
eof
string eof
```

2. (Optional) View the results.

```
SGOS#(config exceptions user-defined.test) show exceptions user-
defined.test
$(exception.id):
  test
$(exception.summary):
  Connection failed
$(exception.details):
  Connection failed with stack error
$(exception.contact):
  Tech Support
```

To delete a user-defined exception:

From the `(config)` prompt, enter the following commands:

```
SGOS#(config) exceptions
SGOS#(config exceptions) delete exception_name
  ok
```

> **Note:** You cannot delete a user-defined exception that is referenced by policy. You must remove the reference to the exception from the policy before deleting the exception.

Creating and Installing an Exceptions List

The Management Console allows you to create and install exceptions with the following methods:

- Using the SG appliance Text Editor, which allows you to customize the existing exceptions file.
- Creating a local file on your local system; the SG appliance can browse to the already-created file and install it.

Section D: Defining Exceptions

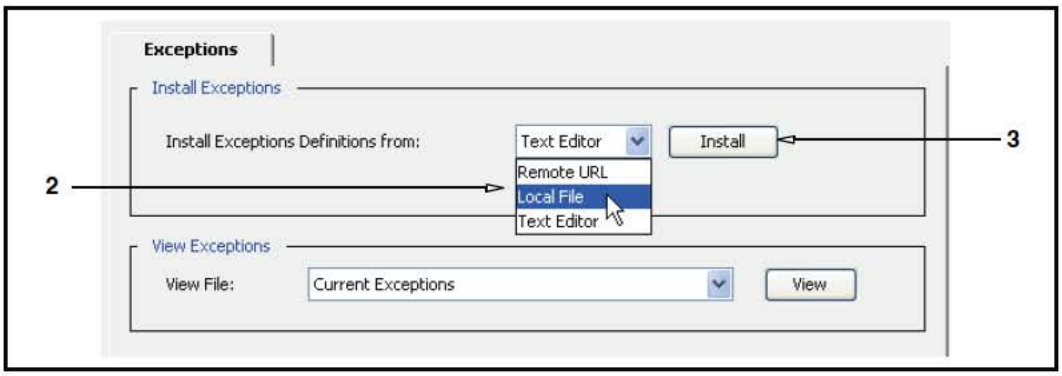
- Using a remote URL, where you place an already-created exceptions list on an FTP or HTTP server to be downloaded to the SG appliance.

Note: A message is written to the event log when you install a list through the SG appliance.

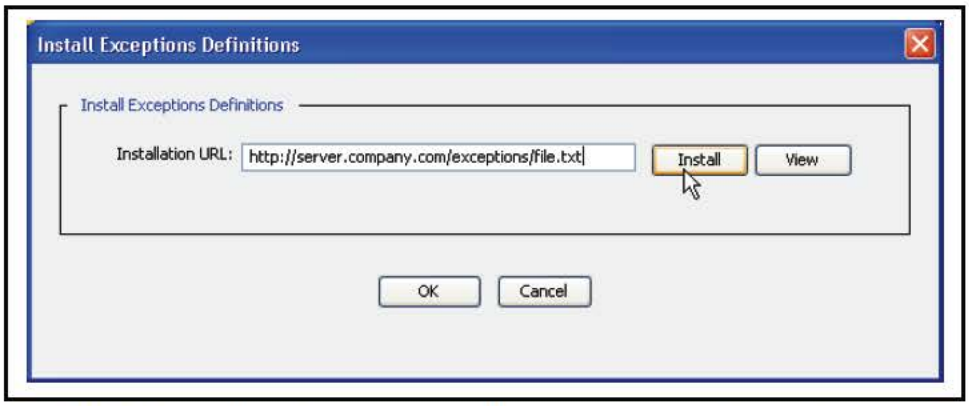
When the Exceptions file is customized, it updates the existing exceptions already on the SG appliance. The configuration remains in effect until it is overwritten by another update; it can be modified or overwritten using CLI commands.

To install an exceptions definition:

1. Select **Configuration > Policy > Exceptions**.

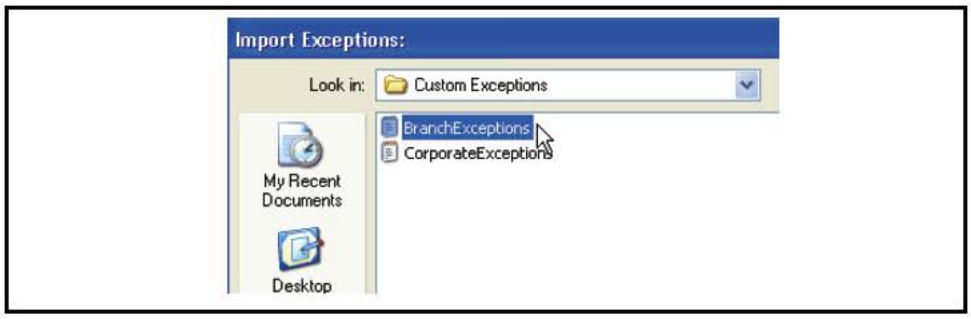


2. From the **Install Exceptions Definitions From** drop-down list, select the method used to install the exceptions configuration.
3. Click **Install**.
   - Installing from a **Remote URL**:



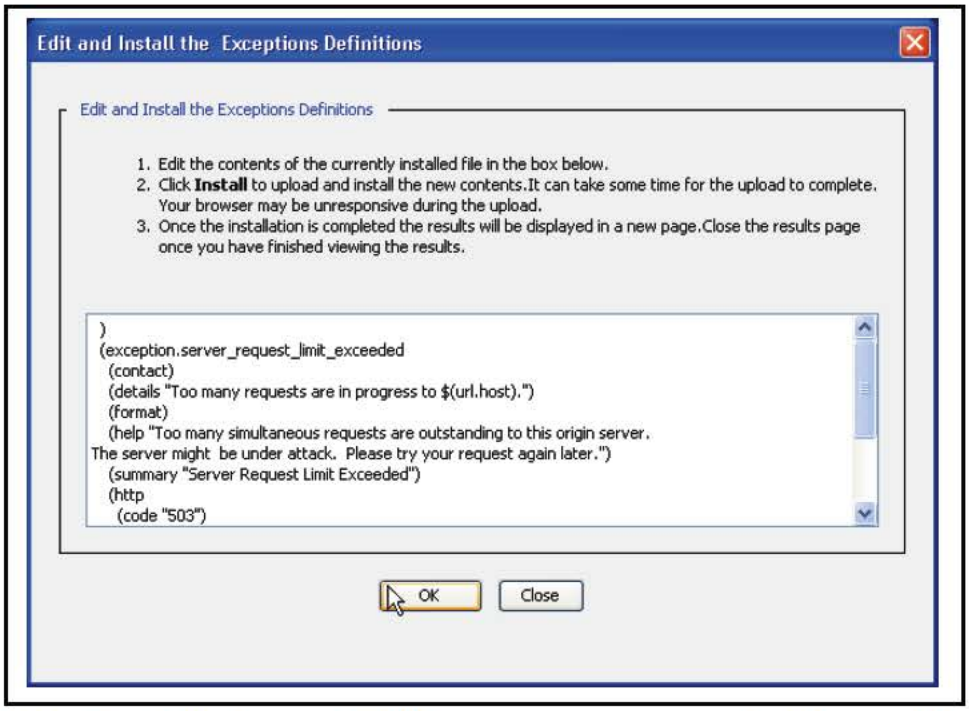
   Enter the fully-qualified URL, including the filename, where the configuration is located. To view the file before installing it, click **View**. Click **Install**. View the installation status; click **OK**.

   - Installing by browsing to a **Local File**: Click **Browse** to bring up the Local File Browse window.



Browse for the file on the local system. Open it and click **Install**. When the installation is complete, a results window opens. View the results, close the window, and click **Close**.

- Installing a policy file using the SG appliance **Text Editor**:



In Structured Data Language (SDL) format, create a custom policy to be installed (added to the existing exceptions file).

4. Click **OK**.

Viewing Exceptions

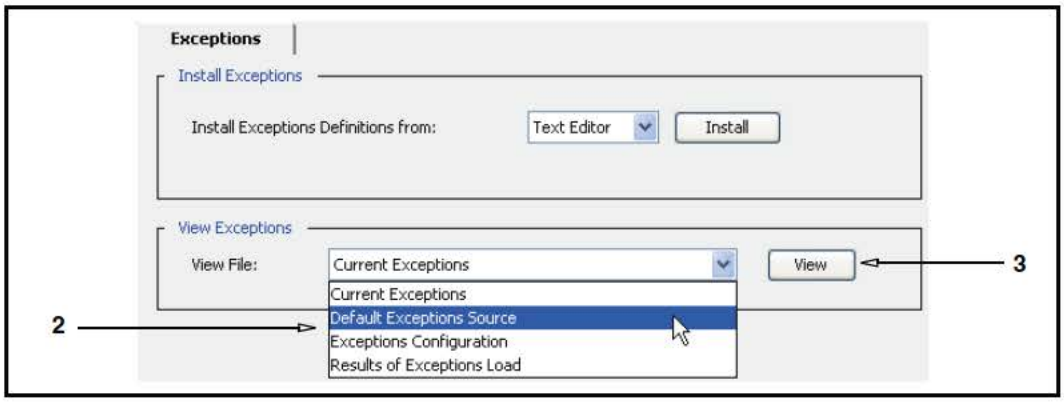
You can view the exceptions defined on the SG appliance, including how the defined HTML appears to users. The following are the viewable defined exception components:

- **Current Exceptions**—Displays all of the exceptions as they are currently defined.
