Concepts & Examples
ScreenOS Reference Guide

Volume 4:
Attack Detection and Defense Mechanisms

Release 5.4.0, Rev. A

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, CA 94089
USA
408-745-2000
www.juniper.net

Part Number: 530-015771-01, Revision A

FCC Statement

The following information is for FCC compliance of Class A devices: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. The equipment generates, uses, and can radiate radio-frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case users will be required to correct the interference at their own expense.

The following information is for FCC compliance of Class B devices: The equipment described in this manual generates and may radiate radio-frequency energy. If it is not installed in accordance with Juniper Networks' installation instructions, it may cause interference with radio and television reception. This equipment has been tested and found to comply with the limits for a Class B digital device in accordance with the specifications in part 15 of the FCC rules. These specifications are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation.

If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Consult the dealer or an experienced radio/TV technician for help.
- Connect the equipment to an outlet on a circuit different from that to which the receiver is connected.

Caution: Changes or modifications to this product could void the user's warranty and authority to operate this device.

Disclaimer

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR JUNIPER NETWORKS REPRESENTATIVE FOR A COPY.

Writers: ScreenOS Team

Editor: Lisa Eldridge

Table of Contents

Chapter 6 Continued

About This Volume

Volume 4: Attack Detection and Defense Mechanisms describes the Juniper Networks security options available in ScreenOS. You can enable many of these options at the security zone level. These options apply to traffic reaching the Juniper Networks security device through any interface bound to a zone for which you have enabled such options. These options offer protection against IP address and port scans, denial-of-service (DoS) attacks, and other kinds of malicious activity. You can apply other network security options, such as web filtering, antivirus checking, and intrusion detection and prevention (IDP), at the policy level. These options only apply to traffic under the jurisdiction of the policies in which they are enabled.

NOTE: The subject of policies is presented only peripherally in this volume, as it applies to the network security options that you can enable at the policy level. For a complete examination of policies, see "Policies" on page **2**-171.

This volume contains the following sections:

- Chapter 1, "Protecting a Network," outlines the basic stages of an attack and the firewall options available to combat the attacker at each stage.
- Chapter 2, "Reconnaissance Deterrence," describes the options available for blocking IP address sweeps, port scans, and attempts to discover the type of operating system (OS) of a targeted system.
- Chapter 3, "Denial-of-Service (DoS) Attack Defenses," explains firewall, network, and OS-specific DoS attacks and how ScreenOS mitigates such attacks.
- Chapter 4, "Content Monitoring and Filtering," describes how to protect HyperText Transfer Protocol (HTTP) and File Transfer Protocol (FTP) users from malicious uniform resource locators (URLs) and how to configure the Juniper Networks security device to work with third-party products to provide antivirus scanning, anti-spam, and web filtering.
- Chapter 5, "Deep Inspection," describes how to configure the Juniper Networks security device to obtain IDP attack object updates, how to create user-defined attack objects and attack object groups, and how to apply IDP at the policy level.

- Chapter 6, “Intrusion Detection and Prevention,” describes Juniper Networks Intrusion Detection and Prevention (IDP) technology which can both detect and then stop attacks when deployed inline to your network. The chapter describes how to apply IDP at the policy level to drop malicious packets or connections before the attacks can enter your network.
- Chapter 7, “Suspicious Packet Attributes,” presents several SCREEN options that protect network resources from potential attacks indicated by unusual IP and ICMP packet attributes.
- Appendix A, “Contexts for User-Defined Signatures,” provides descriptions of contexts that you can specify when defining a stateful signature attack object.

Document Conventions

This document uses several types of conventions, which are introduced in the following sections:

CLI Conventions

The following conventions are used to present the syntax of CLI commands in examples and text.

In examples:

- Anything inside square brackets [ ] is optional.
- Anything inside braces **{ }** is required.
- If there is more than one choice, each choice is separated by a pipe (|). For example:

  set interface { ethernet1 | ethernet2 | ethernet3 } manage

  means "set the management options for the ethernet1, the ethernet2, *or* the ethernet3 interface."

- Variables are in *italic* type:

  set admin user *name1* password *xyz*

In text:

- Commands are in **boldface** type.
- Variables are in *italic* type.

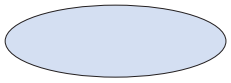
NOTE: When entering a keyword, you only have to type enough letters to identify the word uniquely. For example, typing **set adm u kath j12fmt54** is enough to enter the command **set admin user kathleen j12fmt54**. Although you can use this shortcut when entering commands, all the commands documented here are presented in their entirety.

Illustration Conventions

The following figure shows the basic set of images used in illustrations throughout this document.

Figure 1: Images in Illustrations

Autonomous System



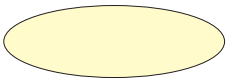
Local Area Network (LAN) with a Single Subnet (example: 10.1.1.0/24)

Generic Security Device

Internet

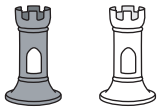
Virtual Routing Domain

Dynamic IP (DIP) Pool



Security Zone

Desktop Computer



Security Zone Interface

White = Protected Zone Interface (example = Trust Zone)

Black = Outside Zone Interface (example = Untrust Zone)

Laptop Computer

Generic Network Device (examples: NAT Server, Access Concentrator)

Tunnel Interface

Server

VPN Tunnel

Hub

Router

Policy Engine

Switch

IP Telephone

Naming Conventions and Character Types

ScreenOS employs the following conventions regarding the names of objects—such as addresses, admin users, auth servers, IKE gateways, virtual systems, VPN tunnels, and zones—defined in ScreenOS configurations:

- If a name string includes one or more spaces, the entire string must be enclosed within double quotes; for example:

  set address trust “local LAN” 10.1.1.0/24

- Any leading spaces or trailing text within a set of double quotes are trimmed; for example, **“ local LAN ”** becomes **“local LAN”**.

- Multiple consecutive spaces are treated as a single space.

- Name strings are case-sensitive, although many CLI keywords are case-insensitive. For example, **“local LAN”** is different from **“local lan”**.

ScreenOS supports the following character types:

- Single-byte character sets (SBCS) and multiple-byte character sets (MBCS). Examples of SBCS are ASCII, European, and Hebrew. Examples of MBCS—also referred to as double-byte character sets (DBCS)—are Chinese, Korean, and Japanese.

- ASCII characters from 32 (0x20 in hexadecimals) to 255 (0xff), except double quotes (“), which have special significance as an indicator of the beginning or end of a name string that includes spaces.

NOTE: A console connection only supports SBCS. The WebUI supports both SBCS and MBCS, depending on the character sets that your browser supports.

WebUI Conventions

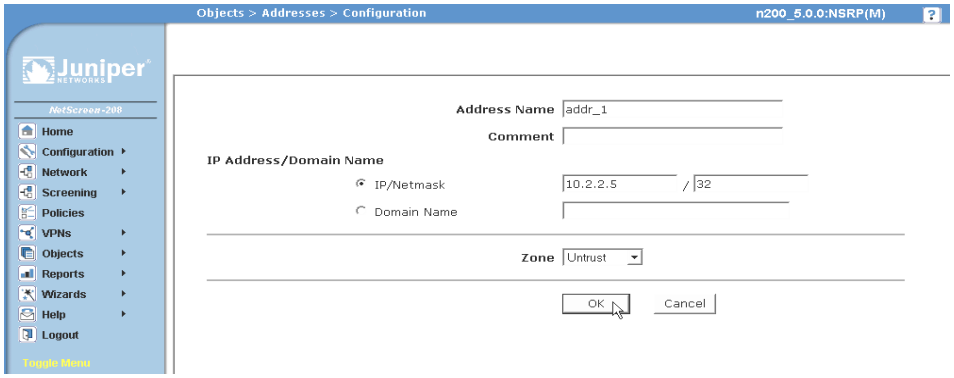
To perform a task with the WebUI, you first navigate to the appropriate dialog box, where you then define objects and set parameters. A chevron (>) shows the navigational sequence through the WebUI, which you follow by clicking menu options and links. The set of instructions for each task is divided into navigational path and configuration settings.

The following figure lists the path to the address configuration dialog box with the following sample configuration settings:

Objects > Addresses > List > New: Enter the following, then click **OK**:

Address Name: addr_1
IP Address/Domain Name:
IP/Netmask: (select), 10.2.2.5/32
Zone: Untrust

Figure 2: Navigational Path and Configuration Settings



Juniper Networks Documentation

To obtain technical documentation for any Juniper Networks product, visit www.juniper.net/techpubs/.

For technical support, open a support case using the Case Manager link at http://www.juniper.net/support/ or call 1-888-314-JTAC (within the United States) or 1-408-745-9500 (outside the United States).

If you find any errors or omissions in this document, please contact us at the email address below:

techpubs-comments@juniper.net

Chapter 1

Protecting a Network

There can be many reasons for invading a protected network. The following list contains some common objectives:

- Gathering the following kinds of information about the protected network:
  - Topology
  - IP addresses of active hosts
  - Numbers of active ports on active hosts
  - Operating systems of active hosts
- Overwhelming a host on a protected network with bogus traffic to induce a Denial-of-Service (DoS)
- Overwhelming the protected network with bogus traffic to induce a network-wide DoS
- Overwhelming a firewall with bogus traffic to induce a DoS for the network behind it
- Causing damage to and stealing data from a host on a protected network
- Gaining access to a host on a protected network to obtain information
- Gaining control of a host to launch other exploits
- Gaining control of a firewall to control access to the network that it protects

ScreenOS provides detective and defensive tools for uncovering and thwarting the efforts of attackers to achieve the above objectives when they attempt to target a network protected by a Juniper Networks security device.

This chapter presents an overview of the main stages of an attack and the various defense mechanisms that you can employ to thwart an attack at each stage:

Stages of an Attack

Each attack typically progresses in two major stages. In the first stage, the attacker gathers information, and in the second stage he or she launches the attack.

1. Perform reconnaissance.
    a. Map the network and determine which hosts are active (IP address sweep).
    b. Discern which ports are active (port scans) on the hosts discovered by the IP address sweep.
    c. Determine the operating system (OS), which might expose a weakness in the OS or suggest an attack to which that particular OS is susceptible.
2. Launch the attack.
    a. Conceal the origin of the attack.
    b. Perform the attack.
    c. Remove or hide evidence.

Detection and Defense Mechanisms

An exploit can be an information-gathering probe or an attack to compromise, disable, or harm a network or network resource. In some cases, the distinction between the two objectives of an exploit can be unclear. For example, a barrage of TCP SYN segments might be an IP address sweep with the intent of triggering responses from active hosts, or it might be a SYN flood attack with the intent of overwhelming a network so that it can no longer function properly. Furthermore, because an attacker usually precedes an attack by performing reconnaissance on the target, we can consider information-gathering efforts as a precursor to an impending attack—that is, they constitute the first stage of an attack. Thus, the term *exploit* encompasses both reconnaissance and attack activities, and the distinction between the two is not always clear.

Juniper Networks provides various detection methods and defense mechanisms at the zone and policy levels to combat exploits at all stages of their execution:

- SCREEN options at the zone level
- Firewall policies at the inter-, intra-, and super-zone policy levels (*super-zone* here means in global policies, where no security zones are referenced).

NOTE: Although the VLAN and MGT zones are function zones and not security zones, you can set SCREEN options for them. The VLAN zone supports the same set of SCREEN options as a Layer 3 security zone. (Layer 2 security zones support an additional SYN flood option that Layer 3 zones do not: Drop Unknown MAC). Because the following SCREEN options do not apply to the MGT zone, they are not available for that zone: SYN flood protection, SYN-ACK-ACK proxy flood protection, HTTP component blocking, and WinNuke attack protection.

To secure all connection attempts, Juniper Networks security devices use a dynamic packet-filtering method known as stateful inspection. Using this method, the security device notes various components in the IP packet and TCP segment headers— source and destination IP addresses, source and destination port numbers, and packet sequence numbers—and maintains the state of each TCP session and pseudo UDP session traversing the firewall. (The device also modifies session states based on changing elements such as dynamic port changes or session termination.) When a responding TCP packet arrives, the device compares the information reported in its header with the state of its associated session stored in the inspection table. If they match, the responding packet is allowed to pass the firewall. If the two do not match, the packet is dropped.

ScreenOS SCREEN options secure a zone by inspecting, then allowing or denying, all connection attempts that require crossing an interface bound to that zone. The security device then applies firewall policies, which can contain content filtering and intrusion detection and prevention (IDP) components, to the traffic that passes the SCREEN filters.

A Juniper Networks firewall provides the following sets of defense mechanisms:

- Reconnaissance deterrence
  - IP address sweep
  - Port scanning
  - Operating system probes
  - Evasion techniques
- Content monitoring and filtering
  - Fragment reassembly
  - Antivirus scanning
  - Anti-spam filtering
  - Web filtering
- Deep inspection
  - Stateful signatures
  - Protocol anomalies
  - Granular blocking of HTTP components

- Denial-of-Service (DoS) attack defenses
  - Firewall DoS attacks
    - Session table flood
    - SYN-ACK-ACK proxy flood
  - Network DoS attacks
    - SYN flood
    - ICMP flood
    - UDP flood
  - OS-specific DoS attacks
    - Ping of death
    - Teardrop attack
    - WinNuke
- Suspicious packet attributes
  - ICMP fragments
  - Large ICMP packets
  - Bad IP options
  - Unknown protocols
  - IP packet fragments
  - SYN fragments

ScreenOS network-protection settings operate at two levels: security zone and policy. The Juniper Networks security device performs reconnaissance deterrence and DoS attack defenses at the security zone level. In the area of content monitoring and filtering, the security device applies fragment reassembly at the zone level and antivirus (AV) scanning and uniform resource locator (URL) filtering at the policy level. The device applies IDP at the policy level, except for the detection and blocking of HTTP components, which occurs at the zone level. Zone-level firewall settings are SCREEN options. A network protection option set in a policy is a component of that policy.

Exploit Monitoring

Although you typically want the security device to block exploits, there might be times when you want to gather intelligence about them. You might want to learn specifically about a particular exploit—to discover its intention, its sophistication, and possibly (if the attacker is careless or unsophisticated) its source.

If you want to gather information about an exploit, you can let it occur, monitor it, analyze it, perform forensics, and then respond as delineated in a previously prepared incident response plan. You can instruct the security device to notify you of an exploit, but, instead of taking action, the device allows the exploit to transpire. You can then study what occurred, and try to understand the attacker's method, strategy, and objectives. Increased understanding of the threat to the network can then allow you to better fortify your defenses. Although a smart attacker can conceal his or her location and identity, you might be able to gather enough information to discern where the attack originated. You also might be able to estimate the attacker's capabilities. This kind of information allows you to gauge your response.

Example: Monitoring Attacks from the Untrust Zone

In this example, IP spoofing attacks from the Untrust zone have occurred on a daily basis, usually between 21:00 PM and 0:00 AM. Instead of dropping the packets with the spoofed source IP addresses, you want the security device to notify you of their arrival but allow them to pass, perhaps directing them to a honeypot (a decoy network server that is designed to lure attackers and then record their actions during an attack) that you have connected on the DMZ interface connection. At 20:55 PM, you change the firewall behavior from notification and rejection of packets belonging to a detected attack to notification and acceptance. When the attack occurs, you can then use the honeypot to monitor the attacker's activity after crossing the firewall. You might also work in cooperation with the upstream ISP to begin tracking the source of the packets back to their source.

WebUI

Screening > Screen (Zone: Untrust): Enter the following, then click **Apply**:

Generate Alarms without Dropping Packet: (select)
IP Address Spoof Protection: (select)

CLI

```
set zone untrust screen alarm-without-drop
set zone untrust screen ip-spoofing
save
```

Chapter 2

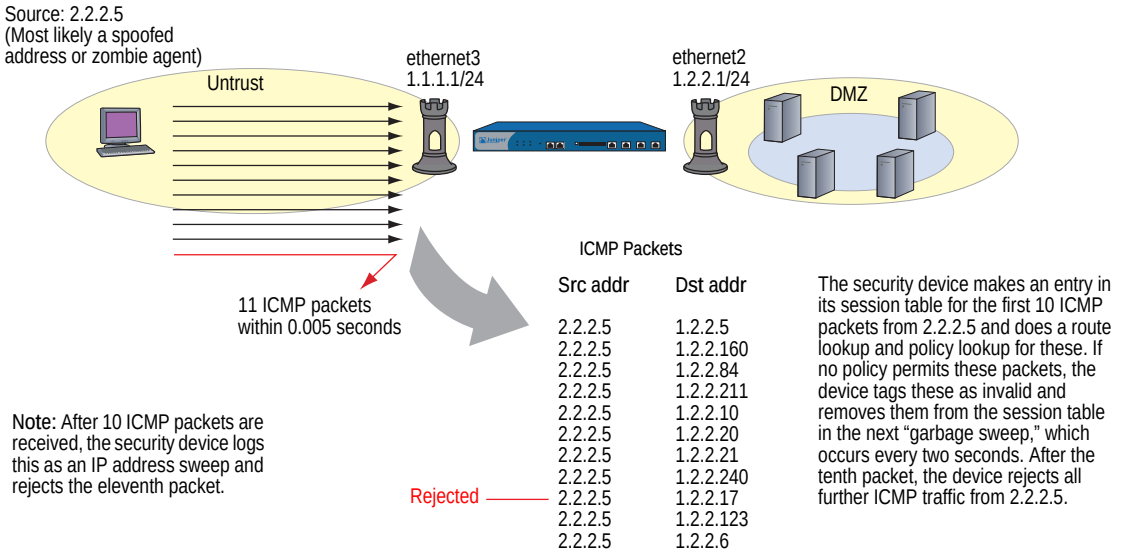
Reconnaissance Deterrence

Attackers can better plan their attack when they first know the layout of the targeted network (which IP addresses have active hosts), the possible entry points (which port numbers are active on the active hosts), and the constitution of their victims (which operating system the active hosts are running). To gain this information, attackers must perform reconnaissance. Juniper Networks provides several SCREEN options to deter attackers' reconnaissance efforts and thereby hinder them from obtaining valuable information about the protected network and network resources.

IP Address Sweep

An address sweep occurs when one source IP address sends 10 ICMP packets to different hosts within a defined interval (5000 microseconds is the default). The purpose of this scheme is to send ICMP packets—typically echo requests—to various hosts in the hopes that at least one replies, thus uncovering an address to target. The security device internally logs the number of ICMP packets to different addresses from one remote source. Using the default settings, if a remote host sends ICMP traffic to 10 addresses in 0.005 seconds (5000 microseconds), the security device flags this as an address sweep attack, and rejects all further ICMP echo requests from that host for the remainder of the specified threshold time period. The device detects and drops the tenth packet that meets the address sweep attack criterion.

Figure 1: Address Sweep



Consider enabling this SCREEN option for a security zone only if there is a policy permitting ICMP traffic from that zone. Otherwise, you do not need to enable it. The lack of such a policy denies all ICMP traffic from that zone, precluding an attacker from successfully performing an IP address sweep anyway.

To block IP address sweeps originating in a particular security zone, do either of the following:

WebUI

Screening > Screen (Zone: select a zone name): Enter the following, then click **Apply**:

IP Address Sweep Protection: (select)
Threshold: (enter a value to trigger IP address sweep protection)

NOTE: The value unit is microseconds. The default value is 5000 microseconds.

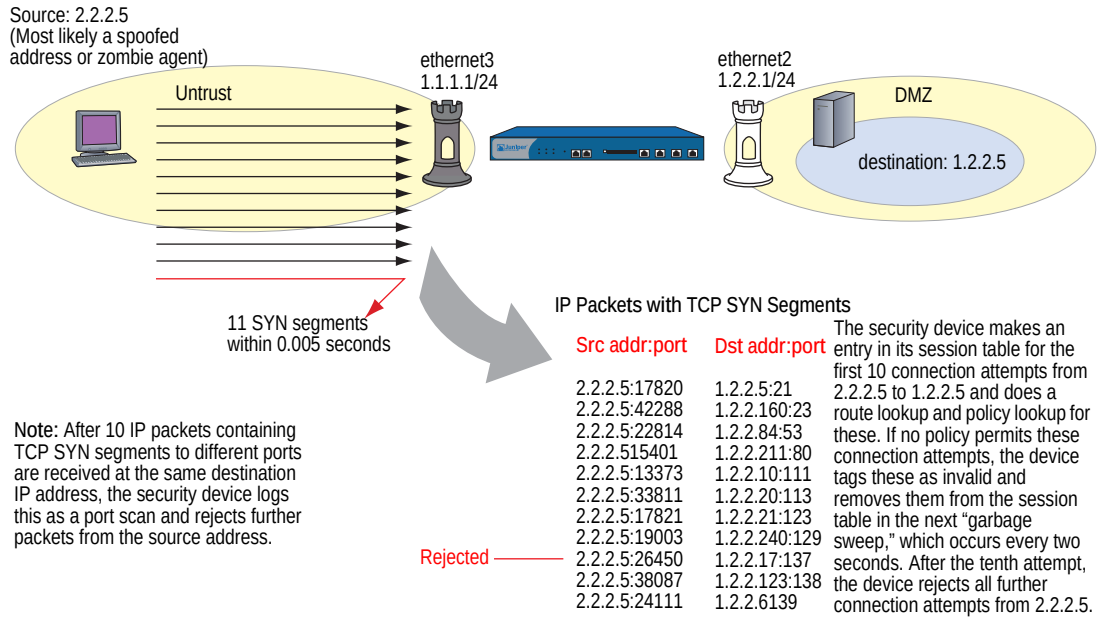
CLI

set zone *zone* screen ip-sweep threshold *number*
set zone *zone* screen ip-sweep

Port Scanning

A port scan occurs when one source IP address sends IP packets containing TCP SYN segments to 10 different ports at the same destination IP address within a defined interval (5000 microseconds is the default). The purpose of this scheme is to scan the available services in the hopes that at least one port will respond, thus identifying a service to target. The security device internally logs the number of different ports scanned from one remote source. Using the default settings, if a remote host scans 10 ports in 0.005 seconds (5000 microseconds), the device flags this as a port scan attack, and rejects all further packets from the remote source for the remainder of the specified timeout period. The device detects and drops the tenth packet that meets the port scan attack criterion.

Figure 2: Port Scan



To block port scans originating in a particular security zone, do either of the following:

WebUI

Screening > Screen (Zone: select a zone name): Enter the following, then click **Apply**:

Port Scan Protection: (select)
Threshold: (enter a value to trigger protection against port scans)

NOTE: The value unit is microseconds. The default value is 5000 microseconds.

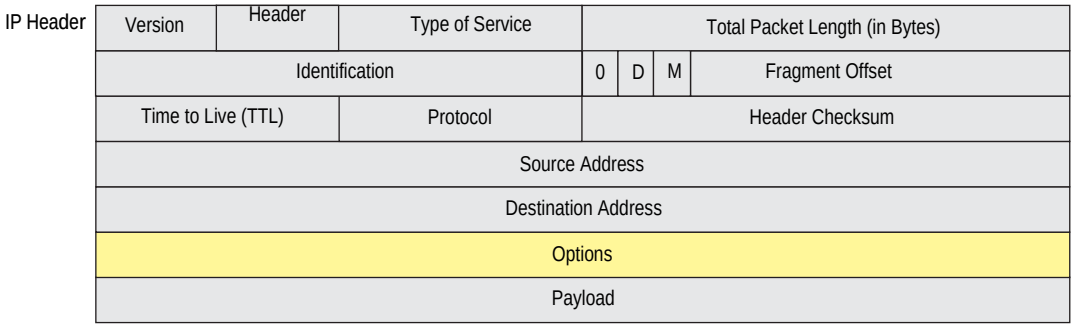
CLI

set zone *zone* screen port-scan threshold *number*
set zone *zone* screen port-scan

Network Reconnaissance Using IP Options

The Internet Protocol standard RFC 791, *Internet Protocol*, specifies a set of options to provide special routing controls, diagnostic tools, and security. These options appear after the destination address in an IP packet header, as shown in Figure 3.

Figure 3: Routing Options



RFC 791 states that these options are "unnecessary for the most common communications" and, in reality, they rarely appear in IP packet headers. When they do appear, they are frequently being put to some illegitimate use. Table 1 lists the IP options and their accompanying attributes.

Table 1: IP Options and Attributes

Type	Class	Number	Length	Intended Use	Nefarious Use
End of Options	0[1]	0	0	Indicates the end of one or more IP options.	None.
No Options	0	1	0	Indicates there are no IP options in the header.	None.
Security	0	2	11 bits	Provides a way for hosts to send security, TCC (closed user group) parameters, and Handling Restriction Codes compatible with Department of Defense (DoD) requirements. (This option, as specified in RFC 791, *Internet Protocol,* and RFC 1038, *Revised IP Security Option,* is obsolete.)	Unknown. However, because it is obsolete, its presence in an IP header is suspect.
Loose Source Route	0	3	Varies	Specifies a partial route list for a packet to take on its journey from source to destination. The packet must proceed in the order of addresses specified, but it is allowed to pass through other routers in between those specified.	Evasion. The attacker can use the specified routes to hide the true source of a packet or to gain access to a protected network. (See "IP Source Route Options" on page 23.)