- **Default Exceptions Source**—Displays the default SG appliance exceptions.
- **Exceptions Configuration**—Displays a page from which you can click links to view how exceptions appear in HTML to users.

- **Results of Exception Load**—Displays the results of the last installable list load, including any errors and warning to be fixed.

To view existing exceptions:

1. Select **Configuration > Policy > Exceptions**.



2. From the **View Exceptions** field, **View File** drop-down list, select the page to view.
   - **Current Exceptions**—Displays all of the exceptions as they are currently defined.
   - **Default Exceptions Source**—Displays the default SG appliance exceptions.
   - **Exceptions Configuration**—Displays a page from which you can click links to view how exceptions appear in HTML to users.
   - **Results of Exception Load**—Displays the results of the last installable list load, including any errors and warning to be fixed.
3. Click **View**. An new browser appears with the current requested information.
4. Click **Apply**.

Section E: Managing Peer-to-Peer Services

This section describes the Blue Coat solution for managing and blocking peer-to-peer traffic.

About Peer-to-Peer Communications

The use of peer-to-peer (P2P) technologies and services consumes an estimated 60% of broadband ISP bandwidth. By design, most P2P services are port-agnostic, which makes attempting to block them at the firewall extremely difficult. One peer finds another IP address and port that is willing to share the file, but different peers can use different ports. Furthermore, P2P is not based on any standards, which makes it nearly impossible for network administrations to control or even detect.

Although P2P provides some practical business uses in enterprises, unmanaged P2P activity creates risks:

- Excessive bandwidth consumptions affects mission-critical applications.
- Exponential security risk of exposure to viruses, spyware, and other malicious content.
- The threat of legal action concerning the unlawful downloading of copyrighted music and movies.

Managing P2P is a dynamic challenge, as the administrator must be able to evaluate both P2P use and enterprise requirements.

About The Blue Coat Solution

The SG appliance recognizes P2P activity relating to P2P file sharing applications. By constructing policy, you can control, block, and log P2P activity and limit the bandwidth consumed by P2P traffic.

Note: Neither caching nor acceleration are provided with this feature.

Supported Services

This version of SGOS supports the following P2P services:

- FastTrack (Kazaa)
- EDonkey
- BitTorrent
- Gnutella

Note: Refer to the Release Notes for the most current list of P2P services and versions the SG appliance supports.

Deployment

To effectively manage P2P activity, the SG appliance must be deployed to intercept outbound network traffic and the firewall configured to block outbound connections that are *not* initiated by the SG appliance.

Notes:

- The SG appliance intercepts outbound TCP network connections, as routed through an L4 switch or a SG appliance in bridging mode.
- Configure SG appliance HTTP, SOCKS, and TCP tunnel services for destination ports to be monitored.
- Create firewall rules that allow only outbound connections that are initiated by the SG appliance.
- You can block all known P2P ports and define policy to stop P2P traffic attempting to come through over HTTP

Note: This features does not include additional configurations for intercepting or controlling UDP traffic.

Policy Control

This section lists the policy used to manage P2P.

VPM Support

The following VPM components relate to P2P control:

Figure 4-2. Web Access Layer Rule with P2P Objects

- Web Access Layer; Source column; P2P Client object. See "P2P Client" on page 62.
- Web Access Layer, Service column; Client Protocols. See "Client Protocol" on page 76.

CPL Support

CPL Triggers

- `http.connect={yes | no}`
- `p2p.client={yes | no | bittorrent | edonkey | fasttrack | gnutella}`

CPL Properties

- `force_protocol()`
- `detect_protocol.`*`protocol`*`(yes | no)`
- `detect_protocol.[`*`protocol1, protocol2, ...`*`](yes | no)`
- `detect_protocol(all | none)`
- `detect_protocol(`*`protocol1, protocol2, ...`*`)`

  Where protocol is: `http`, `bittorrent`, `edonkey`, `fasttrack`, or `gnutella`.

The default is `detect_protocol(all)`.

Support CPL

The following properties can be used in conjunction with the P2P-specific CPL:

- `allow`, `deny`, `force_deny`
- `access_server(yes | no)`—If the value is determined as no, the client is disconnected.
- `authenticate(realm)`—Unauthenticated clients are disconnected.
- `socks_gateway(alias_list | no)`
- `socks_gateway.fail_open(yes | no)`
- `forward(alias_list) | no)`—Only forwarding hosts currently supported by TCP tunnels are supported.
- `forward.fail_open(yes | no)`
- `reflect_ip(auto | no | client | vip | ip_address)`

For complete CPL references, refer to *Volume 10: Blue Coat SG Appliance Content Policy Language Guide*.

Policy Example

The following policy example demonstrates how to deny network traffic that the SG appliance recognizes as P2P:

```
<proxy>
  p2p.client=yes deny
```

P2P History Statistics

You can construct policy that controls, blocks, and logs peer-to-peer (P2P) activity and limits the bandwidth consumed by P2P traffic (refer to *Volume 6: VPM and Advanced Policy* for information about constructing P2P policy). The following section explains how to view P2P statistics, using either the Management Console or the CLI.

Note: Some P2P statistics (P2P client connections and total bytes sent and received over a period of time) can only be viewed through the Management Console (see "P2P Clients" and "P2P Bytes", below).

P2P Data

The P2P Data tab on the Management Console displays P2P statistics, either all P2P services at once or one service at a time.

The following table details the statistics provided through the Management Console P2P Data tab or through the CLI

Table 4-2. P2P Data Statistics

Status	Description
Current Tunneled Sessions	The current number of P2P client connections using native transport.

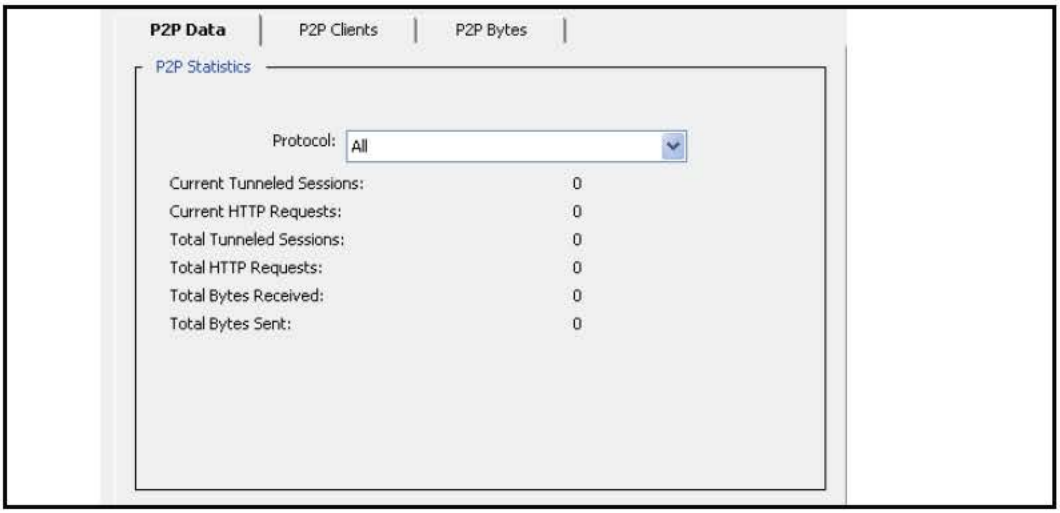
Table 4-2. P2P Data Statistics (Continued)

Current HTTP Requests	The current number of HTTP requests from P2P clients.
Total Tunneled Sessions	The cumulative number of P2P client connections using native transport since the SG appliance was last rebooted.
Total HTTP Requests	The cumulative number of HTTP requests from P2P clients since the SG appliance was last rebooted.
Total Bytes Received	The total number of bytes received from all P2P clients.
Total Bytes Sent	The total number of bytes sent to all P2P clients.

To view P2P data statistics:

1. Select **Statistics > Protocol Details > P2P History > P2P Data**.

   The default view shows all P2P protocols.