Type	Class	Number	Length	Intended Use	Nefarious Use
Record Route	0	7	Varies	Records the IP addresses of the network devices along the path that the IP packet travels. The destination machine can then extract and process the route information. (Due to the size limitation of 40 bytes for both the option and storage space, this can only record up to 9 IP addresses.)	Reconnaissance. If the destination host is a compromised machine in the attacker's control, he or she can glean information about the topology and addressing scheme of the network through which the packet passed.
Stream ID	0	8	4 bits	(Obsolete) Provided a way for the 16-bit SATNET stream identifier to be carried through networks that did not support the stream concept.	Unknown. However, because it is obsolete, its presence in an IP header is suspect.
Strict Source Route	0	9	Varies	Specifies the complete route list for a packet to take on its journey from source to destination. The last address in the list replaces the address in the destination field.	Evasion. An attacker can use the specified routes to hide the true source of a packet or to gain access to a protected network. (See "IP Source Route Options" on page 23.)
Timestamp	2[2]	4		Records the time (in Universal Time[3]) when each network device receives the packet during its trip from the point of origin to its destination. The network devices are identified by IP number. This option develops a list of IP addresses of the routers along the path of the packet and the duration of transmission between each one.	Reconnaissance. If the destination host is a compromised machine in the attacker's control, he or she can glean information about the topology and addressing scheme of the network through which the packet passed.

1.The class of options identified as "0" was designed to provide extra packet or network control.
2.The class of options identified as "2" was designed diagnostics, debugging, and measurement.
3.The timestamp uses the number of milliseconds since midnight Universal Time (UT). UT is also known as Greenwich Mean Time (GMT), which is the basis for the international time standard.

The following SCREEN options detect IP options that an attacker can use for reconnaissance or for some unknown but suspect purpose:

- **Record Route:** The security device detects packets where the IP option is 7 (Record Route) and records the event in the SCREEN counters list for the ingress interface.
- **Timestamp:** The security device detects packets where the IP option list includes option 4 (Internet Timestamp) and records the event in the SCREEN counters list for the ingress interface.
- **Security:** The security device detects packets where the IP option is 2 (security) and records the event in the SCREEN counters list for the ingress interface.
- **Stream ID:** The security device detects packets where the IP option is 8 (Stream ID) and records the event in the SCREEN counters list for the ingress interface.

To detect packets with the above IP options set, do either of the following, where the specified security zone is the one from which the packets originate:

WebUI

Screening > Screen (Zone: select a zone name): Enter the following, then click **Apply**:

IP Record Route Option Detection: (select)
IP Timestamp Option Detection: (select)
IP Security Option Detection: (select)
IP Stream Option Detection: (select)

CLI

set zone *zone* screen ip-record-route
set zone *zone* screen ip-timestamp-opt
set zone *zone* screen ip-security-opt
set zone *zone* screen ip-stream-opt

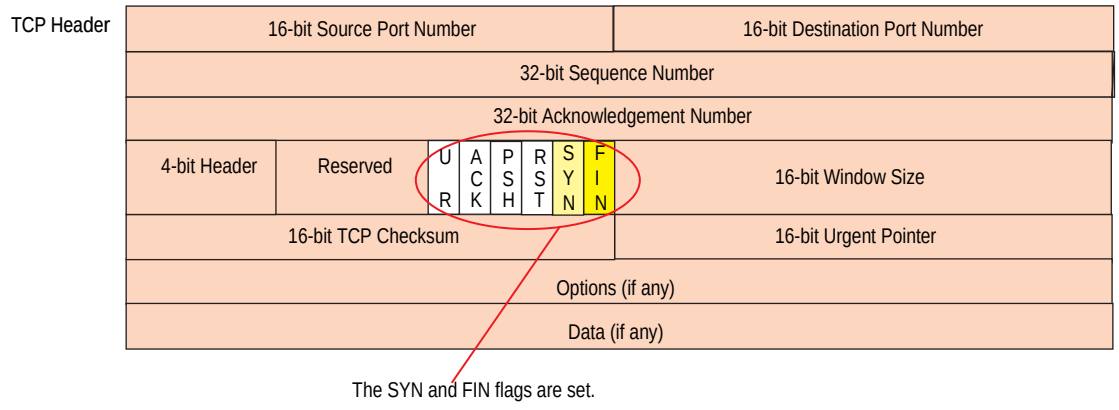
Operating System Probes

Before launching an exploit, an attacker might try to probe the targeted host to learn its operating system (OS). With that knowledge, he can better decide which attack to launch and which vulnerabilities to exploit. A Juniper Networks security device can block reconnaissance probes commonly used to gather information about OS types.

SYN and FIN Flags Set

Both the SYN and FIN control flags are not normally set in the same TCP segment header. The SYN flag synchronizes sequence numbers to initiate a TCP connection. The FIN flag indicates the end of data transmission to finish a TCP connection. Their purposes are mutually exclusive. A TCP header with the SYN and FIN flags set is anomalous TCP behavior, causing various responses from the recipient, depending on the OS. See Figure 4.

Figure 4: TCP Header with SYN and FIN Flags Set



An attacker can send a segment with both flags set to see what kind of system reply is returned and thereby determine what kind of OS is on the receiving end. The attacker can then use any known system vulnerabilities for further attacks.

When you enable this SCREEN option, the security device checks if the SYN and FIN flags are set in TCP headers. If it discovers such a header, it drops the packet.

To block packets with both the SYN and FIN flags set, do either of the following, where the specified security zone is the one from which the packets originate:

WebUI

Screening > Screen (Zone: select a zone name): Select **SYN and FIN Bits Set Protection**, then click **Apply**.

CLI

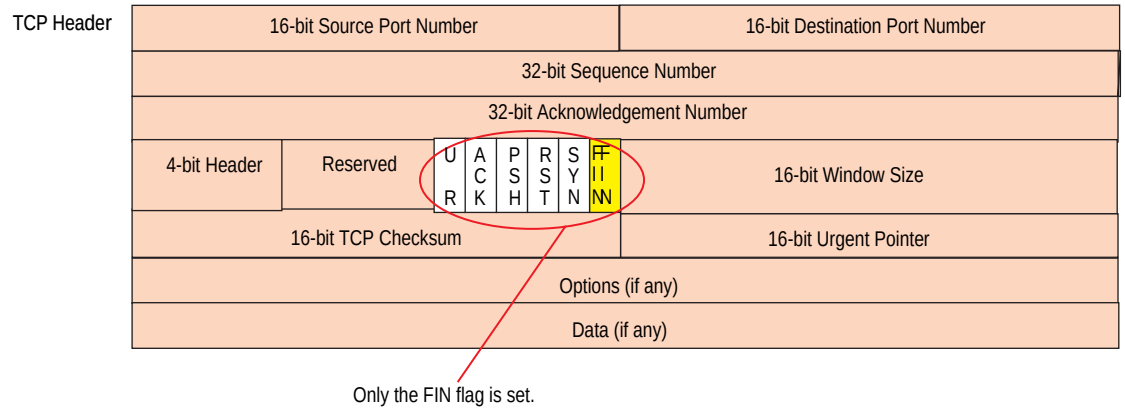
set zone *zone* screen syn-fin

FIN Flag Without ACK Flag

Figure 5 shows TCP segments with the FIN control flag set (to signal the conclusion of a session and terminate the connection). Normally, TCP segments with the FIN flag set also have the ACK flag set (to acknowledge the previous packet received). Because a TCP header with the FIN flag set but not the ACK flag is anomalous TCP behavior, there is no uniform response to this. The OS might respond by sending a TCP segment with the RST flag set. Another might completely ignore it. The victim's response can provide the attacker with a clue as to its OS. (Other purposes for sending a TCP segment with the FIN flag set are to evade detection while performing address and port scans and to evade defenses on guard for a SYN flood by performing a FIN flood instead. For information about FIN scans, see "FIN Scan" on page 15.)

NOTE: Vendors have interpreted RFC 793, *Transmission Control Protocol*, variously when designing their TCP/IP implementations. When a TCP segment arrives with the FIN flag set but not the ACK flag, some implementations send RST segments. Some drop the packet without sending an RST.

Figure 5: TCP Header with FIN Flag Set



When you enable this SCREEN option, the security device checks if the FIN flag is set but not the ACK flag in TCP headers. If it discovers a packet with such a header, it drops the packet.

To block packets with the FIN flag set but not the ACK flag, do either of the following, where the specified security zone is the one from which the packets originate:

WebUI

Screening > Screen (Zone: select a zone name): Select **FIN Bit with No ACK Bit in Flags Protection**, then click **Apply**.

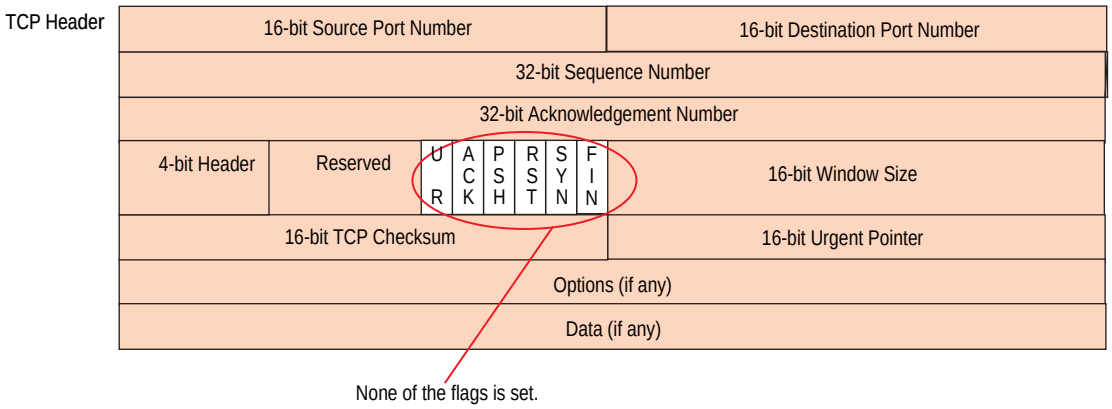
CLI

set zone *zone* screen fin-no-ack

TCP Header Without Flags Set

A normal TCP segment header has at least one flag control set. A TCP segment with no control flags set is an anomalous event. Because different operating systems respond differently to such anomalies, the response (or lack of response) from the targeted device can provide a clue as to the type of OS it is running. See Figure 6.

Figure 6: TCP Header with No Flags Set



When you enable the security device to detect TCP segment headers with no flags set, the device drops all TCP packets with a missing or malformed flags field.

To block packets with no flags set, do either of the following, where the specified security zone is the one from which the packets originate:

WebUI

Screening > Screen (Zone: select a zone name): Select **TCP Packet without Flag Protection**, then click **Apply**.

CLI

set zone *zone* screen tcp-no-flag

Evasion Techniques

Whether gathering information or launching an attack, it is generally expected that the attacker avoids detection. Although some IP address and port scans are blatant and easily detectable, more wily attackers use a variety of means to conceal their activity. Such techniques as using FIN scans instead of SYN scans—which attackers know most firewalls and intrusion detection programs detect—indicate an evolution of reconnaissance and exploit techniques to evade detection and successfully accomplish their tasks.

FIN Scan

A FIN scan sends TCP segments with the FIN flag set in an attempt to provoke a response (a TCP segment with the RST flag set) and thereby discover an active host or an active port on a host. An attacker might use this approach rather than perform an address sweep with ICMP echo requests or an address scan with SYN segments because he or she knows that many firewalls typically guard against the latter two approaches—but not necessarily against FIN segments. The use of TCP segments with the FIN flag set might evade detection and thereby help the attacker succeed in his or her reconnaissance efforts.

To thwart a FIN scan, you can do either or both of the following:

- Enable the SCREEN option that specifically blocks TCP segments with the FIN flag set but not the ACK flag, which is anomalous for a TCP segment:

  WebUI: Screening > Screen: Select the zone to which you want to apply this SCREEN option from the Zone drop-down list, and then select **FIN Bit With No ACK Bit in Flags Protection**.

  CLI: Enter **set zone** *name* **screen fin-no-ack**, in which *name* is the name of the zone to which you want to apply this SCREEN option

- Change the packet processing behavior to reject all non-SYN packets that do not belong to an existing session by entering the CLI command: **set flow tcp-syn-check**. (For more information about SYN flag checking, see "Non-SYN Flags" on page 15.)

NOTE: Changing the packet flow to check that the SYN flag is set for packets that do not belong to existing sessions also thwarts other types of non-SYN scans, such as a null scan (when no TCP flags are set).

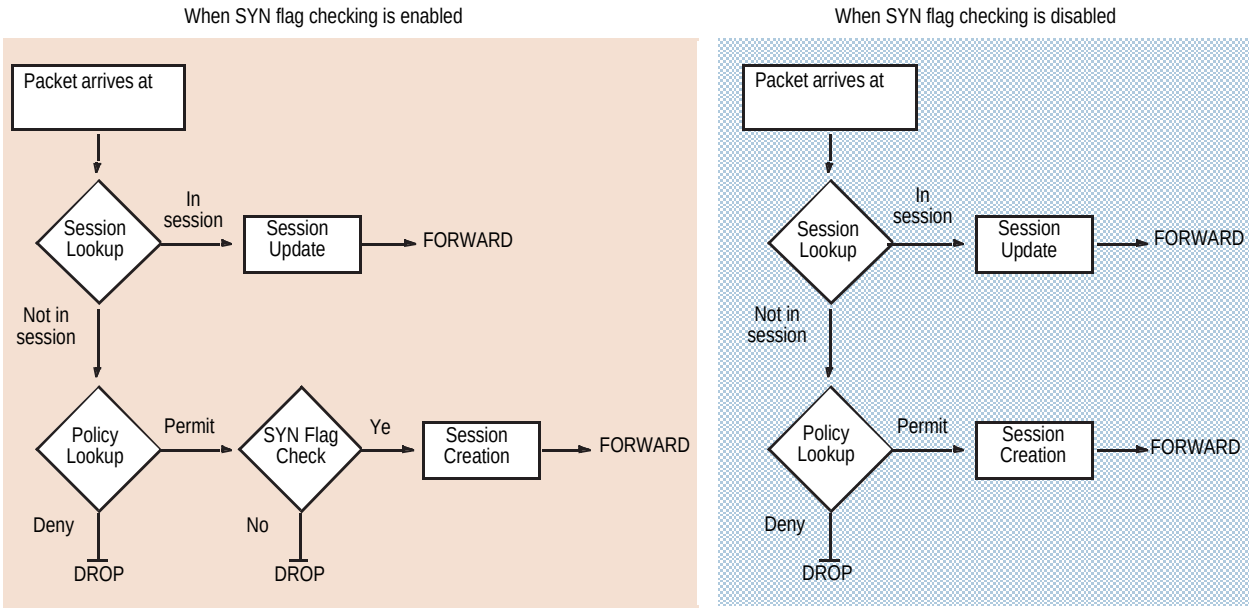
Non-SYN Flags

By default, the security device checks for SYN flags in the first packet of a session and rejects any TCP segments with non-SYN flags attempting to initiate a session. You can leave this packet flow as is or change it to so that the device does not enforce SYN flag checking before creating a session. Figure 7 on page 16 illustrates packet flow sequences when SYN flag checking is enabled and when it is disabled.

NOTE: By default, checking for the TCP SYN flag in the initial packet of a session is enabled when you install a Juniper Networks security device running ScreenOS 5.1.0 or higher. If you upgrade from a release prior to ScreenOS 5.1.0, SYN checking remains disabled by default—unless you have previously changed the default behavior.

These packet flows are the same whether the ingress interface is operating at Layer 3 (Route or NAT mode) or at Layer 2 (Transparent mode).

Figure 7: SYN Flag Checking



When the security device with SYN flag checking enabled receives a non-SYN TCP segment that does not belong to an existing session, it drops the packet and sends the source host to a TCP RST—unless the code bit of the initial non-SYN TCP packet is also RST. In that case, the security device simply drops the packet.

You can enable and disable SYN checking with the following CLI commands:

```
set flow tcp-syn-check
unset flow tcp-syn-check
```

Not checking for the SYN flag in the first packets offers the following advantages:

- **NSRP with Asymmetric Routing:** In an active/active NSRP configuration in a dynamic routing environment, a host might send the initial TCP segment with the SYN flag set to one security device (Device-A) but the SYN/ACK might be routed to the other security device in the cluster (Device-B). If this asymmetric routing occurs after Device-A has synchronized its session with Device-B, all is well. On the other hand, if the SYN/ACK response reaches Device-B before Device-A synchronizes the session and SYN checking is enabled, Device-B rejects the SYN/ACK, and the session cannot be established. With SYN checking disabled, Device-B accepts the SYN/ACK response—even though there is no existing session to which it belongs—and creates a new session table entry for it.

- **Uninterrupted Sessions:** If SYN checking is enabled and you add a security device operating in Transparent mode to a working network, it disrupts all existing sessions, which must then be restarted. For lengthy sessions, such as large data transfers or database backups, this can be a troublesome disruption. Similarly, if you reset the device or even change a component in the core section of a policy and SYN checking is enabled, all existing sessions or those sessions to which the policy change applies are disrupted and must be restarted. Disabling SYN checking avoids such disruptions to network traffic flows.

NOTE: A solution to this scenario is to install the security device with SYN checking disabled initially. Then, after a few hours—when established sessions are running through the device—enable SYN checking.

The core section in a policy contains the following main components: source and destination zones, source and destination addresses, one or more services, and an action.

However, note that the above advantages exact the following security sacrifices:

- **Reconnaissance Holes:** When an initial TCP segment with a non-SYN flag—such as ACK, URG, RST, FIN—arrives at a closed port, many operating systems (Windows, for example) respond with a TCP segment that has the RST flag set. If the port is open, then the recipient does not generate any response.

  By analyzing these responses or lack thereof, an intelligence gatherer can perform reconnaissance on the protected network and also on the ScreenOS policy set. If he sends a TCP segment with a non-SYN flag set and the policy permits it through, the destination host receiving such a segment might drop it and respond with a TCP segment that has the RST flag set. Such a response informs the perpetrator of the presence of an active host at a specific address and that the targeted port number is closed. The intelligence gatherer also learns that the firewall policy permits access to that port number on that host.

  By enabling SYN flag checking, the security device drops TCP segments without a SYN flag if they do not belong to an existing session. It does not return a TCP RST segment. Consequently, the scanner gets no replies regardless of the policy set or whether the port is open or closed on the targeted host.

- **Session Table Floods:** If SYN checking is disabled, an attacker can bypass the ScreenOS SYN flood protection feature by flooding a protected network with a barrage of TCP segments that have non-SYN flags set. Although the targeted hosts drop the packets—and possibly send TCP RST segments in reply—such a flood can fill up the session table of the security device. When the session table is full, the device cannot process new sessions for legitimate traffic.

  By enabling SYN checking and SYN flood protection, you can thwart this kind of attack. Checking that the SYN flag is set on the initial packet in a session forces all new sessions to begin with a TCP segment that has the SYN flag set. SYN flood protection then limits the number of TCP SYN segments per second so that the session table does not become overwhelmed.

NOTE: For information about session table floods, see “Session Table Flood” on page 28. For information about SYN floods, see “SYN Flood” on page 34.

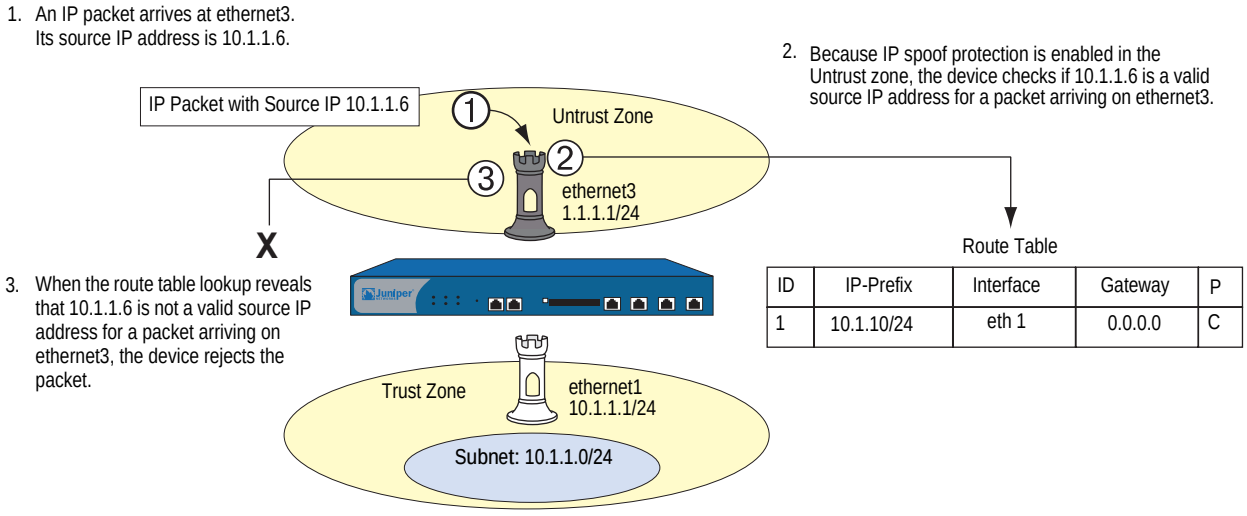
If you do not need SYN checking disabled, Juniper Networks strongly recommends that it be enabled (its default state for an initial installation of ScreenOS). You can enable it with the following command: **set flow tcp-syn-check**. With SYN checking enabled, the security device rejects TCP segments with non-SYN flags set unless they belong to an established session.

IP Spoofing

One method of attempting to gain access to a restricted area of the network is to insert a bogus source address in the packet header to make the packet appear to come from a trusted source. This technique is called IP spoofing. ScreenOS has two IP spoofing detection methods, both of which accomplish the same task: determining that the packet came from a location other than that indicated in its header. The method that a Juniper Networks security device uses depends on whether it is operating at Layer 3 or Layer 2 in the OSI Model.

- **Layer 3**—When interfaces on the security device are operating in Route or NAT mode, the mechanism to detect IP spoofing relies on route table entries. If, for example, a packet with source IP address 10.1.1.6 arrives at ethernet3, but the security device has a route to 10.1.1.0/24 through ethernet1, IP spoof checking notes that this address arrived at an invalid interface—as defined in the route table, a valid packet from 10.1.1.6 can only arrive via ethernet1, not ethernet3. Therefore, the device concludes that the packet has a spoofed source IP address and discards it.

Figure 8: Layer 3 IP Spoofing

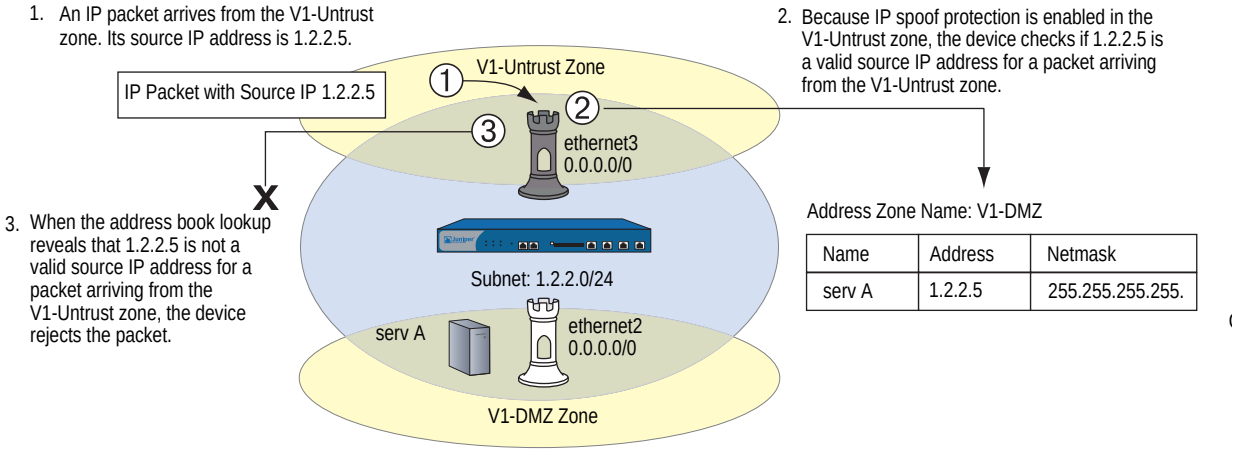


If the source IP address in a packet does not appear in the route table, by default the security device allows that packet to pass (assuming that a policy exists permitting it). Using the following CLI command—where the specified security zone is the one from which the packets originate—you can instruct the security device to drop any packet whose source IP address is not in the route table:

set zone *zone* screen ip-spoofing drop-no-rpf-route

- **Layer 2**—When interfaces on the security device are operating in Transparent mode, the IP spoof checking mechanism makes use of the address book entries. For example, you define an address for "serv A" as 1.2.2.5/32 in the V1-DMZ zone. If a packet with source IP address 1.2.2.5 arrives at a V1-Untrust zone interface (ethernet3), IP spoof checking notes that this address arrived at an invalid interface. The address belongs to the V1-DMZ zone, not to the V1-Untrust zone, and is accepted only at ethernet2, which is bound to V1-DMZ. The device concludes that the packet has a spoofed source IP address and discards it.