2. (Optional) To view the statistics for a specific P2P protocol, make a selection from the **Protocol** drop-down list.

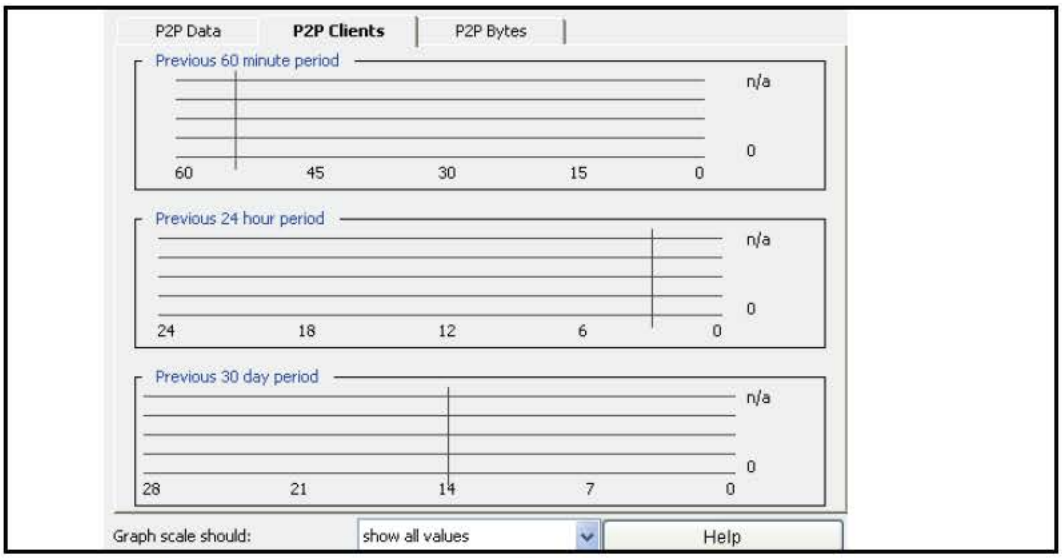
P2P Clients

The P2P Clients tab displays dynamic graphical statistics for client connections received in the last 60-minute, 24-hour, or 30-day period.

Note: The P2P client statistics are available only through the Management Console.

To view P2P client statistics:

1. Select **Statistics > Protocol Details > P2P History > P2P Clients**.



2. (Optional) To set the graph scale to a different value, select a value from the **Graph scale should** drop-down list.

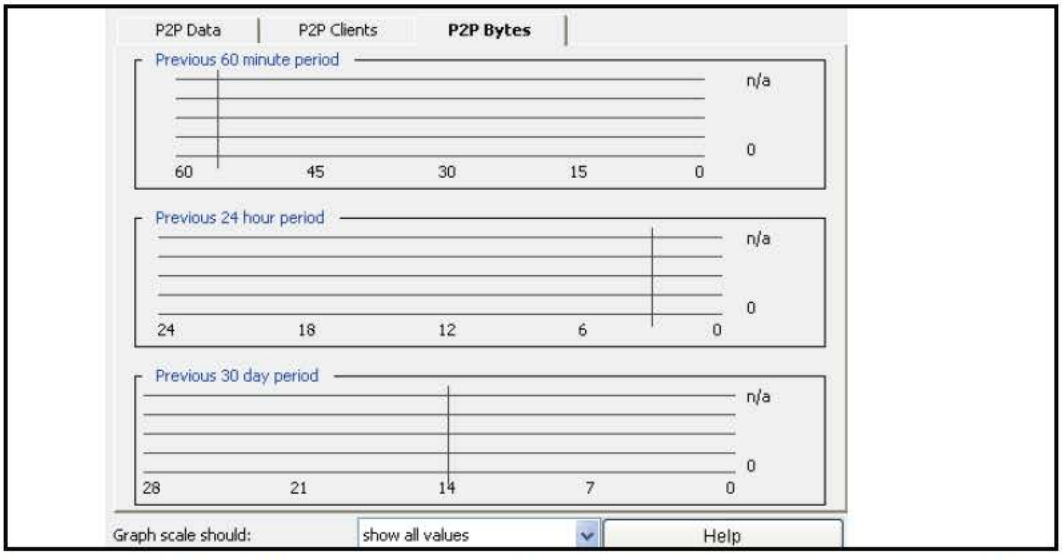
P2P Bytes

The P2P Bytes tab displays dynamic graphical statistics for the total number of bytes sent to and received from P2P clients in the last 60-minute, 24-hour, or 30-day period.

Note: The P2P bytes statistics are available only through the Management Console.

To view P2P byte statistics:

1. Select **Statistics > Protocol Details > P2P History > P2P Bytes**.



2. (Optional) To set the graph scale to a different value, select a value from the **Graph scale should** drop-down list.

Proxy Authentication

While P2P protocols do not support native proxy authentication, most P2P clients support SOCKS v5 and HTTP 1.1 proxies. P2P proxy authentication is supported only for clients using these protocols (that are configured for proxy authentication).

For information about proxy authentication, refer to *Volume 4: Securing the Blue Coat SG Appliance*. For a list of P2P clients suspected of not supporting SOCKS v5 with authentication, see the Release Notes for this release.

Access Logging

P2P activity is logged and reviewable. Refer to *Volume 8: Access Logging*.

Section F: Managing QoS and Differential Services

This section describes how to create policy to manipulate Quality of Service (QoS) information.

About The Blue Coat Solution

Beginning with SGOS 5.1.3, the SG appliance supports QoS detection, which is becoming a more prevalent control point for network layer traffic. Previously, the QoS information was *lost*—or not detected—when the SG appliance terminated the client connection and issued a new connection to server. QoS support allows you to create policy to examine the Type of Service (ToS) fields in the IP header to determine the QoS of the bits. The policy then either tests and matches ToS information and performs an action, or performs an action to manipulate ToS information based on something else in the rule (such as a user group).

You can apply QoS policy to any protocol supported on the SG appliance.

About DSCP Values

Policy matches are based on Differentiated Services Code Point (DSCP) values, which network devices use to identify traffic to be handled with higher or lower priority. Identifying and matching values might trigger defined policy actions that either set a different DSCP value or *preserve* or *echo* existing DSCP values to use for outbound connections, thus regulating the QoS for different user classes (see descriptions in subsequent sections).

Note: The SG appliance policy *requests* a QoS level. Whether or not a level of QoS can be achieved depends upon your network/router configurations, which must also allow the level of requested QoS.

ToS is an eight-bit field in the IP header; the first six bits are used and the final two are reserved for other TCP specification and control. The first six bits constitute the DSCP value. For most networks, the DSCP values adhere to a standard set. The following table lists these values.

Table 4-3. DSCP Values and Descriptions

Name	DCSP Value	Description
Default	000000 (0)	Best effort (Precedence 0)
CS1	001000 (8)	Precedence 1
AF11	001010 (10)	Assured Forwarding Class 1, Low Drop Rate
AF12	001100 (12)	Assured Forwarding Class 1, Medium Drop Rate
AF13	001110 (14)	Assured Forwarding Class 1, High Drop Rate
CS2	010000 (16)	Precedence 2
AF21	010010 (18)	Assured Forwarding Class 2, Low Drop Rate
AF22	010100 (20)	Assured Forwarding Class 2, Low Drop Rate

Table 4-3. DSCP Values and Descriptions (Continued)

AF23	010110 (22)	Assured Forwarding Class 2, Low Drop Rate
CS3	011000 (24)	Precedence 3
AF31	011010 (26)	Assured Forwarding Class 3, Low Drop Rate
AF32	011100 (28)	Assured Forwarding Class 3, Medium Drop Rate
AF33	011110 (30)	Assured Forwarding Class 3, High Drop Rate
CS4	100000 (32)	Precedence 4
AF41	100010 (34)	Assured Forwarding Class 4, Low Drop Rate
AF42	100100 (36)	Assured Forwarding Class 4, Medium Drop Rate
AF43	100110 (38)	Assured Forwarding Class 4, High Drop Rate
CS5	101000 (40)	Precedence 5
EF	101110 (46)	Expedited Forwarding—low drop rate, low latency
CS6	110000 (48)	Precedence 6
CS7	111000 (56)	Precedence 7

Note: Before creating policy, verify that your network adheres to these values. Other DSCP values are possible. You can specify a numerical range from 0 to 63. However, Blue Coat recommends using the above classifications, as most applications are associated to these classes already, which makes defining policy an easier task.

The conceptual definitions of the different classes are:

- Best Effort—This is the default DSCP value if an application does not specify any quality of service. The network delivers these packets if it can, but with no special assigned priority. You can use other DSCP values to specify priorities that are either above or below the Best Effort class; however, in most cases DSCP is used to specify priorities that are better than Best Effort.