Figure 9: Layer 2 IP Spoofing



Be careful when defining addresses for the subnet that straddles multiple security zones. In Figure 9, 1.2.2.0/24 belongs to both the V1-Untrust and V1-DMZ zones. If you configure the security device as follows, the device will block traffic from the V1-DMZ zone that you want it to permit:

- You define an address for 1.2.2.0/24 in the V1-Untrust zone.
- You have a policy permitting traffic from any address in the V1-DMZ zone to any address in the V1-Untrust zone (**set policy from v1-dmz to v1-untrust any any any permit**).
- You enable IP spoof checking.

Because addresses in the V1-DMZ zone are also in the 1.2.2.0/24 subnet, when traffic from these addresses reaches ethernet2, the IP spoof check refers to the address book and finds 1.2.2.0/24 in the V1-Untrust zone. Consequently, the security device blocks the traffic.

Example: L3 IP Spoof Protection

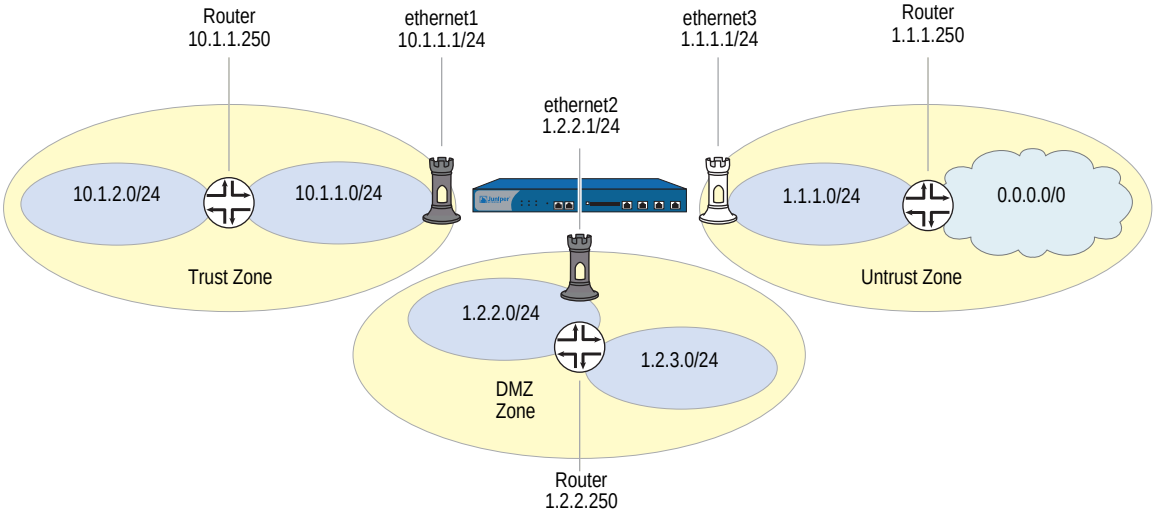
In this example, you enable IP spoof protection for the Trust, DMZ, and Untrust zones for a Juniper Networks security device operating at Layer 3. By default, the device automatically makes entries in the route table for the subnets specified in interface IP addresses. In addition to these automatic route table entries, you manually enter the three routes shown in the following table:

Destination	Egress Interface	Next Gateway
10.1.2.0/24	ethernet1	10.1.1.250
1.2.3.0/24	ethernet2	1.2.2.250
0.0.0.0/0	ethernet3	1.1.1.250

If you enable the IP spoof protection SCREEN option but do not enter the above three routes, the device drops all traffic from the addresses in the "Destination" column and enters alarms in the event log. For example, if a packet with the source address 10.1.2.5 arrives at ethernet1 and there is no route to the 10.1.2.0/24 subnet via ethernet1, the device determines that packet has arrived at an invalid interface and drops it.

All the security zones in this example are in the trust-vr routing domain.

Figure 10: Example of Layer 3 IP Spoofing



WebUI

1. **Interfaces**

   Network > Interfaces > Edit (for ethernet1): Enter the following, then click **Apply**:

   Zone Name: Trust
   Static IP: (select this option when present)
   IP Address/Netmask: 10.1.1.1/24

   Enter the following, then click **OK**:

   Interface Mode: NAT

   Network > Interfaces > Edit (for ethernet2): Enter the following, then click **OK**:

   Zone Name: DMZ
   Static IP: (select this option when present)
   IP Address/Netmask: 1.2.2.1/24

   Network > Interfaces > Edit (for ethernet3): Enter the following, then click **OK**:

   Zone Name: Untrust
   Static IP: (select this option when present)
   IP Address/Netmask: 1.1.1.1/24

2. **Routes**

   Network > Routing > Routing Entries > trust-vr New: Enter the following, then click **OK**:

   Network Address/Netmask: 10.1.2.0/24
   Gateway: (select)
   Interface: ethernet1
   Gateway IP Address: 10.1.1.250

   Network > Routing > Routing Entries > trust-vr New: Enter the following, then click **OK**:

   Network Address/Netmask: 1.2.3.0/24
   Gateway: (select)
   Interface: ethernet2
   Gateway IP Address: 1.2.2.250

   Network > Routing > Routing Entries > trust-vr New: Enter the following, then click **OK**:

   Network Address/Netmask: 0.0.0.0/0
   Gateway: (select)
   Interface: ethernet3
   Gateway IP Address: 1.1.1.250

3. **IP Spoof Protection**
   Screening > Screen (Zone: Trust): Select **IP Address Spoof Protection**, then click **Apply**.

   Screening > Screen (Zone: DMZ): Select **IP Address Spoof Protection**, then click **Apply**.

   Screening > Screen (Zone: Untrust): Select **IP Address Spoof Protection**, then click **Apply**.

CLI

1. **Interfaces**
   ```
   set interface ethernet1 zone trust
   set interface ethernet1 ip 10.1.1.1/24
   set interface ethernet1 nat
   set interface ethernet2 zone dmz
   set interface ethernet2 ip 1.2.2.1/24
   set interface ethernet3 zone untrust
   set interface ethernet3 ip 1.1.1.1/24
   ```

2. **Routes**
   ```
   set vrouter trust-vr route 10.1.2.0/24 interface ethernet1 gateway 10.1.1.250
   set vrouter trust-vr route 1.2.3.0/24 interface ethernet2 gateway 1.2.2.250
   set vrouter trust-vr route 0.0.0.0/0 interface ethernet3 gateway 1.1.1.250
   ```

3. **IP Spoof Protection**
   ```
   set zone trust screen ip-spoofing
   set zone dmz screen ip-spoofing
   set zone untrust screen ip-spoofing
   save
   ```

Example: L2 IP Spoof Protection

In this example, you protect the V1-DMZ zone from IP spoofing on traffic originating in the V1-Untrust zone. First, you define the following addresses for three webservers in the V1-DMZ zone:

- servA: 1.2.2.10
- servB: 1.2.2.20
- servC: 1.2.2.30

You then enable IP spoofing in the V1-Untrust zone.

If an attacker in the V1-Untrust zone attempts to spoof the source IP address using any of the three addresses in the V1-DMZ zone, the security device checks the address against those in the address books. When it finds that the source IP address on a packet coming from the V1-Untrust zone belongs to a defined address in the V1-DMZ zone, the device rejects the packet.

WebUI

1. **Addresses**
   Objects > Addresses > List > New: Enter the following, then click **OK**:

   Address Name: servA
   IP Address/Domain Name:
   IP/Netmask: (select), 1.2.2.10/32
   Zone: V1-DMZ

   Objects > Addresses > List > New: Enter the following, then click **OK**:

   Address Name: servB
   IP Address/Domain Name:
   IP/Netmask: (select), 1.2.2.20/32
   Zone: V1-DMZ

   Objects > Addresses > List > New: Enter the following, then click **OK**:

   Address Name: servC
   IP Address/Domain Name:
   IP/Netmask: (select), 1.2.2.30/32
   Zone: V1-DMZ

2. **IP Spoof Protection**
   Screening > Screen (Zone: V1-Trust): Select **IP Address Spoof Protection**, then click **Apply**.

CLI

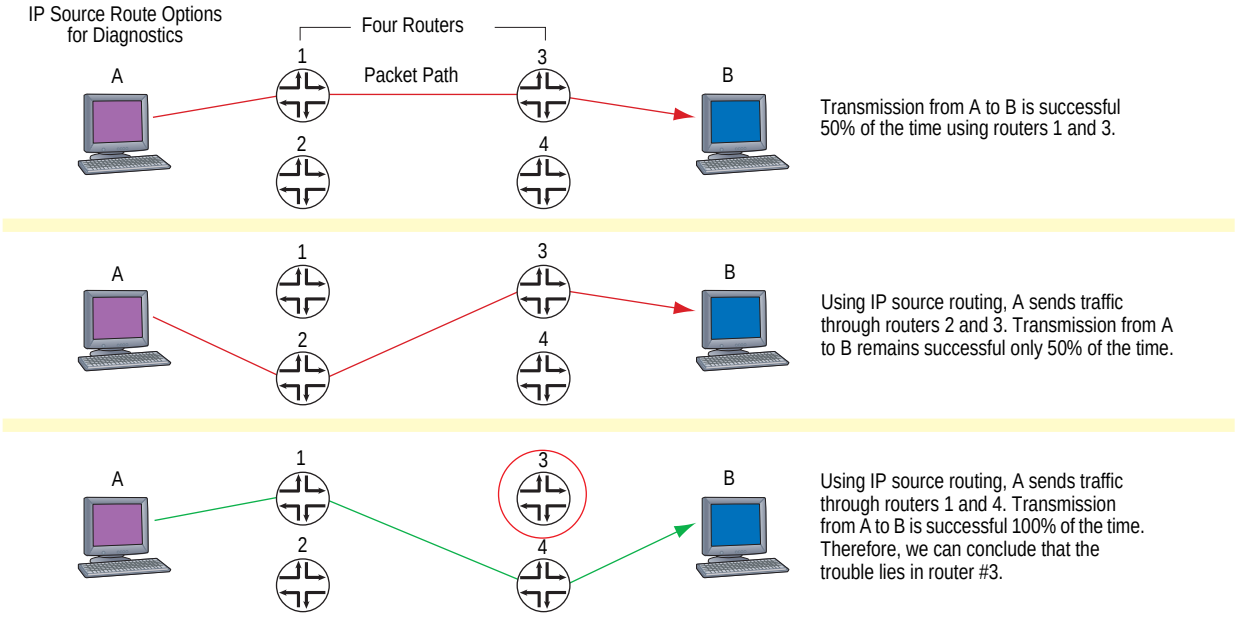
1. **Addresses**
   set address v1-dmz servA 1.2.2.10/32
   set address v1-dmz servB 1.2.2.20/32
   set address v1-dmz servC 1.2.2.30/32

2. **IP Spoof Protection**
   set zone v1-untrust screen ip-spoofing
   save

IP Source Route Options

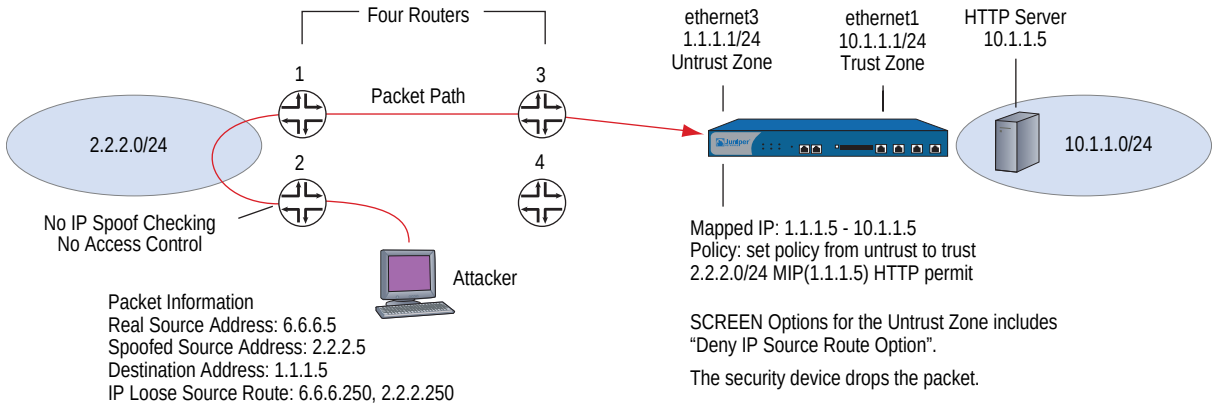
Source routing was designed to allow the user at the source of an IP packet transmission to specify the IP addresses of the routers (also referred to as "hops") along the path that he or she wants an IP packet to take on its way to its destination. The original intent of the IP source route options was to provide routing control tools to aid diagnostic analysis. If, for example, the transmission of a packet to a particular destination meets with irregular success, you might first use either the record route or timestamp IP option to discover the addresses of routers along the path or paths that the packet takes. You can then use either the loose or strict source route option to direct traffic along a specific path, using the addresses you learned from the results that the record route or timestamp options produced. By changing router addresses to alter the path and sending several packets along different paths, you can note changes that either improve or lessen the success rate. Through analysis and the process of elimination, you might be able to deduce where the trouble lies.

Figure 11: IP Source Routing



Although the uses of IP source route options were originally benign, attackers have learned to put them to more devious uses. They can use IP source route options to hide their true address and access restricted areas of a network by specifying a different path. For an example showing how an attacker can put both deceptions to use, consider the following scenario as illustrated in Figure 12.

Figure 12: Loose IP Source Route Option for Deception



The Juniper Networks security device only allows traffic 2.2.2.0/24 if it comes through ethernet1, an interface bound to the Untrust zone. Routers 3 and 4 enforce access controls but routers 1 and 2 do not. Furthermore, router 2 does not check for IP spoofing. The attacker spoofs the source address, and by using the loose source route option, directs the packet through router 2 to the 2.2.2.0/24 network and from there out router 1. Router 1 forwards it to router 3, which forwards it to the security device. Because the packet came from the 2.2.2.0/24 subnet and has a source address from that subnet, it seems to be valid. However, one remnant of the earlier chicanery remains: the loose source route option. In this example, you have enabled the "Deny IP Source Route Option" SCREEN option for the Untrust zone. When the packet arrives at ethernet3, the device rejects it.

You can enable the security device to either block any packets with loose or strict source route options set or detect such packets and then record the event in the counters list for the ingress interface. The SCREEN options are as follows:

- **Deny IP Source Route Option:** Enable this option to block all IP traffic that employs the loose or strict source route option. Source route options can allow an attacker to enter a network with a false IP address.

- **Detect IP Loose Source Route Option:** The security device detects packets where the IP option is 3 (Loose Source Routing) and records the event in the SCREEN counters list for the ingress interface. This option specifies a partial route list for a packet to take on its journey from source to destination. The packet must proceed in the order of addresses specified, but it is allowed to pass through other routers in between those specified.

- **Detect IP Strict Source Route Option:** The security device detects packets where the IP option is 9 (Strict Source Routing) and records the event in the SCREEN counters list for the ingress interface. This option specifies the complete route list for a packet to take on its journey from source to destination. The last address in the list replaces the address in the destination field.

  (For more information about all the IP options, see "Network Reconnaissance Using IP Options" on page 10.)

To block packets with either a loose or strict source route option set, do either of the following, where the specified security zone is the one from which the packets originate:

WebUI

Screening > Screen (Zone: select a zone name): Select **IP Source Route Option Filter**, then click **Apply**.

CLI

set zone *zone* screen ip-filter-src

To detect and record (but not block) packets with a loose or strict source route option set, do either of the following, where the specified security zone is the one from which the packets originate:

WebUI

Screening > Screen (Zone: select a zone name): Enter the following, then click **Apply**:

IP Loose Source Route Option Detection: (select)
IP Strict Source Route Option Detection: (select)

CLI

set zone *zone* screen ip-loose-src-route
set zone *zone* screen ip-strict-src-route

Chapter 3

Denial-of-Service (DoS) Attack Defenses

The intent of a denial-of-service (DoS) attack is to overwhelm the targeted victim with a tremendous amount of bogus traffic so that the victim becomes so preoccupied processing the bogus traffic that it is unable to process legitimate traffic. The target can be the Juniper Networks firewall, the network resources to which the firewall controls access, or the specific hardware platform or operating system (OS) of an individual host.

If a DoS attack originates from multiple source addresses, it is known as a distributed denial-of-service (DDoS) attack. Typically, the source address of a DoS attack is spoofed. The source addresses in a DDoS attack might be spoofed or the actual addresses of hosts that the attacker has previously compromised and which he or she is now using as "zombie agents" from which to launch the attack.

The security device can defend itself and the resources it protects from DoS and DDoS attacks. The following sections describe the various defense options available:

Firewall DoS Attacks

If an attacker discovers the presence of the Juniper Networks firewall, he or she might launch a denial-of-service (DoS) attack against it instead of the network behind it. A successful DoS attack against a firewall amounts to a successful DoS attack against the protected network in that it thwarts attempts of legitimate traffic to traverse the firewall. This section explains two methods that an attacker might use to fill up the session table of a Juniper Networks security device and thereby produce a DoS: Session Table Flood and SYN-ACK-ACK Proxy Flood.

Session Table Flood

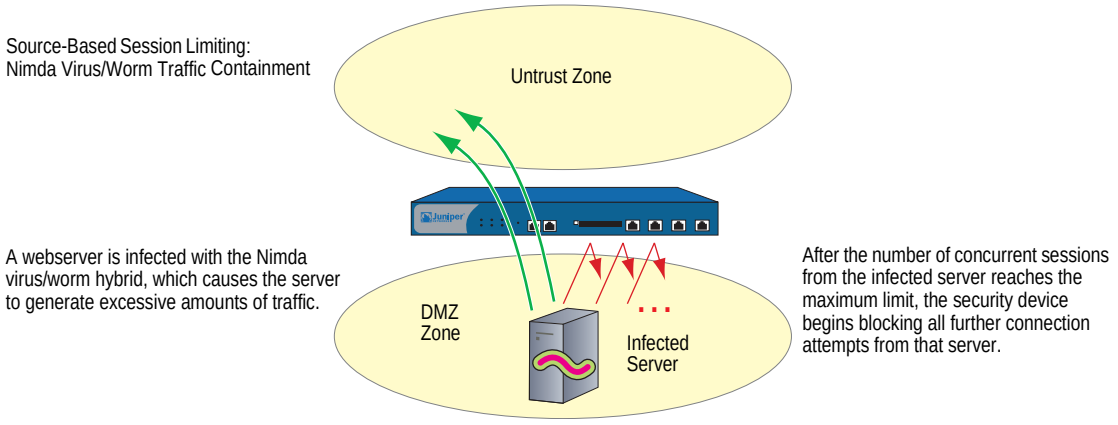
A successful DoS attack overwhelms its victim with such a massive barrage of false simulated traffic that it becomes unable to process legitimate connection requests. DoS attacks can take many forms—SYN flood, SYN-ACK-ACK flood, UDP flood, ICMP flood, and so on—but they all seek the same objective: to fill up their victim's session table. When the session table is full, that host cannot create any new sessions and begins rejecting new connection requests. The following SCREEN options help mitigate such attacks:

- Source-Based and Destination-Based Session Limits
- Aggressive Aging

Source-Based and Destination-Based Session Limits

In addition to limiting the number of concurrent sessions from the same source IP address, you can also limit the number of concurrent sessions to the same destination IP address. One benefit of setting a source-based session limit is that it can stem an attack such as the Nimda virus (which is actually both a virus and a worm) that infects a server and then begins generating massive amounts of traffic from that server. Because all the virus-generated traffic originates from the same IP address, a source-based session limit ensures that the firewall can curb such excessive amounts of traffic.

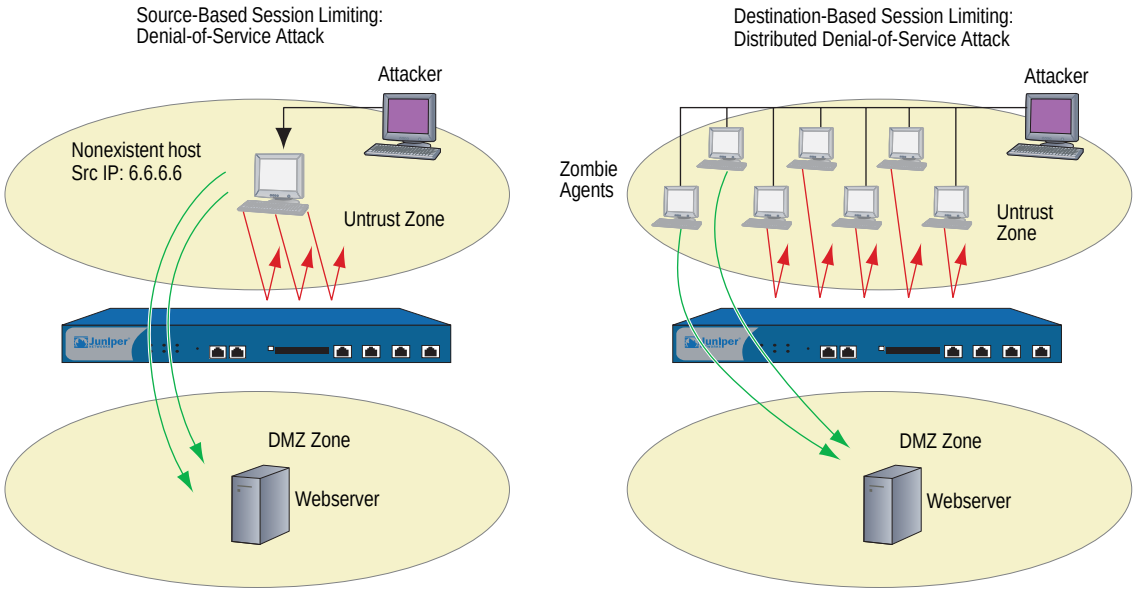
Figure 13: Limiting Sessions Based on Source IP Address



Another benefit of source-based session limiting is that it can mitigate attempts to fill up the ScreenOS session table—if all the connection attempts originate from the same source IP address. However, a wily attacker can launch a distributed denial-of-service (DDoS) attack. In a DDoS attack, the malicious traffic can come

from hundreds of hosts, known as *zombie agents*, that are surreptitiously under the control of an attacker. In addition to the SYN, UDP, and ICMP flood detection and prevention SCREEN options, setting a destination-based session limit can ensure that the security device allows only an acceptable number of concurrent connection requests—no matter what the source—to reach any one host.

Figure 14: Distributed DOS Attack



Determining what constitutes an acceptable number of connection requests requires a period of observation and analysis to establish a baseline for typical traffic flows. You also need to consider the maximum number of concurrent sessions required to fill up the session table of the particular Juniper Networks platform you are using. To see the maximum number of sessions that your session table supports, use the CLI command **get session**, and then look at the first line in the output, which lists the number of current (allocated) sessions, the maximum number of sessions, and the number of failed session allocations:

```
alloc 420/max 128000, alloc failed 0
```

The default maximum for both source- and destination-based session limits is 128 concurrent sessions, a value that might need adjustment to suit the needs of your network environment and the platform.

Example: Source-Based Session Limiting

In this example, you want to limit the amount of sessions that any one server in the DMZ and Trust zones can initiate. Because the DMZ zone only contains webservers, none of which should initiate traffic, you set the source-session limit at the lowest possible value: 1 session. On the other hand, the Trust zone contains personal computers, servers, printers, and so on, many of which do initiate traffic. For the Trust zone, you set the source-session limit maximum to 80 concurrent sessions.

WebUI

Screening > Screen (Zone: DMZ): Enter the following, then click **OK**:

Source IP Based Session Limit: (select)
Threshold: 1 Sessions

Screening > Screen (Zone: Trust): Enter the following, then click **OK**:

Source IP Based Session Limit: (select)
Threshold: 80 Sessions

CLI

```
set zone dmz screen limit-session source-ip-based 1
set zone dmz screen limit-session source-ip-based
set zone trust screen limit-session source-ip-based 80
set zone trust screen limit-session source-ip-based
save
```

Example: Destination-Based Session Limiting

In this example, you want to limit the amount of traffic to a webserver at 1.2.2.5. The server is in the DMZ zone. After observing the traffic flow from the Untrust zone to this server for a month, you have determined that the average number of concurrent sessions it receives is 2000. Based on this information, you decide to set the new session limit at 4000 concurrent sessions. Although your observations show that traffic spikes sometimes exceed that limit, you opt for firewall security over occasional server inaccessibility.