- Class Selector—These values are defined in RFC 2474 and are designed to be backward compatible with the older **Precedence** field defined in RFC 791. Larger precedence values indicate packets that are more important than packets with smaller values of precedence; therefore, low-valued packets are dropped when a link becomes congested. Most common, Precedence 7 is reserved for link-layer and routing protocol keep-alive messages, and precedence 6 is reserved for other IP routing packets, both of which must get through for the network to function correctly.
- Assured Forwarding—This is defined in RFC 2597. Assured Forwarding (AF) allows you to specify both the relative priority and the drop sensitivity of traffic with a Precendence class. For example, AF31 specifies low drop-rate with in the CS3 Precedence class.
- Expedited Forwarding—This is defined in RFC 2598. Expedited Forwarding (EF) is usually reserved for premium traffic, or traffic that requires a *virtual leased line*. This traffic is higher priority than AF, but lower priority than precedence 6 and 7 routing messages.

About QoS Policy Tasks

This section describes what is achievable through QoS policy and provides basic examples.

Testing Incoming QoS

Policy triggers test the incoming packets of a client request or a server response. After the SG appliance identifies the DSCP value, other policy in the rule dictates what, or if, any action is required. A common scenario is to create several bandwidth classes (**Configure > Bandwidth Mgmt > BWM Classes**) and allow the DSCP value to dictate which bandwidth applies to the transaction.

Example Policy

Three client connection DSCP Source objects associated with three bandwidth management level Action objects.

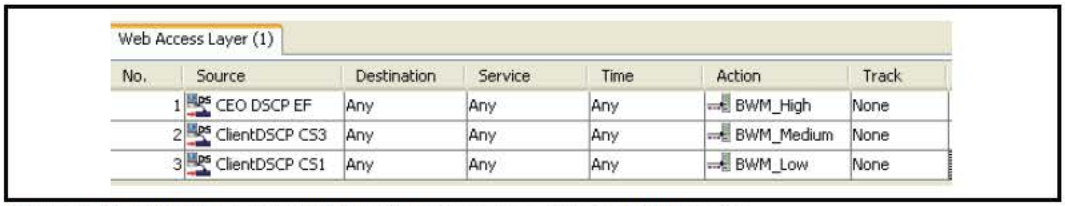


Figure 4-3. A VPM example that tests QoS and assigns a BWM action

The above VPM example generates the following CPL:

```
<Proxy>
  client.connection.dscp=(ef) limit_bandwidth.client.outbound(High)
  client.connection.dscp=(cs3,af31,af32,af33)
limit_bandwidth.client.outbound(Medium)
  client.connection.dscp=(cs1) limit_bandwidth.client.outbound(Low)
```

Caching Behavior

Detecting the QoS cannot occur for cached content. In the case of a cache hit, when no server connection is established, no server connection DSCP value is available for policy checks.

Multiple Connections

Some services use multiple client to server connections. When a service uses multiple connections, the triggers to test the inbound DSCP value apply to the primary control connection, which is (usually) the first connection opened by the client and the corresponding connection (if any) opened to the server. For example:

- FTP connections are comprised of a control connection and a data connection.
- IM connections involve connections to other hosts, such as chat buddies or file sharing hosts.

Setting the Outgoing QoS

You can create policy to preserve, echo, or set the DSCP value.

Preserving the DSCP Value

This is the default SG appliance policy. Using the SG appliance as the frame of reference, the Preserve property instructs the SG appliance to preserve the incoming client DSCP values, on a per-packet basis, when making an outbound server connection and preserve the inbound server values when sending traffic back to the client.

Preserving is valuable for protocols that have multiple connections. For example, FTP connections consist of a control and a data connection; the independent connections might have a differing DSCP values. Preserving the FTP connections prevents the SG appliance from altering one or both of the connections and disrupting the FTP protocol transmission.

While the default policy of preserving the QoS level passes traffic through without any adjustments to QoS, this behavior is different than pre-SGOS 5.1.3 behavior in which QoS data was lost at the point where the SG appliance intercepted the traffic. The preserve property allows for the monitoring of QoS-related network information.

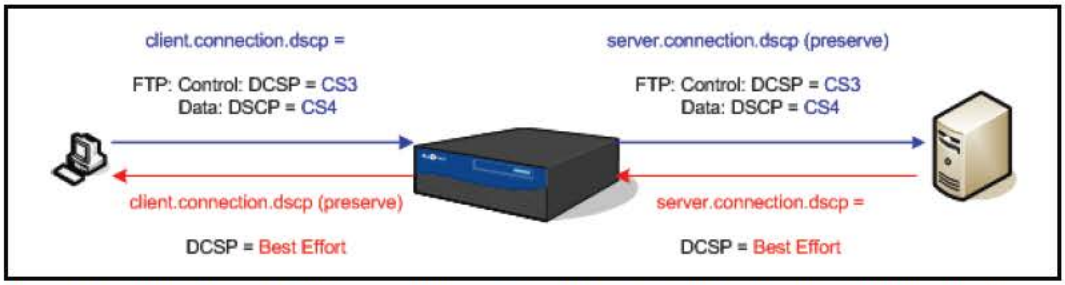


Figure 4-4. The Blue Coat appliance preserves client-to-server and server-to-client DSCP values (default)

Example Policy

```
<proxy>
  client.connection.dscp(preserve) server.connection.dscp(preserve)
```

Echoing the DSCP Value

Echoing is similar to preserving in that the outbound DSCP value remains the same as the inbound connection. The difference is that the point of reference is the SG appliance, not specifically the client-to-SG appliance connection. When policy is set to echo, the SG appliance returns the client's inbound DSCP back to the client or returns the server's inbound DSCP back to the server.

A deployment for which echoing is useful is reverse proxy, in which you want to let the client select the DSCP value in its request and then echo the reply back to that client with the same DSCP, even if the server does not set any DSCP on the packets it sends to the proxy.

The following diagram illustrates two different connections. The blue arrows represent a connection initiated by a client, with the policy set to echo. The red arrows represent a connection initiated by server, again with policy set to echo. Regardless of the DCSP value of the response, the QoS of the SG appliance back to the initiator remains the same as the sent value.

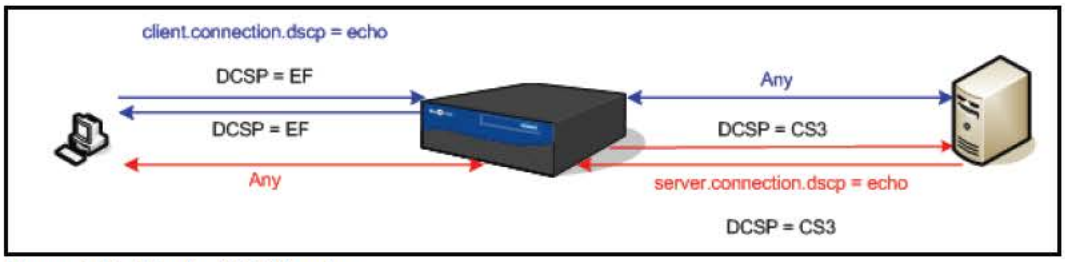


Figure 4-5. Echoing DSCP values

Example Policy

```
<proxy>
user=A client.connection.dscp(echo)
```

Setting the DSCP Value

QoS policy properties allow you to set outgoing (with the SG appliance as the point of reference) DSCP values. At present, the SG appliance supports setting one DSCP value for all connections in a transaction (the only exception is the preserve property). If a cache hit occurs for one of the connection types, thus negating the requirement for a server connection, the default value (**Best Effort**) is assigned.

In the following diagram, the SG appliance intercepts a request that has a default QoS level of `Best Effort`. The SG appliance then initiates the server request at QoS level `CS4`.

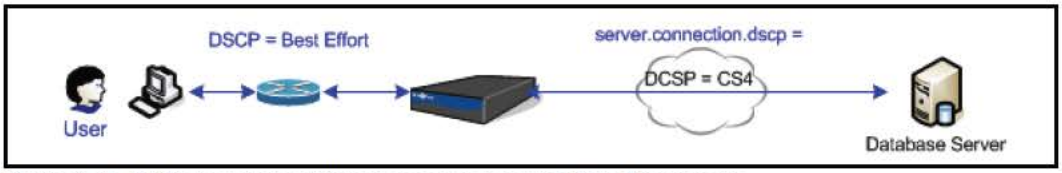


Figure 4-6. Setting an SG appliance-to-server connection DSCP value

Real Solutions: Combining QoS Policies

Applying QoS policies to different connections in your network helps control traffic network traffic flow. Consider the following example:

- A branch sales office is comprised of a VP of Sales and various sales personnel. The VP requires a moderately higher QoS server connection.
- The office has a SG appliance 200-C deployed as its WAN proxy.
- At the core offices, a SG appliance 810 fronts a database server farm, which contains inter-company collateral.