WebUI

Screening > Screen (Zone: Untrust): Enter the following, then click **OK**:

Destination IP Based Session Limit: (select)
Threshold: 4000 Sessions

CLI

```
set zone untrust screen limit-session destination-ip-based 4000
set zone untrust screen limit-session destination-ip-based
save
```

Aggressive Aging

By default, an initial TCP session 3-way handshake takes 20 seconds to time out (that is, to expire because of inactivity). After a TCP session has been established, the timeout value changes to 30 minutes. For HTTP and UDP sessions, the session timeouts are 5 minutes and 1 minute, respectively. The session timeout counter begins when a session starts and is refreshed every 10 seconds if the session is active. If a session becomes idle for more than 10 seconds, the timeout counter begins to decrement.

On certain hardware platforms, ScreenOS provides a mechanism for accelerating the timeout process when the number of sessions in the session table surpasses a specified high-watermark threshold. This feature is not available on the high-end systems.

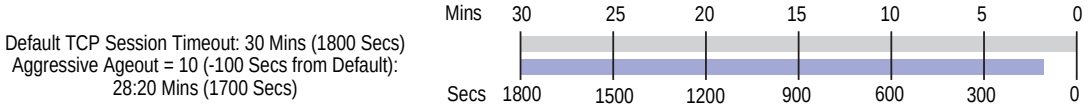
When the number of sessions dips below a specified low-watermark threshold, the timeout process returns to normal. During the period when the aggressive aging out process is in effect, a security device ages out the oldest sessions first, using the aging out rate that you specify. These aged-out sessions are tagged as invalid and are removed in the next "garbage sweep," which occurs every 2 seconds.

The aggressive ageout option shortens default session timeouts by the amount you enter. When you set and enable the aggressive ageout option, the normal session timeout value displayed in the configuration remains unchanged—1800 seconds for TCP, 300 seconds for HTTP, and 60 seconds for UDP sessions. However, when the aggressive ageout period is in effect, these sessions time out earlier—by the amount you specify for early ageout—instead of counting down all the way to zero.

The aggressive ageout value can be between 2 and 10 units, where each unit represents a 10-second interval (that is, the aggressive ageout setting can be between 20 and 100 seconds). The default setting is 2 units, or 20 seconds. If you define the aggressive ageout setting at 100 seconds, for example, you shorten the TCP and HTTP session timeouts as follows:

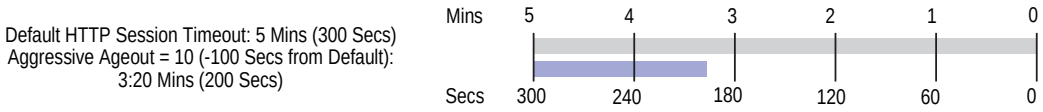
- TCP: The session timeout value shortens from 1800 seconds (30 minutes) to 1700 seconds (28:20 minutes) during the time when the aggressive aging process is in effect. During that period, the security device automatically deletes all TCP sessions whose timeout value has passed 1700 seconds, beginning with the oldest sessions first.

Figure 15: TCP Session Timeout



- HTTP: The session timeout value shortens from 300 seconds (5 minutes) to 200 seconds (3:20 minutes) during the time when the aggressive aging process is in effect. During that period, the security device automatically deletes all HTTP sessions whose timeout value has passed 200 seconds, beginning with the oldest sessions first.

Figure 16: HTTP Session Timeout

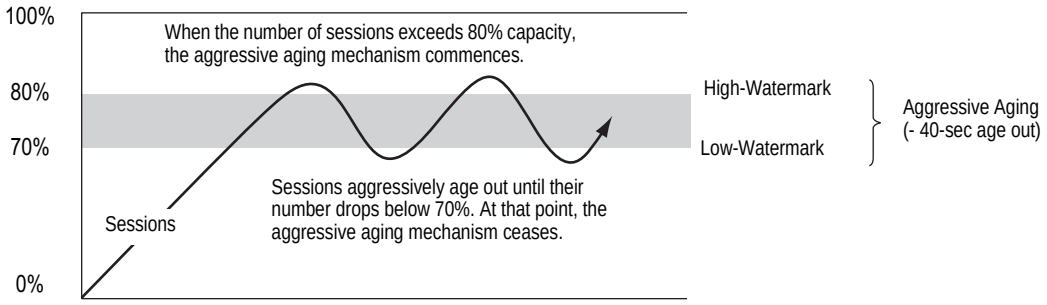


- UDP: Because the default UDP session timeout is 60 seconds, defining an early ageout setting at 100 seconds causes all UDP sessions to ageout and be marked for deletion in the next garbage sweep.

Example: Aggressively Aging Out Sessions

In this example, you set the aggressive aging out process to commence when traffic exceeds a high-watermark of 80 percent and cease when it retreats below a low-watermark of 70 percent. You specify 40 seconds for the aggressive age-out interval. When the session table is more than 80 percent full (the high-mark threshold), the security device decreases the timeout for all sessions by 40 seconds and begins aggressively aging out the oldest sessions until the number of sessions in the table is under 70 percent (the low-mark threshold).

Figure 17: Aging Out Sessions Aggressively



WebUI

NOTE: You must use the CLI to configure the aggressive age-out settings.

CLI

```
set flow aging low-watermark 70
set flow aging high-watermark 80
set flow aging early-ageout 4
save
```

SYN-ACK-ACK Proxy Flood

When an authentication user initiates a Telnet or FTP connection, the user sends a SYN segment to the Telnet or FTP server. The security device intercepts the SYN segment, creates an entry in its session table, and proxies a SYN-ACK segment to the user. The user then replies with an ACK segment. At that point, the initial three-way handshake is complete. The device sends a login prompt to the user. If the user, with malicious intent, does not log in, but instead continues initiating SYN-ACK-ACK sessions, the ScreenOS session table can fill up to the point where the device begins rejecting legitimate connection requests.

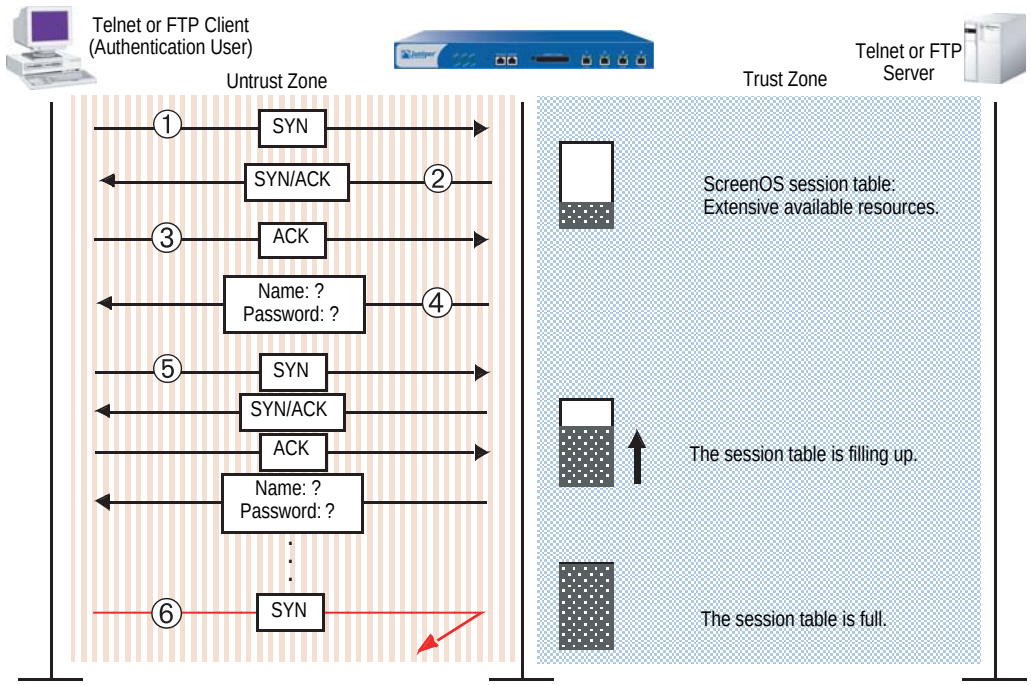
See Figure 18 for a step-by-step process:

1. The client sends a SYN segment to the server.
2. The security device proxies a SYN/ACK segment.
3. The client responds with an ACK segment.
4. The security device prompts the client (auth user) to log in.

5. The client ignores the login prompt and keeps repeating steps 1—4 until the session table is full.

6. Because the session table is full, the security device must reject all further connection requests.

Figure 18: SYN-ACK-ACK Proxy Flood



To prevent such an attack, you can enable the SYN-ACK-ACK proxy protection SCREEN option. After the number of connections from the same IP address reaches the SYN-ACK-ACK proxy threshold, the security device rejects further connection requests from that IP address. By default, the threshold is 512 connections from any single IP address. You can change this threshold (to any number between 1 and 250,000) to better suit the requirements of your network environment.

To enable protection against a SYN-ACK-ACK proxy flood, do either of the following, where the specified zone is that in which the attack originates:

WebUI

Screening > Screen (Zone: select a zone name): Enter the following, then click **Apply**:

SYN-ACK-ACK Proxy Protection: (select)
Threshold: (enter a value to trigger SYN-ACK-ACK proxy flood protection)

NOTE: The value unit is connections per source address. The default value is 512 connections from any single address.

CLI

set zone *zone* screen syn-ack-ack-proxy threshold *number*
set zone *zone* screen syn-ack-ack-proxy

Network DoS Attacks

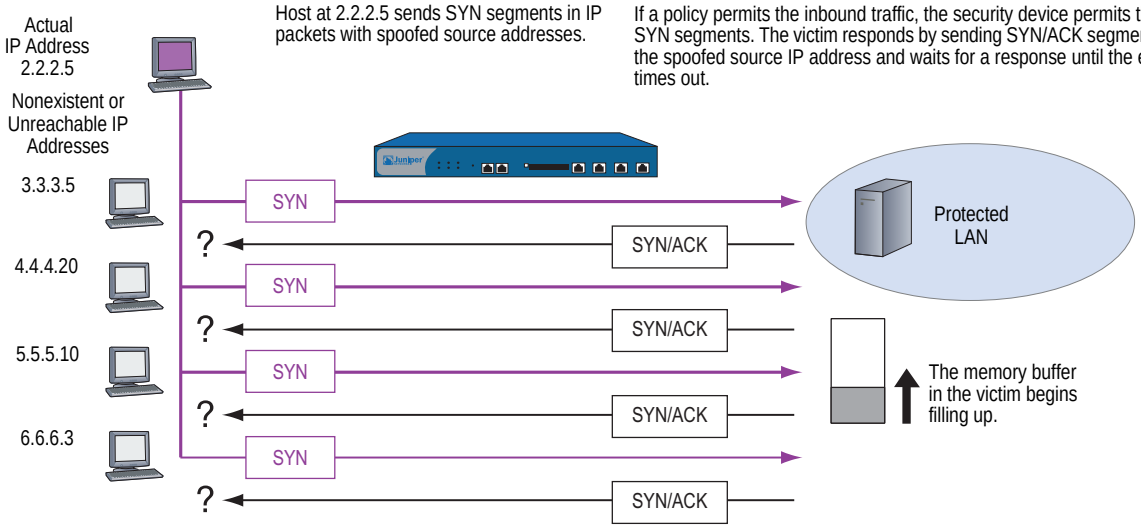
A denial-of-service (DoS) attack directed against one or more network resources floods the target with an overwhelming number of SYN, ICMP, or UDP packets, or with an overwhelming number of SYN fragments. Depending on the attacker's purpose and the extent and success of previous intelligence gathering efforts, the attacker might single out a specific host, such as a router or server; or he or she might aim at random hosts across the targeted network. Either approach has the potential of upsetting service to a single host or to the entire network, depending on how critical the role of the victim is to the rest of the network.

SYN Flood

A SYN flood occurs when a host becomes so overwhelmed by SYN segments initiating incomplete connection requests that it can no longer process legitimate connection requests.

Two hosts establish a TCP connection with a triple exchange of packets known as a *three-way handshake*: A sends a SYN segment to B; B responds with a SYN/ACK segment; and A responds with an ACK segment. A SYN flood attack inundates a site with SYN segments containing forged (spoofed) IP source addresses with nonexistent or unreachable addresses. B responds with SYN/ACK segments to these addresses and then waits for responding ACK segments. Because the SYN/ACK segments are sent to nonexistent or unreachable IP addresses, they never elicit responses and eventually time out.

Figure 19: SYN Flood Attack

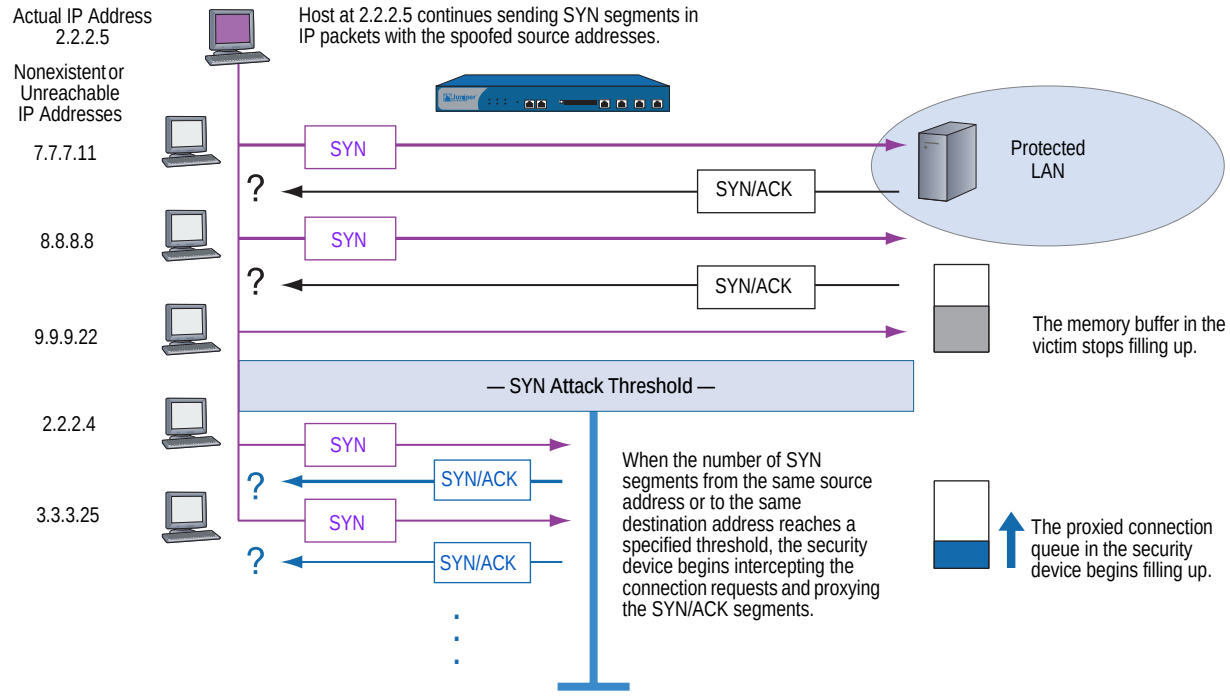


By flooding a host with incomplete TCP connections, the attacker eventually fills the memory buffer of the victim. Once this buffer is full, the host can no longer process new TCP connection requests. The flood might even damage the victim's operating system. Either way, the attack disables the victim and its normal operations.

SYN Flood Protection

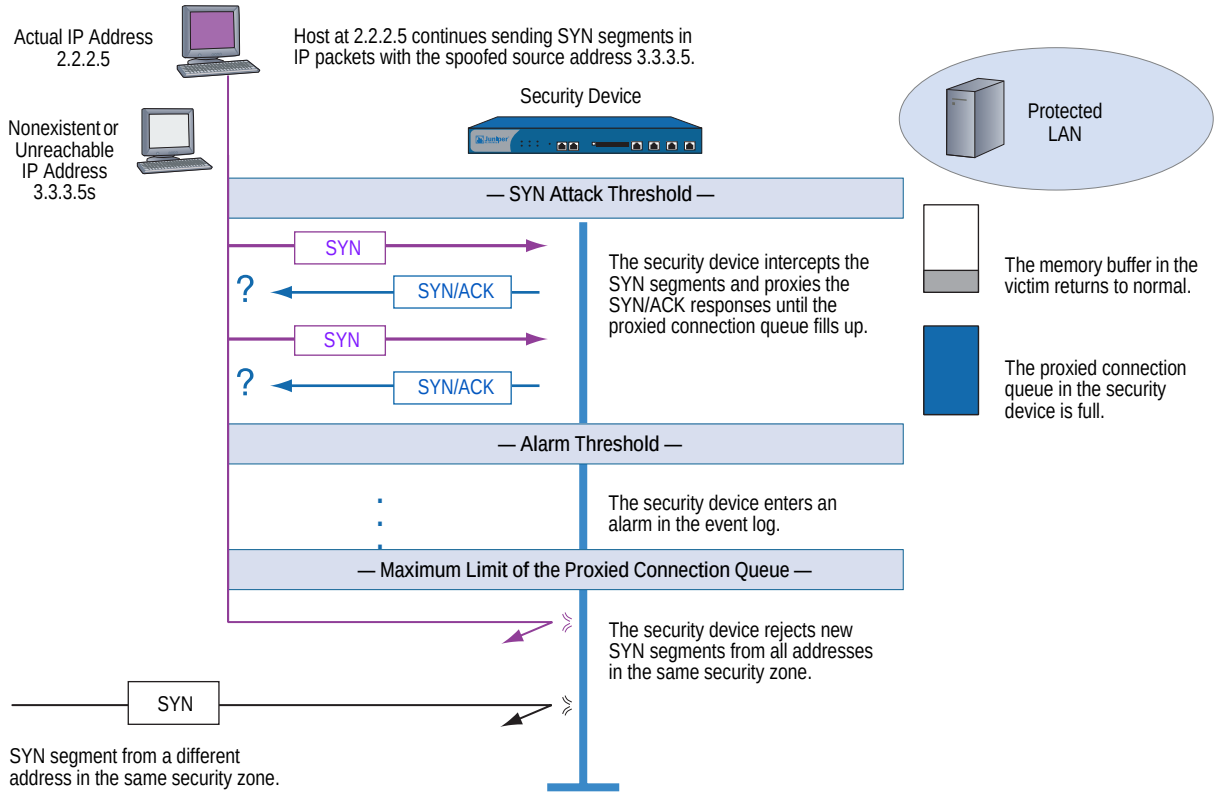
Juniper Networks security devices can impose a limit on the number of SYN segments permitted to pass through the firewall per second. You can base the attack threshold on the destination address and port, the destination address only, or the source address only. When the number of SYN segments per second exceeds one of these thresholds, the security device starts proxying incoming SYN segments, replying with SYN/ACK segments and storing the incomplete connection requests in a connection queue. The incomplete connection requests remain in the queue until the connection is completed or the request times out. In Figure 20, the SYN attack threshold has been passed, and the device has started proxying SYN segments.

Figure 20: Proxying SYN Segments



In Figure 21, the proxied connection queue has completely filled up, and the security device is rejecting new incoming SYN segments. This action shields hosts on the protected network from the bombardment of incomplete three-way handshakes.

Figure 21: Rejecting New SYN Segments



The security device starts receiving new SYN packets when the proxy queue drops below the maximum limit.

NOTE: The procedure of proxying incomplete SYN connections above a set threshold pertains only to traffic permitted by existing policies. Any traffic for which a policy does not exist is automatically dropped.

To enable the SYN flood protection SCREEN option and define its parameters, do either of the following, where the specified zone is that in which a flood might originate:

WebUI

Screening > Screen (Zone: select a zone name): Enter the following, then click **Apply**:

SYN Flood Protection: (select to enable)

Threshold: (enter the number of SYN packets—that is, TCP segments with the SYN flag set—per second required to activate the SYN proxying mechanism)

Alarm Threshold: (enter the number of proxied TCP connection requests required to write an alarm in the event log)

Source Threshold: (enter the number SYN packets per second from a single IP address required for the security device to begin rejecting new connection requests from that source)

Destination Threshold: (enter the number SYN packets per second to a single IP address required for the security device to begin rejecting new connection requests to that destination)

Timeout Value: (enter the length of time in seconds that the security device holds an incomplete TCP connection attempt in the proxied connection queue)

Queue Size: (enter the number of proxied TCP connection requests held in the proxied connection queue before the security device starts rejecting new connection requests)

NOTE: For more details about each of these parameters, see the descriptions in the following CLI section.

CLI

To enable SYN flood protection:

set zone *zone* **screen syn-flood**

You can set the following parameters for proxying uncompleted TCP connection requests:

- **Attack Threshold:** The number of SYN segments (that is, TCP segments with the SYN flag set) to the same destination address and port number per second required to activate the SYN proxying mechanism. Although you can set the threshold at any number, you need to know the normal traffic patterns at your site to set an appropriate threshold for it. For example, if it is an e-business site that normally gets 20,000 SYN segments per second, you might want to set the threshold at 30,000/second. If a smaller site normally gets 20 SYN segments/second, you might consider setting the threshold at 40.

  set zone *zone* **screen syn-flood attack-threshold** *number*

- **Alarm Threshold:** The number of proxied, half-complete TCP connection requests per second after which the security device enters an alarm in the event log. The value you set for an alarm threshold triggers an alarm when the number of proxied, half-completed connection requests to the same destination address and port number per second exceeds that value. For example, if you set the SYN attack threshold at 2000 SYN segments per second and the alarm at 1000, then a total of 3001 SYN segments to the same destination address and port number per second is required to trigger an alarm entry in the log. More precisely:

  1. The firewall passes the first 2000 SYN segments per second that meet policy requirements.

  2. The firewall proxies the next 1000 SYN segments in the same second.

  3. The 1001st proxied connection request (or 3001st connection request in that second) triggers the alarm.

  set zone *zone* **screen syn-flood alarm-threshold** *number*

  For each SYN segment to the same destination address and port number in excess of the alarm threshold, the attack detection module generates a message. At the end of the second, the logging module compresses all similar messages into a single log entry that indicates how many SYN segments to the same destination address and port number arrived after exceeding the alarm threshold. If the attack persists beyond the first second, the event log enters an alarm every second until the attack stops.

- **Source Threshold:** This option allows you to specify the number of SYN segments received per second from a single source IP address—regardless of the destination IP address and port number—before the security device begins dropping connection requests from that source.

  set zone *zone* **screen syn-flood source-threshold** *number*

  Tracking a SYN flood by source address uses different detection parameters from tracking a SYN flood by destination address and destination port number. When you set a SYN attack threshold and a source threshold, you put both the basic SYN flood protection mechanism and the source-based SYN flood tracking mechanism in effect.

- **Destination Threshold:** This option allows you to specify the number of SYN segments received per second for a single destination IP address before the security device begins dropping connection requests to that destination. If a protected host runs multiple services, you might want to set a threshold based on destination IP address only—regardless of the destination port number.

  set zone *zone* **screen syn-flood destination-threshold** *number*

  When you set a SYN attack threshold and a destination threshold, you put both the basic SYN flood protection mechanism and the destination-based SYN flood tracking mechanism in effect.

Tracking a SYN flood by destination address uses different detection parameters from tracking a SYN flood by destination address and destination port number. Consider the following case where the security device has policies permitting FTP requests (port 21) and HTTP requests (port 80) to the same server. If the SYN flood attack threshold is 1000 packets per second (pps) and an attacker sends 999 FTP packets and 999 HTTP packets per second, neither set of packets (where a set is defined as having the same destination address and port number) activates the SYN proxying mechanism. The basic SYN flood attack mechanism tracks destination address and port number, and neither set exceeds the attack threshold of 1000 pps. However, if the destination threshold is 1000 pps, the device treats both FTP and HTTP packets with the same destination address as members of a single set and rejects the 1001st packet—FTP or HTTP—to that destination.

- **Timeout:** The maximum length of time before a half-completed connection is dropped from the queue. The default is 20 seconds, and you can set the timeout from 0–50 seconds. You might try decreasing the timeout value to a shorter length until you begin to see any dropped connections during normal traffic conditions. Twenty seconds is a very conservative timeout for a three-way handshake ACK response.

  set zone *zone* **screen syn-flood timeout** *number*

- **Queue size:** The number of proxied connection requests held in the proxied connection queue before the security device starts rejecting new connection requests. The longer the queue size, the longer the device needs to scan the queue to match a valid ACK response to a proxied connection request. This can slightly slow the initial connection establishment; however, because the time to begin data transfer is normally far greater than any minor delays in initial connection setup, users would not see a noticeable difference.

  set zone *zone* **screen syn-flood queue-size** *number*

- **Drop Unknown MAC:** When a security device detects a SYN attack, it proxies all TCP connection requests. However, a device in Transparent mode cannot proxy a TCP connection request if the destination MAC address is not in its MAC learning table. By default, a device in Transparent mode that has detected a SYN attack passes SYN packets containing unknown MAC addresses. You can use this option to instruct the device to drop SYN packets containing unknown destination MAC addresses instead of letting them pass.