Therefore, the policy instructs the SG appliance 200-C to echo the connections between the clients and the proxy; that is, they receive the same QoS level as they requested over the WAN. Then, the policy instructs the SG appliance 200-C to make the server connection with a QoS level of `CS2`, except when user `VP_Sales` is identified. The VP is granted a QoS level of `CS4`, which in this case is defined as a higher QoS than `CS2`. The following diagram illustrates this example.

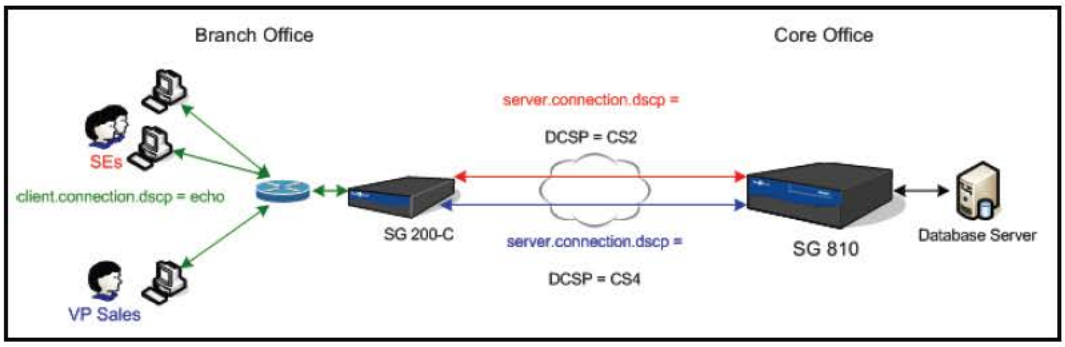


Figure 4-7. Setting DSCP values, based on user level, from the SG appliance to users

Example Policy

```
<proxy>
client.connection.dscp(echo)
user=vp_sales server.connection.dscp(CS4)
server.connection.dscp(cs2)
```

DSCP for ADN Tunnels

Through policy, you can manage DSCP values for upstream and downstream server connections over ADN tunnels. This was not achievable in SGOS 5.1.x.

Policy Components

This section lists the existing VPM and CPL policy components.

VPM Objects

Objects: (the cross-references are to the object descriptions in Chapter 3: "The Visual Policy Manager"):

- "Client Connection DSCP Trigger" on page 63—**Web Access, DNS Access layers: Source** column.
- "Server Connection DSCP Trigger" on page 74—**Web Access, DNS Access, Web Content, Forwarding** layers: **Destination** column.
- "Set Server Connection DSCP Value" on page 115—**Web Access, DNS Access, Web Content, Forwarding** layers: **Destination** column.
- "Set Client Connection DSCP Value" on page 114—**Web Access, DNS Access** layers: **Action** column.
- "Set Server Connection DSCP Value" on page 115—**Web Access, Forwarding** layers: **Action** column.
- "Set ADN Connection DSCP" on page 115—**Forwarding** layer: **Action** column.

VPM Example

The following VPM screen illustrates configuring a Web Access rule to set the DSCP value for P2P connections to **Best Effort** (no priority).

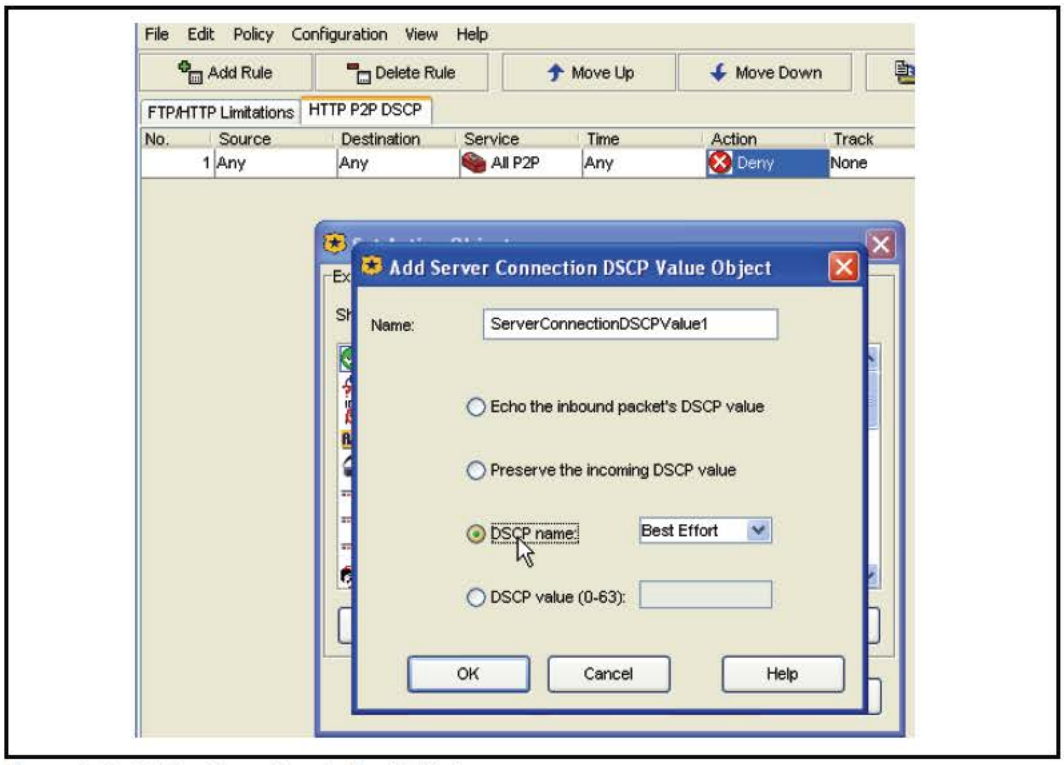


Figure 4-8. Setting the action to Best Effort

CPL Components

The following are the CPL triggers and properties:

Triggers

- `client.connection.dscp = 0..63 | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | best-effort | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef`

  Valid layers: `<proxy>, <dns-proxy>, <forward>`

- `server.connection.dscp = 0..63 | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | best-effort | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef`

  Valid layers: `<proxy>, <dns-proxy>, <cache>`

Properties

- `adn.connection.dscp(0..63 | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | best-effort | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | preserve)`

  Valid layers: `<forward>`

- `client.connection.dscp(0..63 | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | best-effort | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | echo | preserve)`

  Valid layers: `<proxy>, <dns-proxy>`

- `server.connection.dscp(0..63 | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | best-effort | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | echo | preserve)`

  Valid layers: `<proxy>`, `<dns-proxy>`, `<cache>`, `<forward>`

Access Logging

The following access log formats are associated with QoS activity:

- `x-cs-connection-dscp`: The incoming client DSCP value.
- `x-rs-connection-dscp`: The incoming server DSCP value.
- `x-sc-connection-dscp-decision`: The `client.connection.dscp ()` property value, or `preserve` or `echo`.
- `x-sr-connection-dscp-decision`: The `server.connection.dscp ()` property value, or `preserve` or `echo`.

Appendix A: Glossary

A

access control list	Allows or denies specific IP addresses access to a server.
access log	A list of all the requests sent to an appliance. You can read an access log using any of the popular log-reporting programs. When a client uses HTTP streaming, the streaming entry goes to the same access log.
account	A named entity that has purchased the appliance or the Entitlements from Blue Coat.
activation code	A string of approximately 10 characters that is generated and mailed to customers when they purchase the appliance.
active content stripping	Provides a way to identify potentially dangerous mobile or active content and scripts, and strip them out of a response.
active content types	Used in the Visual Policy Manager. Referring to Web Access policies, you can create and name lists of active content types to be stripped from Web pages. You have the additional option of specifying a customized message to be displayed to the user
administration access policy	A policy layer that determines who can access the SG appliance to perform administrative tasks.
administration authentication policy	A policy layer that determines how administrators accessing the SG appliance must authenticate.
Application Delivery Network (ADN)	A WAN that has been optimized for acceleration and compression by Blue Coat. This network can also be secured through the use of appliance certificates. An ADN network is composed of an ADN manager and backup ADN manager, ADN nodes, and a network configuration that matches the environment.
ADN backup manager	Takes over for the ADN manager in the event it becomes unavailable. See *ADN manager*.
ADN manager	Responsible for publishing the routing table to SG Clients (and to other SG appliances).
ADN optimize attribute	Controls whether to optimize bandwidth usage when connecting upstream using an ADN tunnel.
asx rewrite	Allows you to rewrite URLs and then direct a client's subsequent request to the new URL. One of the main applications of ASX file rewrites is to provide explicit proxy-like support for Windows Media Player 6.4, which cannot set explicit proxy mode for protocols other than HTTP.
audit	A log that provides a record of who accessed what and how.

authenticate-401 attribute	All transparent and explicit requests received on the port always use transparent authentication (cookie or IP, depending on the configuration). This is especially useful to force transparent proxy authentication in some proxy-chaining scenarios
authenticated content	Cached content that requires authentication at the origin content server (OCS). Supported authentication types for cached data include basic authentication and IWA (or NTLM).
authentication	Allows you to verify the identity of a user. In its simplest form, this is done through usernames and passwords. Much more stringent authentication can be employed using digital certificates that have been issued and verified by a Certificate Authority. *See also* basic authentication, proxy authentication, and SSL authentication.
authentication realm	Authenticates and authorizes users to access SG services using either explicit proxy or transparent proxy mode. These realms integrate third-party vendors, such as LDAP, Windows, and Novell, with the Blue Coat operating system.
authorization	The permissions given to an authenticated user.

B

bandwidth class	A defined unit of bandwidth allocation.
bandwidth class hierarchy	Bandwidth classes can be grouped together in a class hierarchy, which is a tree structure that specifies the relationship among different classes. You create a hierarchy by creating at least one parent class and assigning other classes to be its children.
bandwidth management	Classify, control, and, if needed, limit the amount of bandwidth used by network traffic flowing in or out of an SG appliance.
basic authentication	The standard authentication for communicating with the target as identified in the URL.
BCAAA	Blue Coat Authentication and Authorization Agent. Allows SGOS 5.x to manage authentication and authorization for IWA, CA eTrust SiteMinder realms, Oracle COREid, Novell, and Windows realms. The agent is installed and configured separately from SGOS 5.x and is available from the Blue Coat Web site.
BCLP	Blue Coat Licensing Portal.
byte-range support	The ability of the SG appliance to respond to byte-range requests (requests with a `Range:` HTTP header).

C

cache	An "object store," either hardware or software, that stores information (objects) for later retrieval. The first time the object is requested, it is stored, making subsequent requests for the same information much faster. A cache helps reduce the response time and network bandwidth consumption on future, equivalent requests. The SG appliance serves as a cache by storing content from many users to minimize response time and prevent extraneous network traffic.
cache control	Allows you to configure which content the SG appliance stores.

cache efficiency	A tab found on the Statistics pages of the Management Console that shows the percent of objects served from cache, the percent loaded from the network, and the percent that were non-cacheable.
cache hit	Occurs when the SG appliance receives a request for an object and can serve the request from the cache without a trip to the origin server.
cache miss	Occurs when the appliance receives a request for an object that is not in the cache. The appliance must then fetch the requested object from the origin server. .
cache object	Cache contents includes all objects currently stored by the SG appliance. Cache objects are not cleared when the SG appliance is powered off.
Certificate Authority (CA)	A trusted, third-party organization or company that issues digital certificates used to create digital signatures and public key/private key pairs. The role of the CA is to guarantee that the individuals or company representatives who are granted a unique certificate are who they claim to be.
child class (bandwidth gain)	The child of a parent class is dependent upon that parent class for available bandwidth (they share the bandwidth in proportion to their minimum/maximum bandwidth values and priority levels). A child class with siblings (classes with the same parent class) shares bandwidth with those siblings in the same manner.
client consent certificates	A certificate that indicates acceptance or denial of consent to decrypt an end user's HTTPS request.
client-side transparency	A way of replacing the appliance IP address with the Web server IP address for all port 80 traffic destined to go to the client. This effectively conceals the SG appliance address from the client and conceals the identity of the client from the Web server.
concentrator	An SG appliance, usually located in a data center, that provides access to data center resources, such as file servers.
content filtering	A way of controlling which content is delivered to certain users. SG appliances can filter content based on content categories (such as gambling, games, and so on), type (such as http, ftp, streaming, and mime type), identity (user, group, network), or network conditions. You can filter content using vendor-based filtering or by allowing or denying access to URLs.

D

default boot system	The system that was successfully started last time. If a system fails to boot, the next most recent system that booted successfully becomes the default boot system.
default proxy listener	*See* proxy service (d efault).
denial of service (DoS)	A method that hackers use to prevent or deny legitimate users access to a computer, such as a Web server. DoS attacks typically send many request packets to a targeted Internet server, flooding the server's resources and making the system unusable. Any system connected to the Internet and equipped with TCP-based network services is vulnerable to a DoS attack. The SG appliance resists DoS attacks launched by many common DoS tools. With a hardened TCP/IP stack, SG appliance resists common network attacks, including traffic flooding.

destination objects	Used in Visual Policy Manager. These are the objects that define the target location of an entry type.
detect protocol attribute	Detects the protocol being used. Protocols that can be detected include: HTTP, P2P (eDonkey, BitTorrent, FastTrack, Gnutella), SSL, and Endpoint Mapper.
diagnostic reporting	Found in the Statistics pane, the Diagnostics tab allows you to control whether Daily Heartbeats and/or Blue Coat Monitoring are enabled or disabled.
directives	Commands used in installable lists to configure forwarding and SOCKS gateway.
DNS access	A policy layer that determines how the SG appliance processes DNS requests.
domain name system (DNS)	An Internet service that translates domain names into IP addresses. *See also* private DNS or public DNS.
dynamic bypass	Provides a maintenance-free method for improving performance of the SG appliance by automatically compiling a list of requested URLs that return various kinds of errors.
dynamic real-time rating (DRTR)	Used in conjunction with the Blue Coat Web Filter (BCWF), DRTR (also known as *dynamic categorization*) provides real-time analysis and content categorization of requested Web pages to solve the problem of new and previously unknown uncategorized URLs—those not in the database. When a user requests a URL that has not already been categorized by the BCWF database (for example, a brand new Web site), the SG appliance dynamic categorization service analyzes elements of the requested content and assigns a category or categories. The dynamic service is consulted *only* when the installed BCWF database does not contain category information for an object.

E

early intercept attribute	Controls whether the proxy responds to client TCP connection requests before connecting to the upstream server. When early intercept is disabled, the proxy delays responding to the client until after it has attempted to contact the server.
ELFF-compatible format	A log type defined by the W3C that is general enough to be used with any protocol.
emulated certificates	Certificates that are presented to the user by SG appliance when intercepting HTTPS requests. Blue Coat emulates the certificate from the server and signs it, copying the subjectName and expiration. The original certificate is used between the SG appliance and the server.
encrypted log	A log is encrypted using an external certificate associated with a private key. Encrypted logs can only be decrypted by someone with access to the private key. The private key is not accessible to the SG appliance.
EULA	End user license agreement.
event logging	Allows you to specify the types of system events logged, the size of the event log, and to configure Syslog monitoring. The appliance can also notify you by email if an event is logged. *See also* access logging.

explicit proxy	A configuration in which the browser is explicitly configured to communicate with the proxy server for access to content. This is the default for the SG appliance, and requires configuration for both browser and the interface card.
extended log file format (ELFF)	A variant of the common log file format, which has two additional fields at the end of the line—the referer and the user agent fields.

F

fail open/closed	Failing open or closed applies to forwarding hosts and groups and SOCKS gateways. Fail open or closed applies when health checks are showing sick for each forwarding or SOCKS gateway target in the applicable fail-over sequence. If no systems are healthy, the SG appliance fails open or closed, depending on the configuration. If closed, the connection attempt simply fails. If open, an attempt is made to connect without using any forwarding target (or SOCKS gateway). Fail open is usually a security risk; fail closed is the default if no setting is specified.
filtering	*See* content filtering.
forward proxy	A proxy server deployed close to the clients and used to access many servers. A forward proxy can be explicit or transparent.
FTP	*See* Native FTP; Web FTP.

G

gateway	A device that serves as entrance and exit into a communications network.

H

hardware serial number	A string that uniquely identifies the appliance; it is assigned to each unit in manufacturing.
health check tests	The method of determining network connectivity, target responsiveness, and basic functionality. The following tests are supported: • ICMP • TCP • SSL • HTTP • HTTPS • Group • Composite and reference to a composite result • ICAP • Websense • DRTR rating service

health check type	The kind of device or service the specific health check tests. The following types are supported: • Forwarding host and forwarding group • SOCKS gateway and SOCKS gateway group • CAP service and ICAP service group • Websense off-box service and Websense off-box service group • DRTR rating service • User-defined host and a user-defined composite
heartbeat	Messages sent once every 24 hours that contain the statistical and configuration data for the SG appliance, indicating its health. Heartbeats are commonly sent to system administrators and to Blue Coat. Heartbeats contain no private information, only aggregate statistics useful for pre-emptively diagnosing support issues. The SG appliance sends emergency heartbeats whenever it is rebooted. Emergency heartbeats contain core dump and restart flags in addition to daily heartbeat information.
host affinity	The attempt to direct multiple connections by a single user to the same group member. Host affinity is closely tied to load balancing behavior; both should be configured if load balancing is important.
host affinity timeout	The host affinity timeout determines how long a user remains idle before the connection is closed. The timeout value checks the user's IP address, SSL ID, or cookie in the host affinity table.