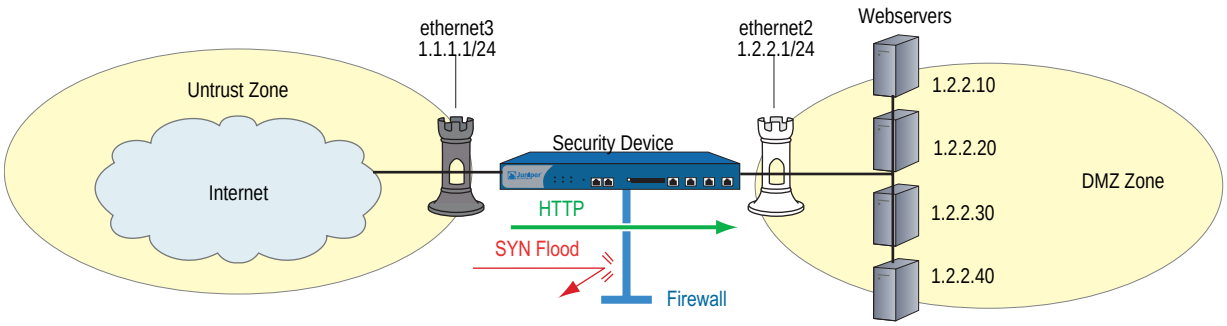
  set zone *zone* **screen syn-flood drop-unknown-mac**

Example: SYN Flood Protection

In this example, you protect four webservers in the DMZ zone from SYN flood attacks originating in the Untrust zone by enabling the SYN flood protection SCREEN option for the Untrust zone.

NOTE: We recommend that you augment the SYN flood protection that the security device provides with device-level SYN flood protection on each of the webservers. In this example, the webservers are running UNIX, which also provides some SYN flood defenses, such as adjusting the length of the connection request queue and changing the timeout period for incomplete connection requests.

Figure 22: Device-Level SYN Flood Protection



To configure the SYN flood protection parameters with appropriate values for your network, you must first establish a baseline of typical traffic flows. For one week, you run a sniffer on ethernet3—the interface bound to the Untrust zone—to monitor the number of new TCP connection requests arriving every second for the four webservers in the DMZ. Your analysis of the data accumulated from one week of monitoring produces the following statistics:

- Average number of new connection requests per server: 250/second
- Average peak number of new connection requests per server: 500/second

NOTE: A sniffer is a network-analyzing device that captures packets on the network segment to which you attach it. Most sniffers allow you to define filters to collect only the type of traffic that interests you. Later, you can view and evaluate the accumulated information. In this example, you want the sniffer to collect all TCP packets with the SYN flag set arriving at ethernet3 and destined for one of the four webservers in the DMZ.

You might want to continue running the sniffer at regular intervals to see if there are traffic patterns based on the time of day, days of the week, the time of month, or the season of the year. For example, in some organizations, traffic might increase dramatically during a critical event. Significant changes probably warrant adjusting the various thresholds.

Based on this information, you set the following SYN flood protection parameters for the Untrust zone, as shown in Table 2.

Table 2: SYN Flood Protection Parameters

Parameter	Value	Reason for Each Value
Attack Threshold	625 packets per second (pps)	This is 25% higher than the average peak number of new connection requests per second per server, which is unusual for this network environment. When the number of SYN packets per second for any one of the four webservers exceeds this number, the device begins proxying new connection requests to that server. (In other words, beginning with the 626th SYN packet to the same destination address and port number in one second, the device begins proxying connection requests to that address and port number.)
Alarm Threshold	250 pps	250 pps is 1/4 of the queue size (1000 proxied, half-completed connection requests[1]). When the device proxies 251 new connection requests in one second, it makes an alarm entry in the event log. By setting the alarm threshold somewhat higher than the attack threshold, you can avoid alarm entries for traffic spikes that only slightly exceed the attack threshold.
Source Threshold	25 pps	When you set a source threshold, the device tracks the source IP address of SYN packets, regardless of the destination address and port number. (Note that this source-based tracking is separate from the tracking of SYN packets based on destination address and destination port number that constitutes the basic SYN flood protection mechanism.) In the one week of monitoring activity, you observed that no more than 1/25 of new connection requests for all servers came from any one source within a one-second interval. Therefore, connection requests exceeding this threshold are unusual and provide sufficient cause for the device to execute its proxying mechanism. (25 pps is 1/25 of the attack threshold, which is 625 pps.) If the device tracks 25 SYN packets from the same source IP address, beginning with the 26th packet, it rejects all further SYN packets from that source for the remainder of that second and the next second as well.
Destination Threshold	0 pps	When you set a destination threshold, the device runs a separate tracking of only the destination IP address, regardless of the destination port number. Because the four webservers only receive HTTP traffic (destination port 80)—no traffic to any other destination port number reaches them—setting a separate destination threshold offers no additional advantage.
Timeout	20 seconds	Because the queue size is relatively short (1000 proxied connection requests), the default value of 20 seconds is a reasonable length of time to hold incomplete connection requests in the queue for this configuration.
Queue Size	1000 proxied, half-completed connections	1000 proxied, half-completed connection requests is twice the average peak number of new connection requests (500 pps). The device proxies up to 1000 requests per second before dropping new requests. Proxying twice the average peak number of new connection requests provides a conservative buffer for legitimate connection requests to get through.

1.Half-completed connection requests are incomplete three-way handshakes. A three-way handshake is the initial phase of a TCP connection. It consists of a TCP segment with the SYN flag set, a response with the SYN and ACK flags set, and a response to that with the ACK flag set.

WebUI

1. **Interfaces**
   Network > Interfaces > Edit (for ethernet2): Enter the following, then click **OK**:

   Zone Name: DMZ
   Static IP: (select this option when present)
   IP Address/Netmask: 1.2.2.1/24

   Network > Interfaces > Edit (for ethernet3): Enter the following, then click **OK**:

   Zone Name: Untrust
   Static IP: (select this option when present)
   IP Address/Netmask: 1.1.1.1/24

2. **Addresses**
   Objects > Addresses > List > New: Enter the following, then click **OK**:

   Address Name: ws1
   IP Address/Domain Name:
   IP/Netmask: (select), 1.2.2.10/32
   Zone: DMZ

   Objects > Addresses > List > New: Enter the following, then click **OK**:

   Address Name: ws2
   IP Address/Domain Name:
   IP/Netmask: (select), 1.2.2.20/32
   Zone: DMZ

   Objects > Addresses > List > New: Enter the following, then click **OK**:

   Address Name: ws3
   IP Address/Domain Name:
   IP/Netmask: (select), 1.2.2.30/32
   Zone: DMZ

   Objects > Addresses > List > New: Enter the following, then click **OK**:

   Address Name: ws4
   IP Address/Domain Name:
   IP/Netmask: (select), 1.2.2.40/32
   Zone: DMZ

   Objects > Addresses > Groups > (for Zone: DMZ) New: Enter the following group name, move the following addresses, then click **OK**:

   Group Name: web_servers

   Select **ws1** and use the < < button to move the address from the Available Members column to the Group Members column.

   Select **ws2** and use the < < button to move the address from the Available Members column to the Group Members column.

Select **ws3** and use the < < button to move the address from the Available Members column to the Group Members column.

Select **ws4** and use the < < button to move the address from the Available Members column to the Group Members column.

3. **Policy**
   Policies > (From: Untrust, To: DMZ) New: Enter the following, then click **OK**:

   Source Address:
   Address Book Entry: (select), Any
   Destination Address:
   Address Book Entry: (select), web_servers
   Service: HTTP
   Action: Permit

4. **Screen**
   Screening > Screen (Zone: Untrust): Enter the following, then click **Apply**:

   SYN Flood Protection: (select)
   Threshold: 625
   Alarm Threshold: 250
   Source Threshold: 25
   Destination Threshold: 0
   Timeout Value: 20
   Queue Size: 1000

NOTE: Because 20 seconds is the default setting, you do not have to set the timeout to 20 seconds unless you have previously set it to another value.

CLI

1. **Interfaces**
   set interface ethernet2 zone dmz
   set interface ethernet2 ip 1.2.2.1/24
   set interface ethernet3 zone untrust
   set interface ethernet3 ip 1.1.1.1/24

2. **Addresses**
   set address dmz ws1 1.2.2.10/32
   set address dmz ws2 1.2.2.20/32
   set address dmz ws3 1.2.2.30/32
   set address dmz ws4 1.2.2.40/32
   set group address dmz web_servers add ws1
   set group address dmz web_servers add ws2
   set group address dmz web_servers add ws3
   set group address dmz web_servers add ws4

3. **Policy**
   set policy from untrust to dmz any web_servers HTTP permit

4. **Screen**

```
set zone untrust screen syn-flood attack-threshold 625
set zone untrust screen syn-flood alarm-threshold 250
set zone untrust screen syn-flood source-threshold 25
set zone untrust screen syn-flood timeout 20
set zone untrust screen syn-flood queue-size 1000
set zone untrust screen syn-flood
save
```

NOTE: Because 20 seconds is the default setting, you do not have to set the timeout to 20 seconds unless you have previously set it to another value.

SYN Cookie

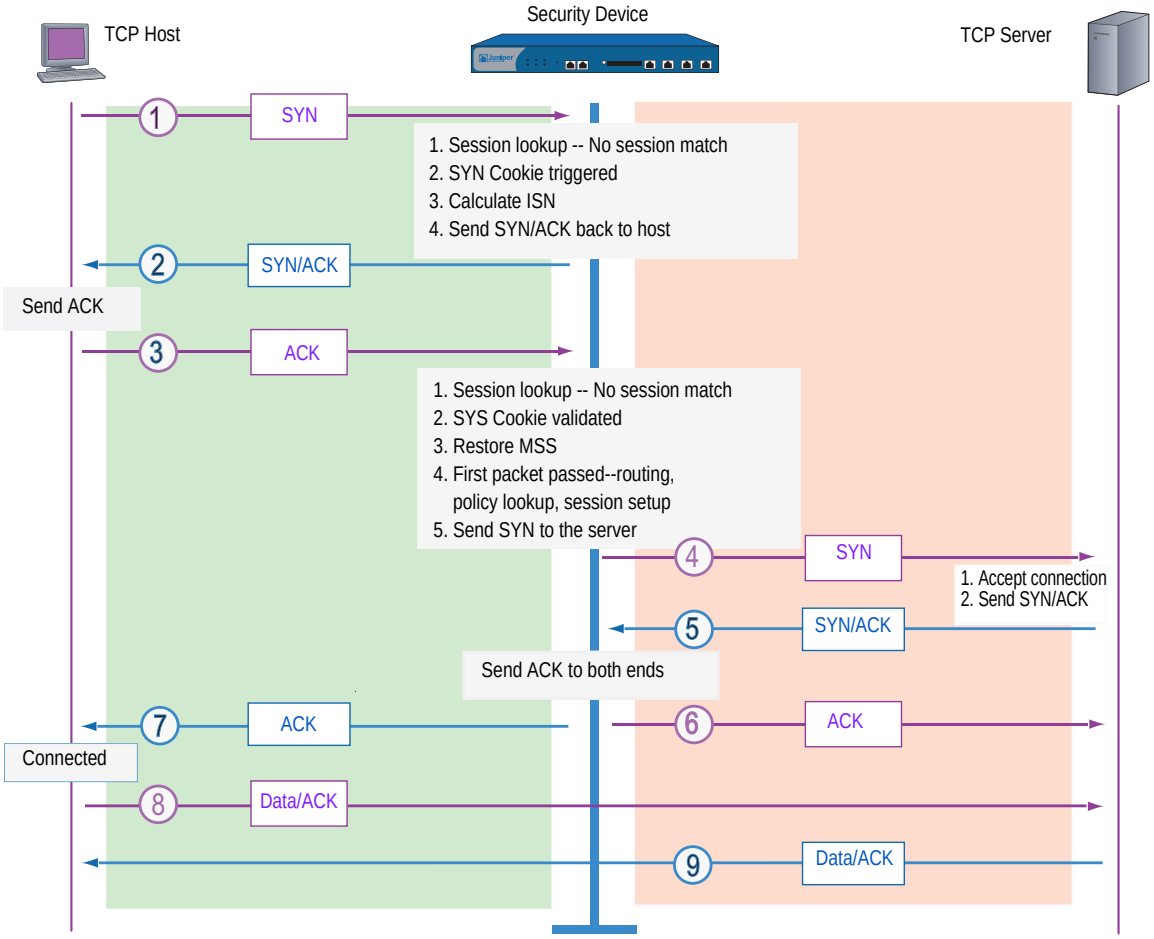
SYN Cookie is a stateless SYN proxy mechanism you can use in conjunction with the defenses against a SYN flood attack described in “SYN Flood” on page 34. Like traditional SYN proxying, SYN Cookie is activated when the SYN flood attack threshold is exceeded, but because SYN Cookie is stateless, it does not set up a session or do policy and route lookups upon receipt of a SYN segment, and maintains no connection request queues. This dramatically reduces CPU and memory usage and is the primary advantage of using SYN Cookie over the traditional SYN proxying mechanism.

When SYN Cookie is enabled on the security device and becomes the TCP-negotiating proxy for the destination server, it replies to each incoming SYN segment with a SYN/ACK containing an encrypted cookie as its Initial Sequence Number (ISN). The cookie is a MD5 hash of the original source address and port number, destination address and port number, and ISN from the original SYN packet. After sending the cookie, the device drops the original SYN packet and deletes the calculated cookie from memory. If there is no response to the packet containing the cookie, the attack is noted as an active SYN attack and is effectively stopped.

If the initiating host responds with a TCP packet containing the cookie + 1 in the TCP ACK field, the device extracts the cookie, subtracts 1 from the value, and recomputes the cookie to validate that it is a legitimate ACK. If it is legitimate, the device starts the TCP proxy process by setting up a session and sending a SYN to the server containing the source information from the original SYN. When the device receives a SYN/ACK from the server, it sends ACKs to the sever and to the initiation host. At this point the connection is established and the host and server are able to communicate directly.

Figure 23 shows how a connection is established between an initiating host and a server when SYN Cookie is active on the security device.

Figure 23: Establishing a Connection with SYN Cookie Active



To enable SYN Cookie, set a SYN flood attack threshold (as described in "SYN Flood" on page 34), and do one of the following:

WebUI

Configuration > Advanced > Flow: Enter the following, then click **Apply**:

TCP SYN-proxy SYN-cookie: (select)

CLI

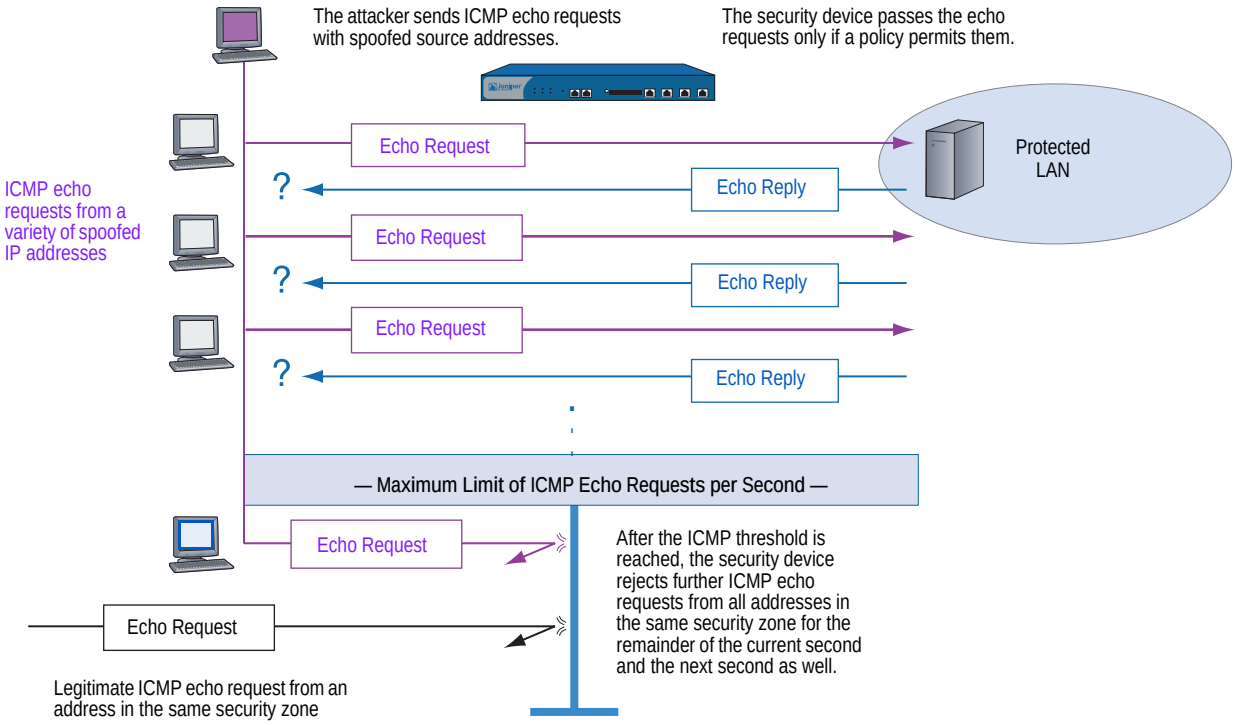
set flow syn-proxy syn-cookie

ICMP Flood

An ICMP flood typically occurs when ICMP echo requests overload its victim with so many requests that it expends all its resources responding until it can no longer process valid network traffic. When enabling the ICMP flood protection feature, you can set a threshold that once exceeded invokes the ICMP flood attack protection feature. (The default threshold value is 1000 packets per second.) If the threshold is exceeded, the security device ignores further ICMP echo requests for the remainder of that second plus the next second as well.

NOTE: An ICMP flood can consist of any type of ICMP message. Therefore, a Juniper Networks security device monitors all ICMP message types, not just echo requests.

Figure 24: ICMP Flooding



To enable ICMP flood protection, do either of the following, where the specified zone is that in which a flood might originate:

WebUI

Screening > Screen (Zone: select a zone name): Enter the following, then click **Apply**:

ICMP Flood Protection: (select)
Threshold: (enter a value to trigger ICMP flood protection)

NOTE: The value unit is ICMP packets per second. The default value is 1000 packets per second.

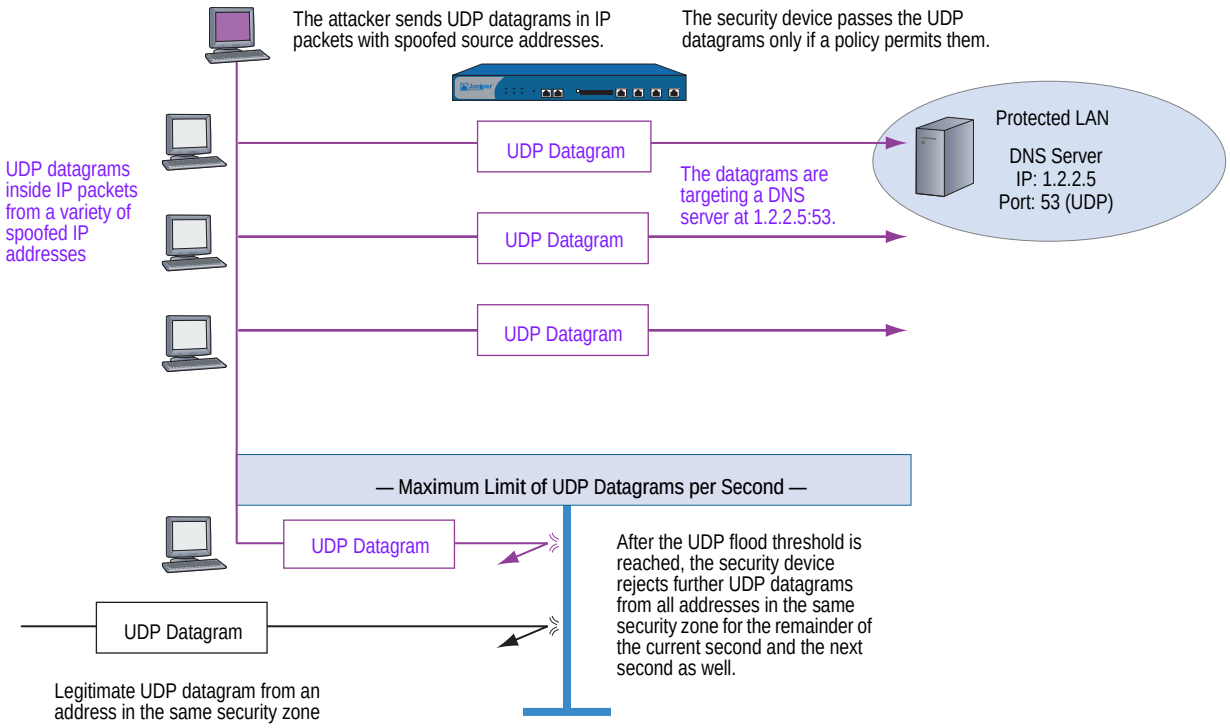
CLI

set zone *zone* screen icmp-flood threshold *number*
set zone *zone* screen icmp-flood

UDP Flood

Similar to the ICMP flood, UDP flooding occurs when an attacker sends IP packets containing UDP datagrams with the purpose of slowing down the victim to the point that it can no longer handle valid connections. After enabling the UDP flood protection feature, you can set a threshold that, once exceeded, invokes the UDP flood attack protection feature. (The default threshold value is 1000 packets per second.) If the number of UDP datagrams from one or more sources to a single destination exceeds this threshold, the security device ignores further UDP datagrams to that destination for the remainder of that second plus the next second as well.

Figure 25: UDP Flooding



To enable UDP flood protection, do either of the following, where the specified zone is that in which a flood might originate:

WebUI

Screening > Screen (Zone: select a zone name): Enter the following, then click **Apply**:

UDP Flood Protection: (select)
Threshold: (enter a value to trigger UDP flood protection)

NOTE: The value unit is UDP packets per second. The default value is 1000 packets per second.

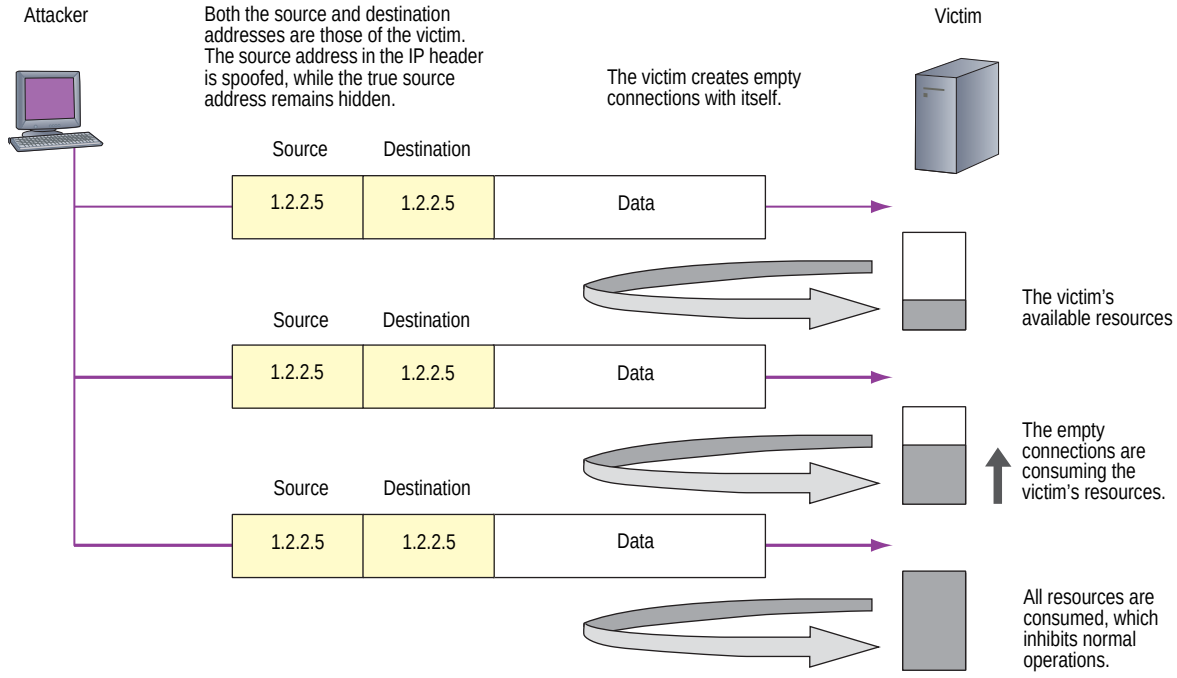
CLI

set zone *zone* screen udp-flood threshold *number*
set zone *zone* screen udp-flood

Land Attack

Combining a SYN attack with IP spoofing, a Land attack occurs when an attacker sends spoofed SYN packets containing the IP address of the victim as both the destination and source IP address. The receiving system responds by sending the SYN-ACK packet to itself, creating an empty connection that lasts until the idle timeout value is reached. Flooding a system with such empty connections can overwhelm the system, causing a denial of service.

Figure 26: Land Attack



When you enable the SCREEN option to block Land attacks, the security device combines elements of the SYN flood defense and IP spoofing protection to detect and block any attempts of this nature.

To enable protection against a Land attack, do either of the following, where the specified zone is that in which the attack originates:

WebUI

Screening > Screen (Zone: select a zone name): Select **Land Attack Protection**, then click **Apply**.

CLI

set zone *zone* screen land

OS-Specific DoS Attacks

If an attacker not only identifies the IP address and responsive port numbers of an active host but also its operating system (OS), instead of resorting to brute-force attacks, he or she can launch more elegant attacks that can produce one- or two-packet "kills." The attacks presented in this section can cripple a system with minimum effort. If your Juniper Networks security device is protecting hosts susceptible to these attacks, you can enable the security device to detect these attacks and block them before they reach their target.

Ping of Death

The maximum allowable IP packet size is 65,535 bytes, including the packet header, which is typically 20 bytes long. An ICMP echo request is an IP packet with a pseudo header, which is 8 bytes long. Therefore, the maximum allowable size of the data area of an ICMP echo request is 65,507 bytes (65,535 - 20 - 8 = 65,507).

However, many ping implementations allow the user to specify a packet size larger than 65,507 bytes. A grossly oversized ICMP packet can trigger a range of adverse system reactions such as denial of service (DoS), crashing, freezing, and rebooting.

When you enable the Ping of Death SCREEN option, the security device detects and rejects such oversized and irregular packet sizes even when the attacker hides the total packet size by purposefully fragmenting it.

NOTE: For information about IP specifications, see RFC 791, *Internet Protocol*.

For information about ICMP specifications, see RFC 792, *Internet Control Message Protocol*.

For information about Ping of Death, see http://www.insecure.org/sploits/ping-o-death.html.

Figure 27: Ping of Death

The size of this packet is 65,538 bytes. It exceeds the size limit prescribed by RFC 791, Internet Protocol, which is 65,535 bytes. As the packet is transmitted, it becomes broken into numerous fragments. The reassembly process might cause the receiving system to crash.

To enable protection against a Ping of Death attack, do either of the following, where the specified zone is that in which the attack originates:

WebUI

Screening > Screen (Zone: select a zone name): Select **Ping of Death Attack Protection**, then click **Apply**.

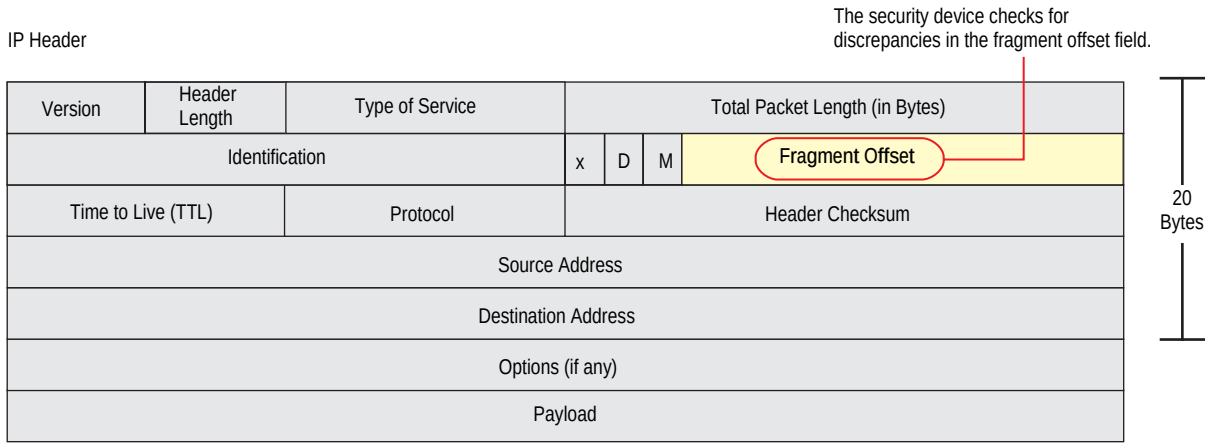
CLI

set zone *zone* screen ping-death

Teardrop Attack

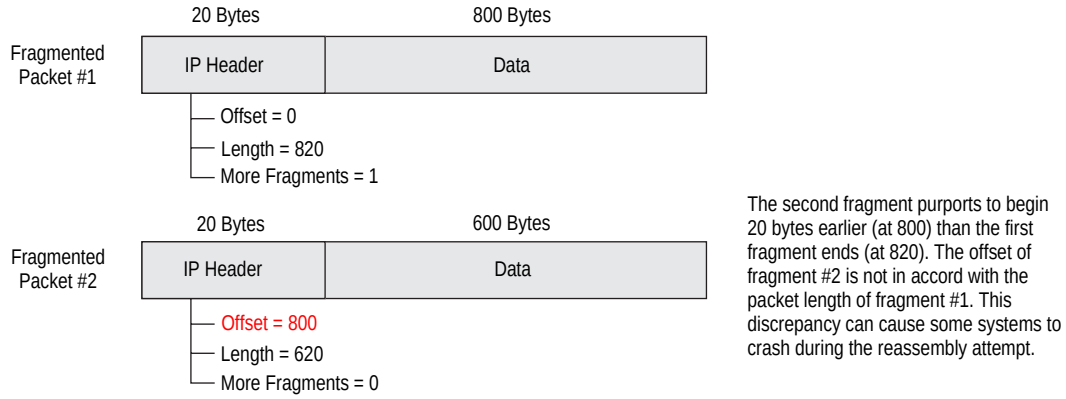
Teardrop attacks exploit the reassembly of fragmented IP packets. In the IP header, one of the fields is the fragment offset field, which indicates the starting position, or offset, of the data contained in a fragmented packet relative to the data of the original unfragmented packet.

Figure 28: Teardrop Attacks



When the sum of the offset and size of one fragmented packet differ from that of the next fragmented packet, the packets overlap, and the server attempting to reassemble the packet can crash, especially if it is running an older operating system that has this vulnerability.

Figure 29: Fragment Discrepancy



After you enable the Teardrop Attack SCREEN option, whenever the device detects this discrepancy in a fragmented packet, it drops it.

To enable protection against a Teardrop attack, do either of the following, where the specified zone is that in which the attack originates:

WebUI

Screening > Screen (Zone: select a zone name): Select **Teardrop Attack Protection**, then click **Apply**.

CLI

set zone *zone* screen tear-drop

WinNuke

WinNuke is a DoS attack targeting any computer on the Internet running Windows. The attacker sends a TCP segment—usually to NetBIOS port 139 with the urgent (URG) flag set—to a host with an established connection. This introduces a NetBIOS fragment overlap, which causes many machines running Windows to crash. After rebooting the attacked machine, the following message appears, indicating that an attack has occurred:

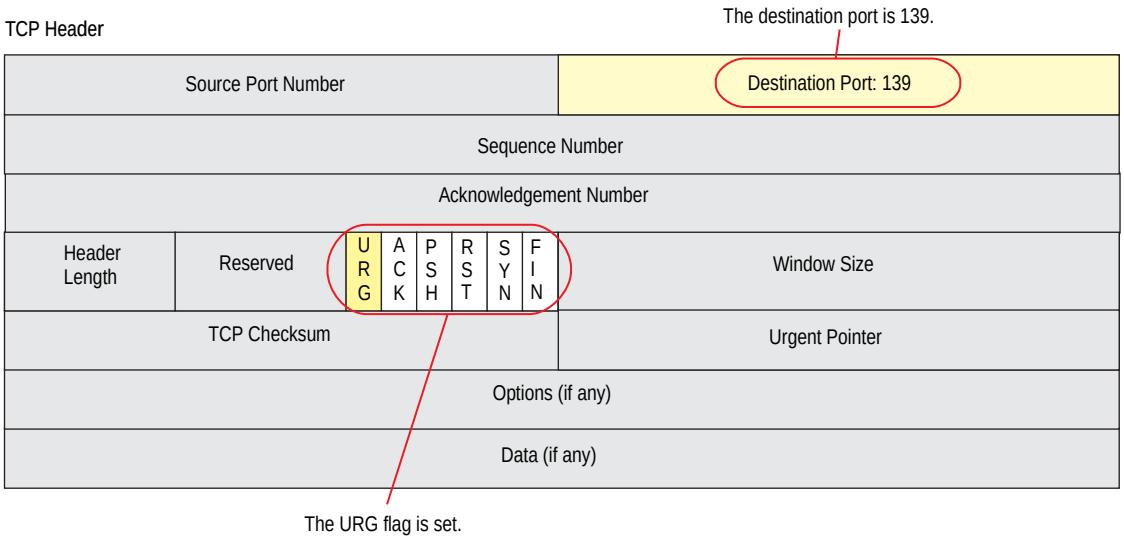
An exception 0E has occurred at 0028:[address] in VxD MSTCP(01) + 000041AE. This was called from 0028:[address] in VxD NDIS(01) + 00008660. It may be possible to continue normally.

Press any key to attempt to continue.

Press CTRL+ALT+DEL to restart your computer. You will lose any unsaved information in all applications.

Press any key to continue.

Figure 30: WinNuke Attack Indicators



If you enable the WinNuke attack defense SCREEN option, the security device scans any incoming Microsoft NetBIOS session service (port 139) packets. If the device observes that the URG flag is set in one of those packets, it unsets the URG flag, clears the URG pointer, forwards the modified packet, and makes an entry in the event log noting that it has blocked an attempted WinNuke attack.

To enable protection against a WinNuke attack, do either of the following, where the specified zone is that in which the attack originates:

WebUI

Screening > Screen (Zone: select a zone name): Select **WinNuke Attack Protection**, then click **Apply**.

CLI

set zone *zone* screen winnuke

Chapter 4
Content Monitoring and Filtering

Juniper Networks provides broad protection and control of network activity through ScreenOS features and the pairing of ScreenOS with Websense, SurfControl, Kaspersky Lab, and Trend Micro products.

This chapter describes how to configure the device to perform segment and packet reassembly, monitor HTTP traffic for malicious URLs, and communicate with other devices to perform AV scanning and web filtering. The chapter is organized into the following sections:

Fragment Reassembly

Typically, a network-forwarding device such as a router or switch does not reassemble the fragmented packets that it receives. Usually the destination host reconstructs the fragmented packets when they all arrive. The main function of a forwarding device is the efficient delivery of traffic. If the forwarding device also needs to queue, reassemble, and refragment all packets, its efficiency is adversely affected. However, passing fragmented packets through a firewall is insecure. An attacker can intentionally break up packets to conceal traffic strings that the firewall otherwise would detect and block.

ScreenOS allows you to enable fragment reassembly on a per zone basis. Doing so allows the security device to expand its ability to detect and block malicious URL strings. Fragment reassembly occurs on Application Layer Gateway (ALG)-enabled traffic only if device is configured for NAT.

Malicious URL Protection

In addition to the web-filtering feature (explained in “Redirect Web Filtering” on page 101), you can define up to 48 malicious URL string patterns per zone, each of which can be up to 64 characters long, for malicious URL protection at the zone level. With the Malicious URL blocking feature enabled, the security device examines the data payload of all HTTP packets. If it locates a URL and detects that the beginning of its string—up to a specified number of characters—matches the pattern you defined, the device blocks that packet from passing through the firewall.

A resourceful attacker, realizing that the string is known and might be guarded against, can deliberately fragment the IP packets or TCP segments to make the pattern unrecognizable during a packet-by-packet inspection. For example, if the malicious URL string is **120.3.4.5/level/50/exec**, IP fragmentation might break up the string into the following sections:

- First packet: **120**
- Second packet: **3.4.5/level/50**
- Third packet: **/exec**

Individually, the fragmented strings can pass undetected through the security device, even if you have the string defined as **120.3.4.5/level/50/exec** with a length of 20 characters. The string in the first packet—”120.”— matches the first part of the defined pattern, but it is shorter than the required length of 20 matching characters. The strings in the second and third packets do not match the beginning of the defined pattern, so would also pass without impedance.

However, if the packets are reassembled, the fragments combine to form a recognizable string that the device can block. Using the Fragment Reassembly feature, the device can buffer fragments in a queue, reassemble them into a complete packet, and then inspect that packet for a malicious URL. Depending on the results of this reassembly process and subsequent inspection, the device performs one of the following actions:

- If the device discovers a malicious URL, it drops the packet and enters the event in the log.
- If the device cannot complete the reassembly process, a time limit is imposed to age out and discard fragments.
- If the device determines that the URL is not malicious but the reassembled packet is too big to forward, the device fragments that packet into multiple packets and forwards them.
- If the device determines that the URL is not malicious and does not need to fragment it, it forwards the packet.

Application Layer Gateway

ScreenOS provides an Application Layer Gateway (ALG) for a number of protocols such as DNS, FTP, H.323, and HTTP. Of these, fragment reassembly can be an important component in the enforcement of policies involving FTP and HTTP services. The ability of the Juniper Networks firewall to screen packets for protocols such as FTP-Get and FTP-Put requires it to examine not only the packet header but also the data in the payload. For example, there might be two policies, one denying FTP-Put from the Untrust to DMZ zones and another permitting FTP-Get from the Untrust to the DMZ zones:

```
set policy from untrust to dmz any any ftp-put deny
set policy from untrust to dmz any any ftp-get permit
```

To distinguish the two types of traffic, the firewall examines the payload. If it reads **RETR** *filename*, the FTP client has sent a request to retrieve the specified file from the FTP server, and the security device allows the packet to pass. If the security device finds **STOR** *filename*, the client has sent a request to store the specified file on the server, and the device blocks the packet.

To thwart this defense, an attacker can deliberately fragment a single FTP-Put packet into two packets that contain the following text in their respective payloads:

- packet 1: **ST**
- packet 2: **OR** *filename*

When the security device inspects each packet individually, it does not find the string **STOR** *filename*, so would consequently allow both fragments to pass.

However, if the packets are reassembled, the fragments combine to form a recognizable string upon which the security device can act. Using the Fragment Reassembly feature, the device buffers the FTP fragments in a queue, reassembles them into a complete packet, and then inspects that packet for the complete FTP request. Depending on the results of this reassembly process and subsequent inspection, the device performs one of the following actions:

- If the device discovers an FTP-Put request, it drops the packet and enters the event in the log.
- If the device cannot complete the reassembly process, a time limit is imposed to age out and discard fragments.
- If the device discovers an FTP-Get request but the reassembled packet is too big to forward, the device fragments that packet into multiple packets and forwards them.
- If the device discovers an FTP-Get request and does not need to fragment it, the device then forwards the packet.

Example: Blocking Malicious URLs in Packet Fragments

In this example, you define the following three malicious URL strings and enable the malicious URL blocking option:

- Malicious URL 1
  - ID: Perl
  - Pattern: scripts/perl.exe
  - Length: 14
- Malicious URL 2
  - ID: CMF
  - Pattern: cgi-bin/phf
  - Length: 11
- Malicious URL 3
  - ID: DLL
  - Pattern: 210.1.1.5/msadcs.dll
  - Length: 18

The values for length indicate the number of characters in the pattern that must be present in a URL—starting from the first character—for a positive match. Note that for 1 and 3, not every character is required.

You then enable fragment reassembly for the detection of the URLs in fragmented HTTP traffic arriving at an Untrust zone interface.

WebUI

Screening > Mal-URL (Zone: Untrust): Enter the following, then click **OK**:

ID: perl
Pattern: /scripts/perl.exe
Length: 14

Screening > Mal-URL (Zone: Untrust): Enter the following, then click **OK**:

ID: cmf
Pattern: cgi-bin/phf
Length: 11

Screening > Mal-URL (Zone: Untrust): Enter the following, then click **OK**:

ID: dll
Pattern: 210.1.1.5/msadcs.dll
Length: 18

Network > Zones > Edit (for Untrust): Select the TCP/IP Reassembly for ALG checkbox, then click **OK**.

CLI

```
set zone untrust screen mal-url perl “scripts/perl.exe” 14
set zone untrust screen mal-url cmf “cgi-bin/phf” 11
set zone untrust screen mal-url dll “210.1.1.5/msadcs.dll” 18
set zone untrust reassembly-for-alg
save
```

Antivirus Scanning

A virus is executable code that infects or attaches itself to other executable code in order to reproduce itself. Some malicious viruses erase files or lock up systems, while other viruses merely infect files and can overwhelm the target host or network with bogus data.

Juniper Networks supports internal and external antivirus (AV) scanning on select security devices. Refer to the *ScreenOS 5.4 Release Notes* for a list of security devices and the supported AV scan engine.

You have the following two antivirus solutions for the ISG series of products:

- Internet Content Adaptation Protocol (ICAP) AV

  Use this solution for lower speeds, such as in T-3 or fractional T-3 deployments. For more details, see "External AV Scanning" on this page.

- Policy-Based Routing (PBR)

  Use this solution for higher speeds, such as in OC-3 deployments. In this scenario, PBR on the ISG offloads specific traffic to a high-end security device running the embedded AV scanner (internal AV scanner). For more details on this configuration, see "Advanced PBR Example" on page **7-**140. For more details on the embedded AV scanner, see "Internal AV Scanning" on page 61.

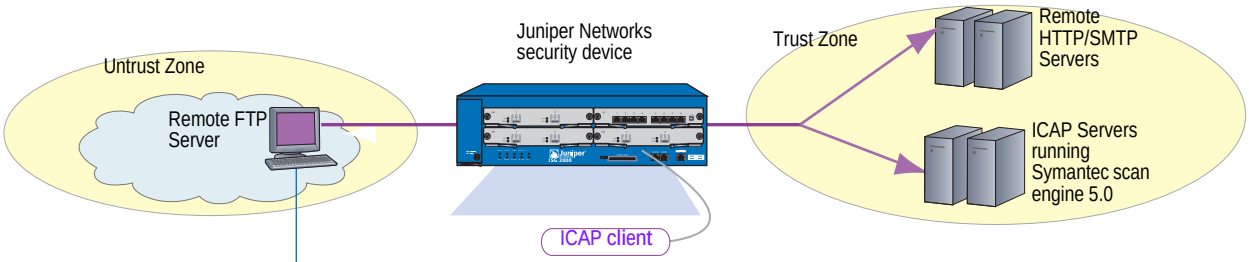
External AV Scanning

External AV scanning occurs when the security device redirects traffic to an external Internet Content Adaptation Protocol (ICAP) AV scan server. The ICAP client on the security device communicates with the external ICAP scan server to provide the following features:

- Supports ICAP v1.0 and is fully compliant with RFC 3507
- Supports Symantec Scan Engine 5.0 ICAP server
- Scalable antivirus scanning (add additional ICAP scan servers)
- Multiple security devices (firewalls) share the same ICAP scan server
- Load balancing traffic among a set of ICAP servers
- Encapsulation of HTTP and SMTP traffic
- Supports custom HTML message for HTTP traffic
- Supports custom x-response header
- Supports persistent connection to the same ICAP server
  Persistent connection reduces processing overhead and enhances AV scanning throughput.

Figure 31 illustrates how external AV scanning works with the security device.

Figure 31: How External Scanning Works



1. A client establishes an HTTP or SMTP session.
2. The AV profile in the security device determines if the data should be sent for antivirus scanning.
3. If the scanning is configured, a connection is established between the firewall and an ICAP server chosen from the ICAP scan server group.
4. The ICAP client sends the scan data (in encapsulated ICAP format) to the ICAP server and holds the data in memory until the ICAP server acknowledges receiving it.
5. The ICAP server returns the scan results with the entire content to the ICAP client.
   - If a virus is not found, the traffic is forwarded to its destination.
   - If a virus is found, the data is replaced and the destination notified. For HTTP, the data is replaced with custom HTML message. If a custom HTML message is not provided, then a log event is generated to record the virus.

Scanning Modes

After the traffic undergoes antivirus scanning, the ICAP server running Symantec Scan Engine 5.0 provides one of the following results:

- **Scan only**. Denies access to the infected file, but does nothing to the infected file.
- **Scan and delete.** Deletes all infected files, including files that are embedded in archive files without attempting to repair.
- **Scan and repair files.** Attempts to repair infected files, but does nothing to files that cannot be repaired.
- **Scan and repair or delete.** Attempts to repair infected files, and deletes any unrecoverable files from archive files.

Refer to the ICAP server documentation for more information about scanning behavior and results.

Load-Balancing ICAP Scan Servers

ScreenOS 5.4 external AV scanning allows you to load-balance ICAP scan servers configured in a ICAP server group. The load-balancing algorithm used among the ICAP scan servers in the group is least request. The ICAP servers are load-balanced based upon the server's health and traffic volume (number of pending requests). Unhealthy servers are bypassed, and traffic is reduced automatically to the overloaded server.

A configured ICAP server can be in either enabled or disabled state. The status of an enabled ICAP server can be *in-service* or *out-of-service*.When an ICAP server is configured as disabled, then the server is not used to serve new requests.

ICAP servers are monitored through a probing mechanism.For example, if the probe interval is set to 30, then an enabled ICAP server is automatically probed every 30 seconds to determine its status (in-service or out-of-service).

An auto probe returns an *out-of-service* result for the following conditions:

- Firewall cannot establish a successful TCP connection to an ICAP server
- Invalid ICAP server AV license
- Client-side error response for ICAP options request
- Server-side error response for ICAP options request

The server goes into an *out-of-service* state when three consecutive probes fail.

Internal AV Scanning

Internal AV scanning occurs when the embedded scanner in the security device scans traffic for viruses. Juniper Networks supports two embedded scan engines, Trend Micro and Juniper-Kaspersky. With a few exceptions, both scan engines support all the same antivirus features.

NOTE: The internal AV scanner requires you to install an AV license on your security device. An AV license is not required if you are using an external AV scanner.

The embedded scan engine allows you to do the following:

- Enable/disable scanning based on file extension and content type

  For example, you can set up a profile that supports scanning of executable files (.exe) but not documentation files (.doc or .pdf).

- Configure decompression layers for specific application protocols

  In each profile, you can configure different decompression levels for each protocol (HTTP/SMTP/POP3/IMAP/FTP). Based on your network environment, for example, you might want to specify the number of embedded zip files to unpack for each protocol.

- Use the Exclude option to define URL patterns for Webmail scanning

  By default, the internal AV scanner examines only predefined HTTP Webmail patterns (for example, AOL, YAHOO!, and MSN mail services). You can add or update other patterns as your environment requires.

  You can specify the optional "exclude" keyword if you want to match a pattern other than the specified URL string. For example, you would use "exclude" to examine for virus patterns in all paths except those containing the matching string.

- Configure email notification to sender/receiver on detected virus and scanning errors

- Configure scanning levels to provide spyware and phishing protection

  The Juniper-Kaspersky scan engine, by default, provides the highest level of security. In addition to stopping all viruses (including polymorphic and other advanced viruses), this scan engine also provides inbound spyware and phishing protection. (This level of security is not included in the Trend Micro scan engine.)

  Spyware protection. The spyware-protection feature adds another layer of protection to Juniper Networks anti-spyware and anti-adware solutions by letting you block incoming spyware, adware, keyloggers, and related malware to prevent it from penetrating your enterprise.

  This solution complements Juniper Networks IDP products, which provide spyware phone-home protection (that is, stopping spyware from sending sensitive data if your laptops/desktops/servers are infected).

Phishing protection. The “phishing” protection allows you to block emails that try to entice users to fake (phishing) sites that steal sensitive data from them.

You may choose to change the default security level of scanning with the following two options:

- **Basic in-the-wild scanning.** This level of scanning administers a lower degree of security by scanning the most prevalent viruses, although it provides increased performance.
- **Extended scanning.** This level of scanning includes traditionally more noisy pieces of spyware/adware in the standard scan. It provides more spyware coverage but potentially can return more false positives.

NOTE: You must use the CLI to modify the default security level of scanning.

```
set av scan-mgr pattern-type standard
```

The standard option is the default.

AV Scanning Results

AV scanning may not occur for several reasons. When your device is configured for external scanning, the device simply redirects the traffic to the external ICAP server. Refer to the ICAP server documentation for information about AV scanning behavior and results.

If your device is configured for internal AV scanning, the **get av stat** command at the CLI informs you of scanning failures. In addition to the following scan code results, an event log is generated with more information about scanning results.

```
Scan Code: Clear
Scan Code: Infect
Scan Code: Psw Archive File
Scan Code: Decompress Layer
Scan Code: Corrupt File
Scan Code: Out Of Resource
Scan Code: Internal Error
Scan Code: Error
Scan Eng: Error:
Fail Mode:
```

See “Scanning Application Protocols” on page 64 for information about AV-scanning failure, including those instances when data cannot be successfully scanned.

Refer to the *ScreenOS Messages Guide* for a list of error messages generated from AV scanning.