I

inbound traffic (bandwidth gain)	Network packets flowing into the SG appliance. Inbound traffic mainly consists of the following: • Server inbound: Packets originating at the origin content server (OCS) and sent to the SG appliance to load a Web object. • Client inbound: Packets originating at the client and sent to the SG appliance for Web requests.
installable lists	Installable lists, comprised of directives, can be placed onto the SG appliance in one of the following ways: • Creating the list using the SG text editor • Placing the list at an accessible URL • Downloading the directives file from the local system
integrated host timeout	An integrated host is an origin content server (OCS) that has been added to the health check list. The host, added through the `integrate_new_hosts` property, ages out of the integrated host table after being idle for the specified time. The default is 60 minutes.
intervals	Time period from the completion of one health check to the start of the next health check.
IP reflection	Determines how the client IP address is presented to the origin server for explicitly proxied requests. All proxy services contain a reflect-ip attribute, which enables or disables sending of client's IP address instead of the SG's IP address.

issuer keyring	The keyring used by the SG appliance to sign emulated certificates. The keyring is configured on the appliance and managed through policy.

L

licensable component (LC)	(Software) A subcomponent of a license; it is an option that enables or disables a specific feature.
license	Provides both the right and the ability to use certain software functions within an AV (or SG) appliance. The license key defines and controls the license, which is owned by an account.
listener	The service that is listening on a specific port. A listener can be identified by any destination IP/subnet and port range. Multiple listeners can be added to each service.
live content	Also called live broadcast. Used in streaming, it indicates that the content is being delivered fresh.
LKF	License key file.
load balancing	A way to share traffic requests among multiple upstream systems or multiple IP addresses on a single host.
local bypass list	A list you create and maintain on your network. You can use a local bypass list alone or in conjunction with a central bypass list. *See* bypass list.
local policy file	Written by enterprises (as opposed to the central policy file written by Blue Coat); used to create company- and department-specific advanced policies written in the Blue Coat Policy Language (CPL).
log facility	A separate log that contains a single logical file and supports a single log format. It also contains the file's configuration and upload schedule information as well as other configurable information such as how often to rotate (switch to a new log) the logs at the destination, any passwords needed, and the point at which the facility can be uploaded.
log format	The type of log that is used: NCSA/Common, SQUID, ELFF, SurfControl, or Websense. The proprietary log types each have a corresponding pre-defined log format that has been set up to produce exactly that type of log (these logs cannot be edited). In addition, a number of other ELFF type log formats are also pre-defined (im, main, p2p, ssl, streaming). These can be edited, but they start out with a useful set of log fields for logging particular protocols understood by the SG appliance. It is also possible to create new log formats of type ELFF or Custom which can contain any desired combination of log fields.
log tail	The access log tail shows the log entries as they get logged. With high traffic on the SG appliance, not all access log entries are necessarily displayed. However, you can view all access log information after uploading the log.

M

MACH5	SGOS 5 MACH5 Edition.

Management Console	A graphical Web interface that lets you to manage, configure, monitor, and upgrade the SG appliance from any location. The Management Console consists of a set of Web pages and Java applets stored on the SG appliance. The appliance acts as a Web server on the management port to serve these pages and applets.
management information base (MIB)	Defines the statistics that management systems can collect. A managed device (gateway) has one or more MIBs as well as one or more SNMP agents, which implements the information and management functionality defined by a specific MIB.
maximum object size	The maximum object size stored in the SG appliance. All objects retrieved that are greater than the maximum size are delivered to the client but are not stored in the SG appliance.
MIME/FILE type filtering	Allows organizations to implement Internet policies for both uploaded and downloaded content by MIME or FILE type.
multi-bit rate	The capability of a single stream to deliver multiple bit rates to clients requesting content from appliances from within varying levels of network conditions (such as different connecting bandwidths and traffic).
multicast	Used in streaming; the ability for hundreds or thousands of users to play a single stream.
multicast aliases	Used in streaming; a streaming command that specifies an alias for a multicast URL to receive an .nsc file. The .nsc files allows the multicast session to obtain the information in the control channel
multicast station	Used in streaming; a defined location on the proxy where the Windows Media player can retrieve streams. A multicast station enables multicast transmission of Windows Media content from the cache. The source of the multicast-delivered content can be a unicast-live source, a multicast (live) source, and simulated live (video-on-demand content converted to scheduled live content).
multimedia content services	Used in streaming; multimedia support includes Real Networks, Microsoft Windows Media, Apple QuickTime, MP3, and Flash.

N

name inputing	Allows an SG appliance to resolve host names based on a partial name specification. When a host name is submitted to the DNS server, the DNS server resolves the name to an IP address. If the host name cannot be resolved, Blue Coat adds the first entry in the name-inputing list to the end of the host name and resubmits it to the DNS server
native FTP	Native FTP involves the client connecting (either explicitly or transparently) using the FTP protocol; the SG appliance then connects upstream through FTP (if necessary).
NCSA common log format	Blue Coat products are compatible with this log type, which contains only basic HTTP access information.
network address translation (NAT)	The process of translating private network (such as intranet) IP addresses to Internet IP addresses and vice versa. This methodology makes it possible to match private IP addresses to Internet IP addresses even when the number of private addresses outnumbers the pool of available Internet addresses.

non-cacheable objects	A number of objects are not cached by the Blue Coat appliance because they are considered non-cacheable. You can add or delete the kinds of objects that the appliance considers non-cacheable. Some of the non-cacheable request types are: • Pragma no-cache, requests that specify non-cached objects, such as when you click refresh in the Web browser. • Password provided, requests that include a client password. • Data in request that include additional client data. • Not a GET request.
.nsc file	Created from the multicast station definition and saved through the browser as a text file encoded in a Microsoft proprietary format. Without an .nsc file, the multicast station definition does not work.
NTP	To manage objects in an appliance, an SG appliance must know the current Universal Time Coordinates (UTC) time. By default, the SG appliance attempts to connect to a Network Time Protocol (NTP) server to acquire the UTC time. SG appliance includes a list of NTP servers available on the Internet, and attempts to connect to them in the order they appear in the NTP server list on the NTP tab.

O

object (used in caching)	An object is the item that is stored in an appliance. These objects can be frequently accessed content, content that has been placed there by content publishers, or Web pages, among other things.
object (used in Visual Policy Manager)	An object (sometimes referred to as a condition) is any collection or combination of entry types you can create individually (user, group, IP address/subnet, and attribute). To be included in an object, an item must already be created as an individual entry.
object pipelining	This patented algorithm opens as many simultaneous TCP connections as the origin server will allow and retrieves objects in parallel. The objects are then delivered from the appliance straight to the user's desktop as fast as the browser can request them.
origin content server (OCS)	Also called origin server. This is the original source of the content that is being requested. An appliance needs the OCS to acquire data the first time, to check that the content being served is still fresh, and to authenticate users.
outbound traffic (bandwidth gain)	Network packets flowing out of the SG appliance. Outbound traffic mainly consists of the following: • Client outbound: Packets sent to the client in response to a Web request. • Server outbound: Packets sent to an OCS or upstream proxy to request a service.

P

PAC (Proxy AutoConfiguration) scripts	Originally created by Netscape, PACs are a way to avoid requiring proxy hosts and port numbers to be entered for every protocol. You need only enter the URL. A PAC can be created with the needed information and the local browser can be directed to the PAC for information about proxy hosts and port numbers.
packet capture (PCAP)	Allows filtering on various attributes of the Ethernet frame to limit the amount of data collected. You can capture packets of Ethernet frames going into or leaving an SG appliance.

parent class (bandwidth gain)	A class with at least one child. The parent class must share its bandwidth with its child classes in proportion to the minimum/maximum bandwidth values or priority levels.
passive mode data connections (PASV)	Data connections initiated by an FTP client to an FTP server.
pipelining	*See* object pipelining.
policies	Groups of rules that let you manage Web access specific to the needs of an enterprise. Policies enhance SG appliance feature areas such as authentication and virus scanning, and let you control end-user Web access in your existing infrastructure. *See also* refresh policies.
policy-based bypass list	Used in policy. Allows a bypass based on the properties of the client, unlike static and dynamic bypass lists, which allow traffic to bypass the appliance based on destination IP address. *See also* bypass lists and dynamic bypass.
policy layer	A collection of rules created using Blue Coat CPL or with the VPM.
pragma: no cache (PNC)	A metatag in the header of a request that requires the appliance to forward a request to the origin server. This allows clients to always obtain a fresh copy *(of the request?).*
proxy	Caches content, filters traffic, monitors Internet and intranet resource usage, blocks specific Internet and intranet resources for individuals or groups, and enhances the quality of Internet or intranet user experiences. A proxy can also serve as an intermediary between a Web client and a Web server and can require authentication to allow identity based policy and logging for the client. The rules used to authenticate a client are based on the policies you create on the SG appliance, which can reference an existing security infrastructure—LDAP, RADIUS, IWA, and the like.
Proxy Edition	SGOS 5 Proxy Edition.
proxy service	The proxy service defines the ports, as well as other attributes. that are used by the proxies associated with the service.
proxy service (default)	The default proxy service is a service that intercepts all traffic not otherwise intercepted by other listeners. It only has one listener whose action can be set to bypass or intercept. No new listeners can be added to the default proxy service, and the default listener and service cannot be deleted. Service attributes can be changed.
public key certificate	An electronic document that encapsulates the public key of the certificate sender, identifies this sender, and aids the certificate receiver to verify the identity of the certificate sender. A certificate is often considered valid if it has been digitally signed by a well-known entity, which is called a Certificate Authority (such as VeriSign).
public virtual IP (VIP)	Maps multiple servers to one IP address and then propagates that information to the public DNS servers. Typically, there is a public VIP known to the public Internet that routes the packets internally to the private VIP. This enables you to “hide” your servers from the Internet.

R

real-time streaming protocol (RTSP)	A standard method of transferring audio and video and other time-based media over Internet-technology based networks. The protocol is used to stream clips to any RTP-based client.
reflect client IP attribute	Enables the sending of the client's IP address instead of the SG's IP address to the upstream server. If you are using an application delivery network (ADN), this setting is enforced on the concentrator proxy through the **Configuration > App. Delivery Network > Tunneling** tab.
registration	An event that binds the appliance to an account, that is, it creates the Serial#, Account association.
remote authentication dial-in user service (RADIUS)	Authenticates user identity via passwords for network access.
reverse proxy	A proxy that acts as a front-end to a small number of pre-defined servers, typically to improve performance. Many clients can use it to access the small number of predefined servers.
routing information protocol (RIP)	Designed to select the fastest route to a destination. RIP support is built into Blue Coat appliances.
router hops	The number of jumps a packet takes when traversing the Internet.

S

secure shell (SSH)	Also known as Secure Socket Shell. SSH is an interface and protocol that provides strong authentication and enables you to securely access a remote computer. Three utilities—login, ssh, and scp—comprise SSH. Security via SSH is accomplished using a digital certificate and password encryption. Remember that the Blue Coat SG appliance requires SSH1. An SG appliance supports a combined maximum of 16 Telnet and SSH sessions.
serial console	A third-party device that can be connected to one or more Blue Coat appliances. Once connected, you can access and configure the appliance through the serial console, even when you cannot access the appliance directly.
server certificate categories	The hostname in a server certificate can be categorized by BCWF or another content filtering vendor to fit into categories such as banking, finance, sports.
server portals	Doorways that provide controlled access to a Web server or a collection of Web servers. You can configure Blue Coat SG appliances to be server portals by mapping a set of external URLs onto a set of internal URLs.
server-side transparency	The ability for the server to see client IP addresses, which enables accurate client-access records to be kept. When server-side transparency is enabled, the appliance retains client IP addresses for all port 80 traffic to and from the SG appliance. In this scheme, the client IP address is always revealed to the server.
service attributes	Define the parameters, such as explicit or transparent, cipher suite, and certificate verification, that the SG appliance uses for a particular service. .

SG appliance	A Blue Coat security and cache box that can help manage security and content on a network.
sibling class (bandwidth gain)	A bandwidth class with the same parent class as another class.
simple network management protocol (SNMP)	The standard operations and maintenance protocol for the Internet. It uses MIBs, created or customized by Blue Coat, to handle *(needs completion).*
simulated live	Used in streaming. Defines playback of one or more video-on-demand files as a scheduled live event, which begins at a specified time. The content can be looped multiple times, or scheduled to start at multiple start times throughout the day.
SmartReporter log type	A proprietary ELFF log type that is compatible with the SmartFilter SmartReporter tool.
SOCKS	A proxy protocol for TCP/IP-based networking applications that allows users transparent access across the firewall. If you are using a SOCKS server for the primary or alternate forwarding gateway, you must specify the appliance's ID for the identification protocol used by the SOCKS gateway. The machine ID should be configured to be the same as the appliance's name.
SOCKS proxy	A generic way to proxy TCP and UDP protocols. The SG appliance supports both SOCKSv4/4a and SOCKSv5; however, because of increased username and password authentication capabilities and compression support, Blue Coat recommends that you use SOCKS v5.
splash page	Custom message page that displays the first time you start the client browser.
split proxy	Employs co-operative processing at the branch and the core to implement functionality that is not possible in a standalone proxy. Examples of split proxies include: • Mapi Proxy • SSL Proxy
SQUID-compatible format	A log type that was designed for cache statistics and is compatible with Blue Coat products.
squid-native log format	The Squid-compatible format contains one line for each request.
SSL authentication	Ensures that communication is with "trusted" sites only. Requires a certificate issued by a trusted third party (Certificate Authority).
SSL interception	Decrypting SSL connections.
SSL proxy	A proxy that can be used for any SSL traffic (HTTPS or not), in either forward or reverse proxy mode.
static route	A manually-configured route that specifies the transmission path a packet must follow, based on the packet's destination address. A static route specifies a transmission path to another network.

statistics	Every Blue Coat appliance keeps statistics of the appliance hardware and the objects it stores. You can review the general summary, the volume, resources allocated, cache efficiency, cached contents, and custom URLs generated by the appliance for various kinds of logs. You can also check the event viewer for every event that occurred since the appliance booted.
stream	A flow of a single type of data, measured in kilobits per second (Kbps). A stream could be the sound track to a music video, for example.
SurfControl log type	A proprietary log type that is compatible with the SurfControl reporter tool. The SurfControl log format includes fully-qualified usernames when an NTLM realm provides authentication. The simple name is used for all other realm types.
syslog	An event-monitoring scheme that is especially popular in Unix environments. Most clients using Syslog have multiple devices sending messages to a single Syslog daemon. This allows viewing a single chronological event log of all of the devices assigned to the Syslog daemon. The Syslog format is: "Date Time Hostname Event."
system cache	The software cache on the appliance. When you clear the cache, all objects in the cache are set to expired. The objects are not immediately removed from memory or disk, but a subsequent request for any object requested is retrieved from the origin content server before it is served.

T

time-to-live (TTL) value	Used in any situation where an expiration time is needed. For example, you do not want authentication to last beyond the current session and also want a failed command to time out instead of hanging the box forever.
traffic flow (bandwidth gain)	Also referred to as *flow.* A set of packets belonging to the same TCP/UDP connection that terminate at, originate at, or flow through the SG appliance. A single request from a client involves two separate connections. One of them is from the client to the SG appliance, and the other is from the SG appliance to the OCS. Within each of these connections, traffic flows in two directions—in one direction, packets flow out of the SG appliance (outbound traffic), and in the other direction, packets flow into the SG (inbound traffic). Connections can come from the client or the server. Thus, traffic can be classified into one of four types: • Server inbound • Server outbound • Client inbound • Client outbound These four traffic flows represent each of the four combinations described above. Each flow represents a single direction from a single connection.
transmission control protocol (TCP)	TCP, when used in conjunction with IP (Internet Protocol) enables users to send data, in the form of message units called packets, between computers over the Internet. TCP is responsible for tracking and handling, and reassembly of the packets; IP is responsible for packet delivery.
transparent proxy	A configuration in which traffic is redirected to the SG appliance without the knowledge of the client browser. No configuration is required on the browser, but network configuration, such as an L4 switch or a WCCP-compliant router, is required.

trial period	Starting with the first boot, the trial period provides 60 days of free operation. All features are enabled during this time.
U	
unicast alias	Defines an name on the appliance for a streaming URL. When a client requests the alias content on the appliance, the appliance uses the URL specified in the unicast-alias command to request the content from the origin streaming server.
universal time coordinates (UTC)	An SG appliance must know the current UTC time. By default, the appliance attempts to connect to a Network Time Protocol (NTP) server to acquire the UTC time. If the SG appliance cannot access any NTP servers, you must manually set the UTC time.
URL filtering	*See* content filtering.
URL rewrite rules	Rewrite the URLs of client requests to acquire the streaming content using the new URL. For example, when a client tries to access content on www.mycompany.com, the appliance is actually receiving the content from the server on 10.253.123.123. The client is unaware that mycompany.com is not serving the content; however, the appliance access logs indicate the actual server that provides the content.
W	
WCCP	Web Cache Communication Protocol. Allows you to establish redirection of the traffic that flows through routers.
Web FTP	Web FTP is used when a client connects in explicit mode using HTTP and accesses an ftp:// URL. The SG appliance translates the HTTP request into an FTP request for the OCS (if the content is not already cached), and then translates the FTP response with the file contents into an HTTP response for the client.
Websense log type	A Blue Coat proprietary log type that is compatible with the Websense reporter tool.
X	
XML responder	HTTP XML service that runs on an external server.
XML requestor	XML realm.