Policy-Based AV Scanning

AV Scanning Profiles increase the flexibility and granularity of AV scans. Profile-based scanning allows you to configure a profile to scan traffic and assign the profile to a policy. Policy-based scanning allows you to

- Select specific data traffic for AV scanning
- Enhance performance and control the AV scan engine

To configure policy-based scanning, you must configure AV profiles for use in policies by doing the following:

1. Initiate an AV profile context. For more information, see "Initiating an AV Profile for Internal AV" on page 81.

2. Configure a profile (*ns-profile* is predefined for internal AV) to examine network traffic for the following protocols:

Protocols	See
File Transfer Protocol (FTP)	"Scanning FTP Traffic" on page 65
HyperText Transfer Protocol (HTTP)	"Scanning HTTP Traffic" on page 66
Internet Mail Access Protocol (IMAP)	"Scanning IMAP and POP3 Traffic" on page 69
Post Office Protocol, version 3 (POP3)	"Scanning IMAP and POP3 Traffic" on page 69
Simple Mail Transfer Protocol (SMTP)	"Scanning SMTP Traffic" on page 71
Internet Content Adaptation Protocol (ICAP)	"Redirecting Traffic to ICAP AV Scan Servers" on page 73

3. Exit the AV profile context.

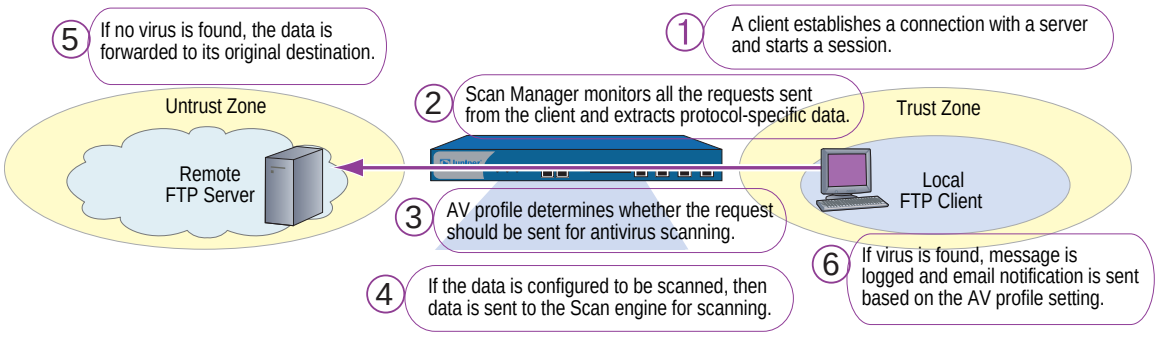
4. Assign the AV profile to a firewall policy. (Only one AV profile can be linked to a policy.)

   To apply AV protection, you reference the AV profile in a security policy. When the security device receives traffic to which a policy requiring AV scanning applies, it directs the content it receives to the AV scanner (internal or external).

5. Save your profile.

Figure 32 shows how the AV profile works with the AV scanner (internal or external).

Figure 32: How the AV Profile Works with the AV Scanner



Scanning Application Protocols

The internal embedded AV scan engine supports scanning for specific Application Layer transactions allowing you to select the content (FTP, HTTP, IMAP, POP3, or SMTP traffic) to scan. For example, scan performance can be enhanced by not scanning certain content. Similarly, external AV scanning is supported for HTTP and SMTP protocols only.

NOTE: You need to assess the risk and determine the best trade-off between security and performance.

This section discusses how to configure the following application protocols for AV scanning:

- “Scanning FTP Traffic” on page 65
- “Scanning HTTP Traffic” on page 66
- “Scanning IMAP and POP3 Traffic” on page 69
- “Scanning SMTP Traffic” on page 71

Each of the above applications can be configured for one or more of the following:

Command	Description
decompress-layer	Specifies how many layers of nested compressed files the internal AV scanner can decompress before it executes the virus scan.
extension list	Specifies the extension list name (*string*) to include or exclude defined extensions.
scan-mode	Specifies how the scan engine scans traffic for a specific protocol.
timeout	Specifies the timeout value for an AV session for a specific protocol.
http skipmime	Skips the specified MIME list from AV scanning. **Note:** Disabling **skipmime** allows the security device to scan all kinds of HTTP traffic regardless of MIME content types.
email-notify	Notifies sender or recipient of detected virus or scanning errors for IMAP, POP3, and SMTP traffic only.

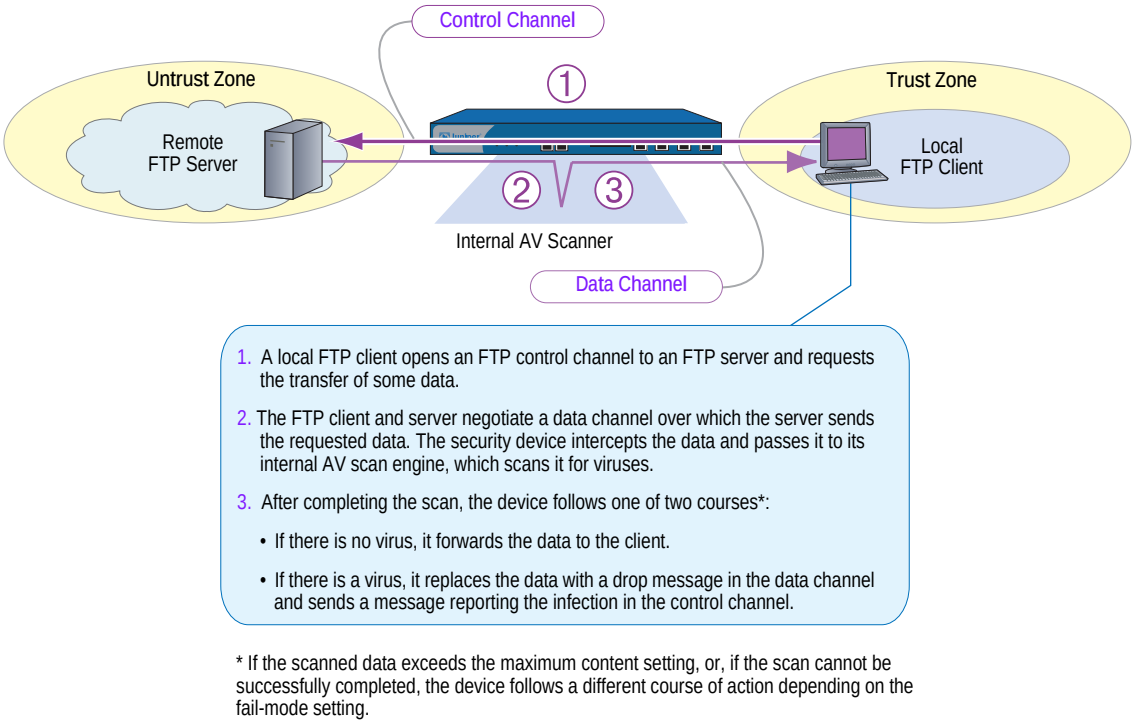
Scanning FTP Traffic

For File Transfer Protocol (FTP) traffic, the security device monitors the control channel and, when it detects one of the FTP commands for transferring data (RETR, STOR, STOU, APPE, or NLST), it scans the data sent over the data channel.

Depending on the results of the scan and how you have configured the fail-mode behavior, the security device takes one of the following actions:

If the Data	And	The Security Device
is uncontaminated		passes the data to the FTP client through the data channel
contains a virus		drops the data from the data channel and sends a virus-notification message to the FTP client through the control channel
exceeds the maximum content size	**drop** is set	drops the data from the data channel and sends a “file too large” message to the FTP client through the control channel
exceeds the maximum content size	**drop** is unset	passes the unexamined data to the FTP client through the data channel
cannot successfully be scanned	**fail mode** is unset (drop)	drops the data from the data channel and sends a “scan error” message to the FTP client through the control channel
cannot successfully be scanned	**fail mode** is permit (**traffic permit** is set)	passes the data to the FTP client through the data channel
exceeds the maximum concurrent messages	**drop** is set	drops the data from the data channel and sends an “exceeding maximum message setting” message to the FTP client through the control channel
exceeds the maximum concurrent messages	**drop** is unset	passes the data to the FTP client through the data channel

Figure 33: Antivirus Scanning for FTP Traffic



Scanning HTTP Traffic

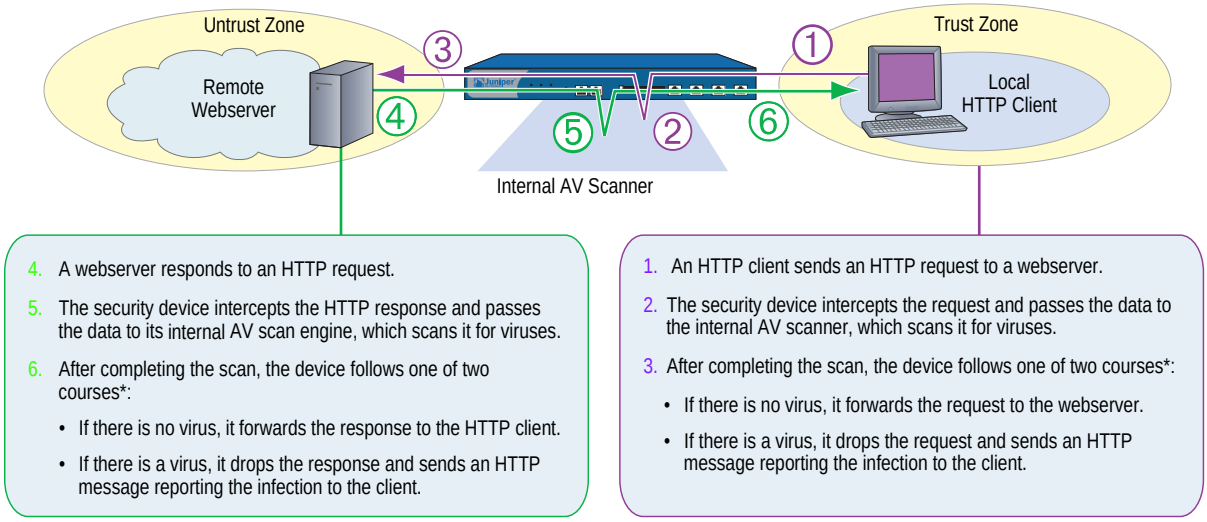
For HTTP traffic, the security device scans both HTTP responses and requests (**get**, **post**, and **put** commands). The internal AV scanner examines HTTP downloads, that is, HTTP data contained in responses from a webserver to HTTP requests from a client. The internal AV scanner also scans uploads, such as when an HTTP client completes a questionnaire on a webserver or when a client writes a message in an email originating on a webserver.

Depending on the results of the scan and how you have configured the fail-mode behavior, the security device takes one of the following actions:

If the Data	And	The Security Device
is uncontaminated		passes the data to the HTTP client
contains a virus		drops the data and sends a virus notification message to the HTTP client
exceeds the maximum content size	**drop** is set	drops the data and sends a “file too large” message to the HTTP client
exceeds the maximum content size	**drop** is unset	passes the data to the HTTP client
cannot successfully be scanned	**fail mode** is unset (drop)	drops the data and sends a “scan error” message to the HTTP client
cannot successfully be scanned	**traffic permit** is set **(fail mode** is permit)	passes the data to the HTTP client

If the Data	And	The Security Device
exceeds the maximum concurrent messages	**drop** is set	drops the data from the data channel and sends an "exceeding maximum message setting" message to the HTTP client through the control channel
exceeds the maximum concurrent messages	**drop** is unset	passes the data to the HTTP client through the data channel

Figure 34: Antivirus Scanning for HTTP Traffic



* If the scanned data exceeds the maximum content setting, or, if the scan cannot be successfully completed, the security device follows a different course of action depending on the fail-mode setting.

HTTP MIME Extensions

By default, HTTP scanning does not scan HTTP entities composed of any of the following Multipurpose Internet Mail Extensions (MIME) content types and subtypes (when present following a slash):

- Application/x-director
- Application/pdf
- Image/
- Video/
- Audio/
- Text/css
- Text/html

To improve performance, Juniper Networks security devices do not scan the above MIME content types. Because most HTTP entities are made up of the above content types, HTTP scanning only applies to a small subset of HTTP entities, such as application/zip and application/exe content types, which are most likely to contain viruses.

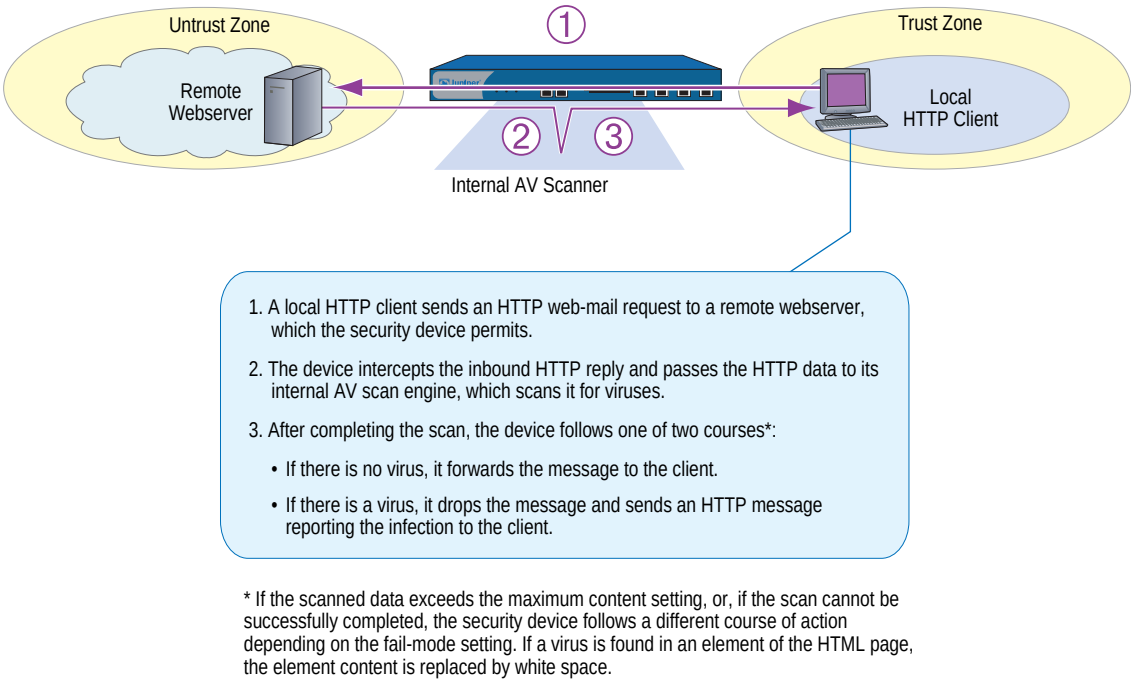
To change HTTP scanning behavior so that the security device scans all HTTP traffic regardless of MIME content types, enter the following command:

```
set av profile jnpr-profile
(av:jnpr-profile)-> unset av http skipmime
(av:jnpr-profile)-> exit
save
```

HTTP Webmail

For HTTP Webmail traffic, the security device redirects the webserver replies (responding to a client's HTTP Webmail requests) to the internal AV scanner before forwarding the traffic to the client.

Figure 35: Antivirus Scanning for HTTP Webmail Traffic



Scanning IMAP and POP3 Traffic

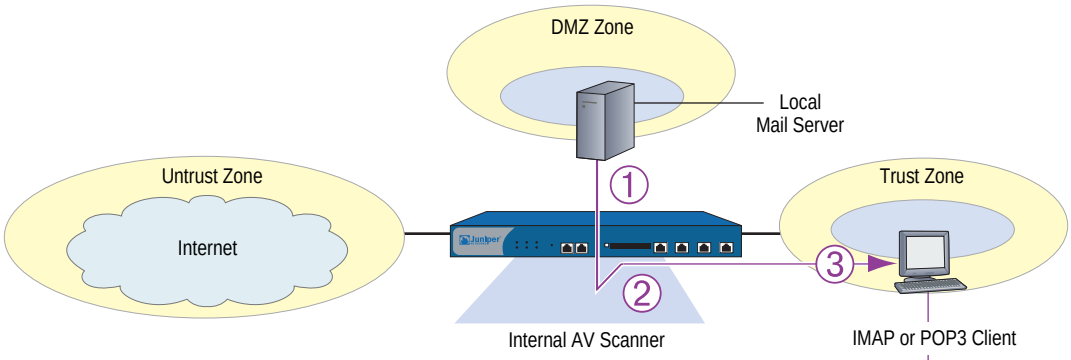
For IMAP and POP3 traffic, the security device redirects traffic from a local mail server to the internal AV scanner before sending it to the local IMAP or POP3 client. Depending on the results of the scan and how you have configured the fail-mode behavior, the security device takes one of the following actions:

If the Data	And	The Security Device
is uncontaminated		passes the message to the IMAP or POP3 client.

contains a virus	**email notification** is set	changes the content type to text/plain, replaces the body of the message with the following notice, sends it to the IMAP or POP3 client, and notifies the sender:
		`VIRUS WARNING.`
		`Contaminated File: filename`
		`Virus Name: virus_name`

exceeds the maximum content size or cannot successfully be scanned or exceeds the maximum concurrent messages	**drop** is set **fail mode** is unset (drop) **email notification** is set	changes the content type to "text/plain," replaces the body of the message with the following notice, and sends it to the IMAP or POP3 client: Content was not scanned for viruses because *reason_text_str* (code *number*), and it was dropped. *reason_text_str* can be one of the following: ■ The file was too large. ■ An error or a constraint was found. ■ The maximum content size was exceeded. ■ The maximum number of messages was exceeded. notifies the sender/recipient of detected virus or scanning errors.
exceeds the maximum content size or cannot successfully be scanned or exceeds the maximum concurrent messages	**drop** is unset **traffic permit** is set (**fail mode** is permit) **drop** is unset **email notification** is set	passes the original message to the IMAP or POP3 client with the original subject line modified as follows: *original_subject_text_str* (No virus check because *reason_text_str*, code *number*) notifies the sender/recipient of detected virus or scanning errors.

Figure 36: Antivirus Scanning for IMAP and POP3 Traffic



1. The IMAP or POP3 client downloads an email message from the local mail server.
2. The security device intercepts the e-mail message and passes the data to the internal AV scanner, which scans it for viruses.
3. After completing the scan, the security device follows one of two courses*:
   - If there is no virus, it forwards the message to the client.
   - If there is a virus, it sends a message reporting the infection to the client.

* If the scanned message exceeds the maximum content setting or, if, the scan cannot be successfully completed, the security device follows a different course of action depending on the fail-mode setting.

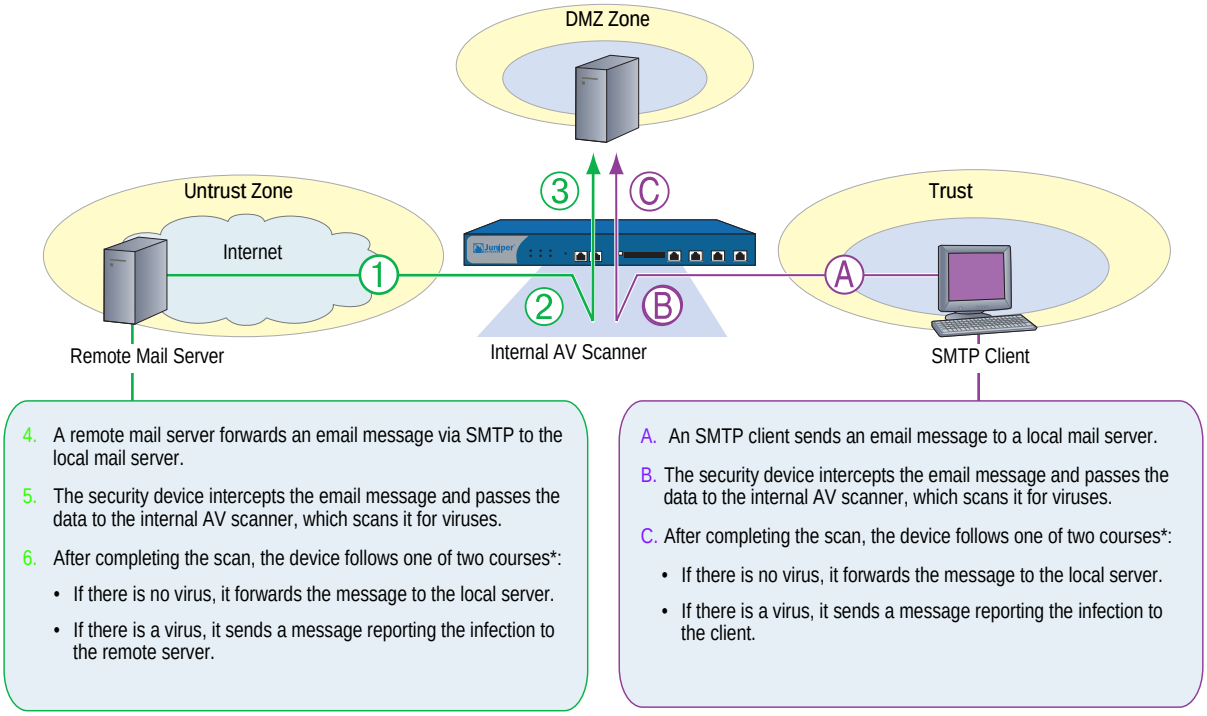
Scanning SMTP Traffic

For SMTP traffic, the security device redirects traffic from local SMTP clients to the internal AV scanner before sending it to the local mail server. Depending on the results of the scan and how you have configured the fail-mode behavior, the security device takes one of the following actions:

If the Data	And	The Security Device
is uncontaminated		passes the message to the SMTP recipient.
contains a virus	**email notification** is set	changes the content type to text/plain, replaces the body of the message with the following notice, sends it to the SMTP recipient, and notifies the sender: `VIRUS WARNING.` `Contaminated File:` *`filename`* `Virus Name:` *`virus_name`*
exceeds the maximum content size or cannot successfully be scanned or exceeds the maximum concurrent messages	**drop** is set **fail mode** is unset (drop) **drop** is set **email notification** is set	changes the content type to text/plain, replaces the body of the message with the following notice, and sends it to the SMTP recipient: Content was not scanned for viruses because *reason_text_str* (code *number*), and it was dropped. *reason_text_str* can be one of the following: ■ The file was too large. ■ An error or constraint was found. ■ The maximum content size was exceeded. ■ The maximum number of messages was exceeded notifies the sender/recipient of detected virus or scanning errors.
exceeds the maximum content level or cannot successfully be scanned or exceeds the maximum concurrent messages	**drop** is disabled **traffic permit** is set (fail mode is permit) **drop** is unset **email notification** is set	passes the original message to the SMTP recipient with the original subject line modified as follows: *original_subject_text_str* (No virus check because *reason_text_str*, code *number*) notifies the sender/recipient of detected virus or scanning errors.

NOTE: Because an SMTP *client* refers to the entity that sends email, a client could, in fact, be another SMTP server.

Figure 37: Antivirus Scanning for SMTP Traffic



* If the scanned message exceeds the maximum content setting, or, if the scan cannot be successfully completed, the security device follows a different course of action depending on the fail-mode setting.

Redirecting Traffic to ICAP AV Scan Servers

Your Juniper Networks security device communicates with an external AV scan engine using the Internet Content Adaptation Protocol (ICAP). ScreenOS 5.4 supports redirection of HTTP and SMTP traffic only.

To configure the security device to support external ICAP AV scanning, perform the following steps:

1. Use the **set icap** command to configure the external ICAP scan server.

2. Configure an ICAP profile and specify one or more of the following:

Command	Description
timeout	Specifies the timeout value for an AV session for a specific protocol (HTTP or SMTP).
http skipmime	Skips the specified files in the MIME list from AV scanning. **Note:** Disabling the skipmime list allows the security device to scan all kinds of HTTP traffic regardless of MIME content types.
email-notify	Notifies sender or recipient of detected virus or scanning errors for SMTP traffic only.

WebUI

Objects > Antivirus > ICAP Server > New: Enter the following, then click **Apply**:

ICAP AV Server Name: **ICAP_Server1**
Enable: (select), Scan Server Name/IP: 1.1.1.1
Scan Server Port: 1344, Scan URL: /SYMCScan-Resp-AV
Probe Interval: 10, Max Connections:

CLI

```
set icap server icap_server1 host 1.1.1.1
save
```

The ICAP server is automatically enabled when it is set up.

Updating the AV Pattern Files for the Embedded Scanner

Internal AV scanning requires that you load a database of AV patterns onto the Juniper Networks security device and periodically update the pattern file.

Before you start updating the AV pattern files, make sure your device supports the following:

Prerequisites	Description
Valid AV license key	■ Juniper-Kaspersky antivirus scanner: av_v2_key ■ Trend Micro antivirus scanner: av_key
Access to the Internet	Your device has a route to the internet
DNS and port settings	Verify your DNS setting and port 80
AV signature service	See "Subscribing to the AV Signature Service" on page 74

Subscribing to the AV Signature Service

To purchase a subscription for the AV signature service you must first register your device. For the life of the subscription, you can load the current version of the database and update it as newer versions become available. The procedure for initiating the AV signature service varies depending on one of the following:

- If you purchased a security device with AV functionality, you can load an AV pattern file for a short period after the initial purchase. You must, however, register the device and purchase a subscription for AV signatures in order to continue receiving pattern updates.
- If you are upgrading a current security device to use internal AV scanning, you must register the device and purchase a subscription for AV signatures before you can begin loading the AV pattern file. After completing the registration process, you must wait up to four hours before initiating the AV pattern file download.

NOTE: For more information about the AV signature service, see "Registration and Activation of Subscription Services" on page **2-**264.

Updating AV Patterns

Figure 38 and Figure 39 illustrate how the pattern file is updated. Update the AV pattern file as follows:

1. On the security device, specify the URL address of the pattern-update server.

   Depending on your AV scan engine type, use one of the following two default pattern-update URLs:

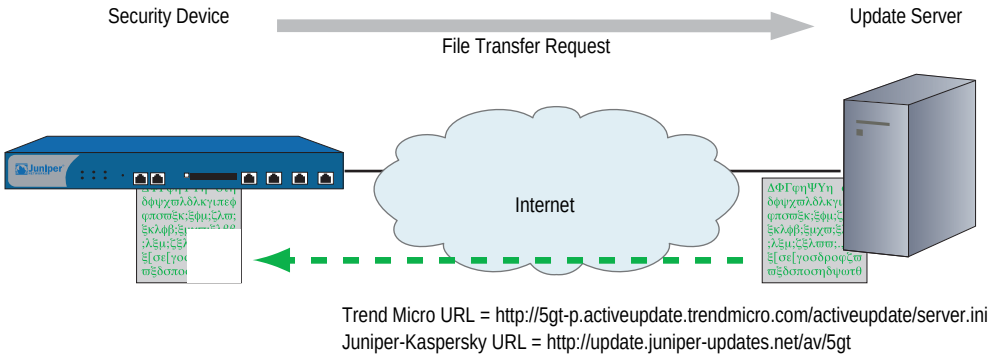
   - Juniper-Kaspersky antivirus scanner

     http://update.juniper-updates.net/av/5gt

   - Trend Micro antivirus scanner

     http://5gt-p.activeupdate.trendmicro.com/activeupdate/server.ini

Figure 38: Updating Pattern Files—Step 1

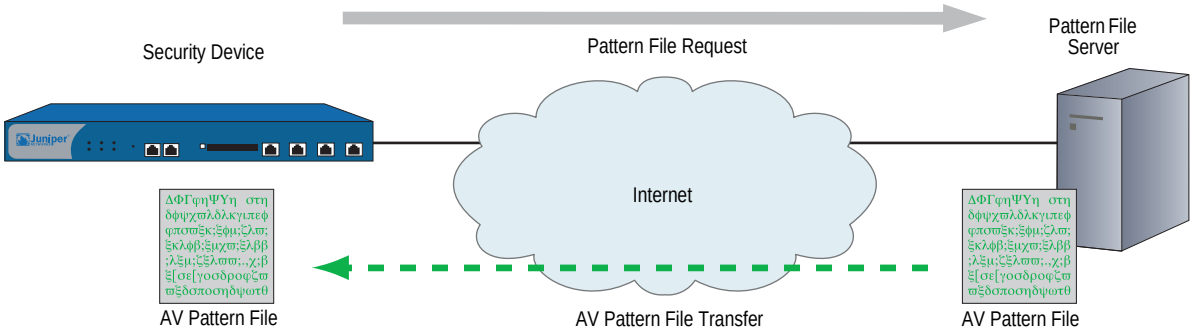


2. After the security device downloads the server-initialization file, the device checks that the pattern file is valid. The device then parses the file to obtain information about it, including the file version, size, and location of the pattern fileserver.

NOTE: ScreenOS contains a CA certificate for authenticating communications with the pattern update fileserver.

3. If the pattern file on the security device is out of date (or nonexistent because this is the first time you are loading it), and, if the AV pattern-update service subscription is still valid, the device automatically retrieves an updated pattern file from the pattern fileserver.

Figure 39: Updating Pattern Files—Step 2



4. The device saves the new pattern file to flash and RAM memory and overwrites the existing file, if there is one.

Updates to the pattern file are added as new viruses propagate. You can configure the security device to regularly update the pattern file automatically, or you can update the file manually.

NOTE: Once your subscription expires, the update server no longer permits you to update the AV pattern file.

Example: Automatic Update

In this example, you configure the security device to update the pattern file automatically every 120 minutes. (The default AV pattern-update interval is 60 minutes.) For example, if the pattern-update server is located at the URL: http://update.juniper-updates.net/av/5gt, you configure automatic update as follows:

WebUI

Screening > Antivirus > Scan Manager: Enter the following, then click **Apply**:

Pattern Update Server: http://update.juniper-updates.net/av/5gt
Auto Pattern Update: (select), Interval: 120 minutes (10~10080)

CLI

```
set av scan-mgr pattern-update-url http://update.juniper-updates.net/av/5gt
    interval 120
save
```

Example: Manual Update

In this example, you update the pattern file manually. The pattern update server is located at the following URL: http://update.juniper-updates.net/av/5gt

WebUI

Screening > Antivirus > Scan Manager: Enter the following, then click **Apply**:

Pattern Update Server: http://update.juniper-updates.net/av/5gt
Update Now: (select)

CLI

exec av scan-mgr pattern-update

The **set** command is not required because the URL address is the default.

AV Scanner Global Settings

You can modify AV scanner settings to serve the needs of your network environment. The following sections explain the global settings for your AV scanner:

AV Resource Allotment

A malicious user might generate a large amount of traffic all at once in an attempt to consume all available resources and hinder the ability of the AV scanner to scan other traffic. To prevent such activity from succeeding, the Juniper Networks security device can impose a maximum percentage of AV resources that traffic from a single source can consume at one time. The default maximum percentage is 70 percent. You can change this setting to any value between 1 and 100 percent, where 100 percent does not impose any restriction on the amount of AV resources that traffic from a single source can consume.

WebUI

NOTE: You must use the CLI to configure this option.

CLI

set av all resources *number*
unset av all resources

The above **unset av** command returns the maximum percentage of AV resources per source to the default (70 percent).

Fail-Mode Behavior

Fail-mode is the behavior that the security device applies when it cannot complete a scan operation—either to permit the unexamined traffic or to block it. By default, if a device cannot complete a scan, it blocks the traffic that a policy with antivirus checking enabled permits. You can change the default behavior from block to permit.

When the AV scan engine is scanning a file and runs out of memory (typically, when decompressing files), the content is either dropped or passed based on the max-content-size (set av scan-mgr max-content-size) setting, instead of the fail-mode setting.

WebUI

Screening > Antivirus > Global: Select **Fail Mode Traffic Permit** to permit unexamined traffic, or clear it to block unexamined traffic, then click **Apply**.

CLI

```
set av all fail-mode traffic permit
unset av all fail-mode traffic
```

The above **unset av** command returns the fail mode behavior to the default (block unexamined traffic).

Maximum Content Size and Maximum Messages (Internal AV Only)

Scan manager settings for maximum content size and maximum messages are supported on internal AV only. ICAP AV does not support maximum content size and maximum messages settings.

The internal AV scanner in some devices examines a maximum of 16 messages and 10 megabytes of decompressed file content at a time. If the total number of messages or the size of the content received concurrently exceeds these limits, by default the scanner drops the content without checking for viruses.

NOTE: The default for Maximum Content Size is 10 MB for some security devices. However, if DI is enabled, we recommend that you configure a value of 6 MB.

For example, the scanner can receive and examine four 4-megabyte messages concurrently. If the scanner receives nine 2-megabyte messages concurrently, it drops the contents of the last two files without scanning it. You can change this default behavior so that the scanner passes the traffic instead of dropping it by doing the following:

WebUI

Screening > Antivirus > Scan Manager: Select **pass** if the file size exceeds 10,000 KB

Or

Select **pass** if the number of files exceeds 16, then click **Apply**.

CLI

```
unset av scan-mgr max-content-size drop
unset av scan-mgr max-msgs drop
```

When the AV scan engine is scanning a file and runs out of memory (typically, when decompressing files), the content is either dropped or passed based on the max-content-size setting, instead of the fail-mode (set av all failmode) setting.

HTTP Keep-Alive

By default, the security device uses the HTTP "keep-alive" connection option, which does not send a TCP FIN to indicate the termination of data transmission. The HTTP server must indicate that it has sent all the data in another way, such as by sending the content length in the HTTP header or by some form of encoding. (The method that a server uses varies by server type.) This method keeps the TCP connection open while the antivirus examination occurs, which decreases latency and improves processor performance.

You can change the default behavior of the security device to use the HTTP "close" connection option for indicating the end of data transmission. (If necessary, the device changes the token in the connection header field from "keep-alive" to "close.") With this method, when the HTTP server completes its data transmission, it sends a TCP FIN to close the TCP connection and indicate that the server has finished sending data. When the device receives a TCP FIN, it has all the HTTP data from the server and can instruct the AV scanner to begin scanning it.

NOTE: The "keep-alive" not as secure as the "close" connection method. You can change the default behavior if you find that HTTP connections are timing out during the antivirus examination.

WebUI

Screening > Antivirus > Global: Select **Keep Alive** to use the "keep-alive" connection option, or clear it to use the "close" connection option, then click **Apply**.

CLI

```
set av http keep-alive
unset av http keep-alive
```

HTTP Trickling (Internal AV Only)

HTTP trickling is the forwarding of specified amounts of unscanned HTTP traffic to the requesting HTTP client to prevent the browser window from timing out while the scan manager examines downloaded HTTP files. (The security device forwards small amounts of data in advance of transferring an entire scanned file.) By default, HTTP trickling is disabled. To enable it and use the default HTTP trickling parameters, do either of the following:

NOTE: HTTP trickling is supported on internal AV only. ICAP AV does not support HTTP trickling.

WebUI

Screening > Antivirus > Global: Select the Trickling Default checkbox, then click **Apply**.

CLI

```
set av http trickling default
```

With the default parameters, the security device employs trickling if the size of an HTTP file is 3 MB or larger. Then it forwards 500 bytes of content for every 1 MB sent for scanning.

To change the parameters for HTTP trickling, do either of the following:

WebUI

Screening > Antivirus > Global: Enter the following, then click **Apply**:

Trickling:
Custom: (select)
Minimum Length to Start Trickling: Enter ***number1***.
Trickle Size: Enter ***number2***.
Trickle for Every MB Sent for Scanning: Enter ***number3***.

CLI

set av http trickling *number1 number3 number2*

The three *number* variables have the following meanings:

- *number1:* The minimum size (in megabytes) of an HTTP file required to trigger trickling
- *number2:* The size (in bytes) of unscanned traffic that the security device forwards
- *number3*: The size (in megabytes) of a block of traffic to which the security device applies trickling

---

NOTE: Data trickled to the client's hard drive appears as a small, unusable file. Because trickling works by forwarding a small amount of data to a client without scanning it, virus code might be among the data that the security device has trickled to the client. We advise users to delete such files.

You can disable HTTP trickling in the WebUI (Screening > Antivirus: Click **Disable** in the Trickling section) or with the CLI command **unset av http trickling enable**. However, if a file being downloaded is larger than 8 MB and HTTP trickling is disabled, the browser window will probably time out.

AV Scanner Profile Settings

Policies use AV profiles to determine which traffic undergoes AV examination and the actions to take as a result of this examination.

NOTE: For internal embedded AV only, a predefined AV profile, **ns-profile,** exists on your Juniper Networks security device.

You must do the following to link the AV profile to a firewall policy. Only one AV profile can be linked to a firewall policy.

WebUI

Policies: Click **Edit** on the policy to which you want to link the AV profile and select the profile under Antivirus Profile. Click **OK**.

CLI

device-> **set policy id** *policy_num* **av ns-profile**

The following sections explain how to initiate an AV profile and configure the profile settings:

- "Initiating an AV Profile for Internal AV" on page 81
- "Example: (Internal AV) Scanning for All Traffic Types" on page 82
- "Example: AV Scanning for SMTP and HTTP Traffic Only" on page 82
- "AV Profile Settings" on page 84

Initiating an AV Profile for Internal AV

The following commands initiate a custom AV profile named *jnpr-profile,* which by default is configured to scan FTP, HTTP, IMAP, POP3, and SMTP traffic.

WebUI

Screening > Antivirus > Profile: Select **New** and enter the profile name, *jnpr-profile,* then click **OK**.

CLI

set av profile *jnpr-profile*
device(av:jnpr-profile)->

device-> **set av profile jnpr-profile**
device(av:jnpr-profile)->

After you enter an AV profile context, all subsequent command executions modify the specified AV profile (*jnpr-profile*).

Example: (Internal AV) Scanning for All Traffic Types

In this example, you configure the AV scanner to examine FTP, HTTP, IMAP, POP3, and SMTP traffic. Because you anticipate that the scanner will be processing a lot of traffic, you also increase the timeout from 180 seconds (the default setting) to 300 seconds.

WebUI

Screening > Antivirus > Profile: Enter *profile_name*, then click **OK**.

By default, traffic for all five protocols is scanned.

NOTE: To change the timeout value, you must use the CLI.

CLI

```
set av profile jnpr-profile
(av:jnpr-profile)-> set http enable
(av:jnpr-profile)-> set http timeout 300
(av:jnpr-profile)-> set ftp enable
(av:jnpr-profile)-> set ftp timeout 300
(av:jnpr-profile)-> set imap enable
(av:jnpr-profile)-> set imap timeout 300
(av:jnpr-profile)-> set pop3 enable
(av:jnpr-profile)-> set pop3 timeout 300
(av:jnpr-profile)-> set smtp enable
(av:jnpr-profile)-> set smtp timeout 300
(av:jnpr-profile)-> exit
save
```

Example: AV Scanning for SMTP and HTTP Traffic Only

By default, the AV scanner examines FTP, HTTP, IMAP, POP3, and SMTP traffic. You can change the default behavior so that the scanner examines specific types of network traffic only.

You can also change the timeout value for each protocol. By default, an AV scan operation times out after 180 seconds if the security device does not start scanning after it receives all the data. The range is 1 to 1800 seconds.

In this example, you configure the AV scanner to examine all SMTP and HTTP traffic. You return the timeout value for both protocols to their defaults: 180 seconds.

NOTE: The internal AV scanner examines specific HTTP Webmail patterns only. The patterns for Yahoo!, Hotmail, and AOL mail services are predefined.

WebUI

Screening > Antivirus > Select **New** and enter the profile name *jnpr-profile*.

Enter the following, then click **OK**.

Protocols to be scanned:
HTTP: (select)
SMTP: (select)
POP3: (clear)
FTP: (clear)
IMAP: (clear)

NOTE: To change the timeout value, you must use the CLI.

CLI

```
set av profile jnpr-profile
(av:jnpr-profile)-> set smtp timeout 180
(av:jnpr-profile)-> set http timeout 180
(av:jnpr-profile)-> unset pop3 enable
(av:jnpr-profile)-> unset ftp enable
(av:jnpr-profile)-> unset imap enable
(av:jnpr-profile)-> exit
unset av http webmail enable
save
```

The unset webmail command enables a full HTTP scan including webmail. Make sure a policy enabling HTTP exists.

AV Profile Settings

The following scanning options are configured for each application protocol:

Decompressing File Attachments

When the device receives content, the internal AV scanner decompresses any compressed files. It decompresses up to two layers of compressed files by default. For example, if the scanner receives a file with an attachment and the attachment is a compressed file layered within another compressed file, the scanner may decompress both layers in order to detect any viruses. You can configure the internal AV scanner to decompress up to four compressed files layered within one another.

WebUI

Screening > Antivirus > Profile: Select **New** or **Edit** to edit an existing profile. Update the Decompress Layer to 2, then click **Apply**.

CLI

```
set av profile jnpr-profile
(av:jnpr-profile)-> set smtp decompress-layer 3
```

When transmitting data, some protocols use content encoding. The AV scan engine needs to decode this layer, which is considered as a decompression level before it scans for viruses.

AV Scanning Based on File Extensions

File-extension lists are used to determine on which files undergo AV scanning for a specific protocol. You can select to *Include* a file-extension list and *Exclude* a file-extension list for each protocol.

A message is scanned when the file extension of a message is in the inclusion file-extension list. A message is not scanned if the file extension is in the exclusion file-extension list. If the file extension is not in either file-extension list, then the scanning decision depends on the default file-extension-scan setting. The default file extension is in the scan engine database, so it is read-only. There is no predefined file extension list for each protocol.

Configure the AV scanner to scan IMAP traffic by extensions and exclude files with the following extensions: .ace, .arj, and .chm.

WebUI

Screening > Antivirus > Ext-list > New > Enter an extension-list name (elist1), and enter the list of extensions (ace;arj;chm). Click **OK**.

Antivirus > Profile > Select the Profile to **Edit** > Select **IMAP** > Select the following options, then click **OK**:

Enable
Scan Mode: Scan by Extension
Exclude Extension List: elist1

CLI

```
set av extension-list elist1 ace;arj;chm
set av profile test1
(av:test1)-> set imap scan-mode scan-ext
(av:test1)-> set imap extension-list exclude elist1
```

AV Scanning Based on HTTP Content Type

You may use this option to determine which HTTP traffic must undergo AV scanning. The HTTP traffic is categorized into default predefined Multipurpose Internet Mail Extensions (MIME) types such as application/x-director, application/pdf, image, and so on. You can configure the AV profile to skip MIME lists containing specific MIME types. The default predefined MIME list is ns-skip-mime-list.

In this example, you configure the security device to scan all kinds of HTTP traffic regardless of MIME content type:

WebUI

Screening > Antivirus > Profile > Select the Profile to **Edit** > Select HTTP and clear the Skipmime Enable option. Click **OK**.

CLI

```
set av profile jnpr-profile
(av:jnpr-profile)-> unset av http skipmime
(av:jnpr-profile)-> exit
save
```

For more information about MIME types, refer to the *ScreenOS CLI Reference Guide IPv4 Command Descriptions*.

Notifying Sender and Recipient via Email

The email-notification option applies only to the IMAP, POP3, and SMTP protocols. You can configure the AV profile to notify senders or recipients scanning errors or virus information.

When a virus is found in an email message, the content of the warning message (virus name, source/destination IP) is included in a notification-level message. The warning-level message is sent via an email through the SMTP protocol.

When a scanning error occurs in a message, the content of the scanning error message should be included in a warning-level message. This message is sent via an email through the SMTP protocol.

In this example, you configure the security device to do the following:

- Notify the sender when a virus is detected
- Notify the sender and recipients if scanning errors occur

WebUI

Screening > Antivirus > Profile > Select the Profile to **Edit** > Select IMAP, then click **OK**.

Enter the following, then click **OK:**
Protocols to be scanned:
Email Notify > Select Virus Sender
Email Notify > Select Scan-error Sender
Email Notify > Select Scan-error Recipient

CLI

```
set av profile jnpr-profile
(av:jnpr-profile)-> set imap email-notify virus sender
(av:jnpr-profile)-> set imap email-notify scan-error sender
(av:jnpr-profile)-> set imap email-notify scan-error recipient
(av:jnpr-profile)-> exit
save
```

Example: Dropping Large Files

In this example, you configure the AV scanner to decompress HTTP traffic of up to three files layered within one another. You also configure the scanner to drop content either if the total number of messages received concurrently exceeds four messages or if the total decompressed size of the content exceeds the configured value. The total decompressed file content size that ScreenOS can handle is device-specific with a minimum of 10MB.

NOTE: The default value for decompressed file content size is per message and not the total number of concurrent messages being examined.

The default values for Maximum Concurrent Messages and Maximum Queue size indicate that the AV scanner can examine a total of 256 concurrent messages at any specific time. The 257th message is dropped or passed as configured.

WebUI

Screening > Antivirus > Scan Manager: Enter the following, then click **OK**:

Drop: (select) file if it exceeds 3000 KB (20~10000)
Drop: (select) file if the number of files exceeds 4 files (1~16)

Screening > Antivirus > Profile: Select Edit > HTTP: Enter the following, then click **OK**:

File decompression: 3 layers (1~4)

CLI

```
set av scan-mgr max-msgs 4
set av scan-mgr max-content-size 3000
set av scan-mgr max-content-size drop
set av profile jnpr-profile
(av:jnpr-profile)-> set http decompress-layer 3
(av:jnpr-profile)-> exit
save
```

Anti-Spam Filtering

Spam consists of unwanted email messages, usually sent by commercial, malicious, or fraudulent entities. The anti-spam feature examines transmitted messages to identify spam. When the device detects a message deemed to be spam, it either drops the message or tags the message field with a preprogrammed string.

This anti-spam feature is not meant to replace your anti-spam server, but to complement it. Configuring this command prevents an internal corporate email server from receiving and distributing spams. Corporate users retrieve emails from an internal email server without going through the firewall. This should be a typical configuration in an enterprise environment.

Juniper Networks anti-spam uses a constantly updated, IP-based, spam-blocking service that uses information gathered worldwide. Because this service is robust and yields very few false positives, you are not required to tune or configure black lists. However, you have the option of adding specific domains and IPs to local white lists or black lists, which the device can enforce locally.

NOTE: This release supports anti-spam for the SMTP protocol only.

To prevent or reduce the volume of spam messages you receive, you can configure an anti-spam profile. You can use the profile in policies to detect and filter out suspected spam messages. An anti-spam profile allows you to designate lists of IP addresses, emails, hostnames, or domain names identified as malicious (spam) or benign (non-spam). The anti-spam profile can include lists of the following types:

- Public-based black or white lists

  If the connection is from a mail- forwarding agent, the device can filter the connection's source IP address using lists of devices deemed to be benign (white list) or malicious (black list).

- Custom defined black or white lists

  - Domain name-based white or black lists.

    The device can use such lists to filter connections that use domain names deemed to be benign or malicious.

  - Address book-based white or black lists.

The device can use such lists to base filtering on the sender's email address or domain. By default, any email server should accept its own user's email.

Black Lists and White Lists

The anti-spam feature requires that the firewall have Internet connectivity with the Spam Block List (SBL) server. Domain Name Service (DNS) must be available to access the SBL server. The firewall performs reverse DNS lookups on the source of the SMTP sender (or relaying agent), adding the name of the SBL server (such as sbl-server) as the authoritative domain. The DNS server then forwards each request to the SBL server, which returns a value to the firewall.

Alternatively, you can configure local white and black lists. In this case, by default the system checks first against the local database of white/black lists. If it does not find the hostname, the firewall proceeds to query the SBL server located on the Internet.

Basic Configuration

The following commands provide an example of basic anti-spam configuration where you are protecting an smtp server (or relay server) from receiving spam emails.

```
set anti-spam profile ns-profile
set policy from untrust to trust any mail-server SMTP permit log anti-spam
    ns-profile
```

In the following example, the firewall tests spammer.org to see if it resides on the white list or the black list.

```
exec anti-spam testscan spammer.org
```

If the black list contains spammer.org, the device might produce the following output:

```
AS: anti spam result: action Tag email subject, reason: Match local blacklist
```

Alternatively, if the white list contains spammer.org, the device may produce the following output:

```
AS: anti spam result: action Pass, reason: Match local whitelist
```

For information about creating black lists or white lists, see "Defining a Black List" on page 89 and "Defining a White List" on page 89.

Filtering Spam Traffic

In the following examples, SMTP traffic that includes spam traverses the security device. However, ScreenOS checks for spam by either DNS name or IP address.

The following commands provide an example of filtering spam traffic.

```
device-> exec anti-spam test 2.2.2.2
AS: anti spam result: action Tag email subject, reason: Match local black list
exec anti-spam testscan spammer.org
AS: anti spam result: action Tag email subject, reason: Match local black list
```

Dropping Spam Messages

Executing the **set anti-spam profile ns-profile** command without specifying further options places the CLI within the context of a new or existing anti-spam profile. For example, the following commands define a profile named **ns-profile** and then enter the **ns-profile** context to instruct the device to drop suspected spam messages:

```
device-> set anti-spam profile ns-profile
device(ns-profile)-> set default action drop
```

After you enter an anti-spam context, all subsequent command executions modify the specified anti-spam profile (**ns-profile** in this example). To save your changes, you must first exit the anti-spam context and then enter the **save** command:

```
device(ns-profile)-> exit
device-> save
```

Defining a Black List

Use the black-list commands to add or remove an IP or email address, a hostname, or a domain name from the local anti-spam black list. Each entry in a black list can identify a possible spammer.

To define a Black List, perform the following steps:

1. Initiate a profile context (**ns-profile**).
2. Give the profile a black list entry that prevents connections with the hostname www.wibwaller.com.
3. Exit the spam context and apply the profile to an existing policy (id 2).

```
device-> set anti-spam profile ns-profile
device(anti-spam:ns-profile)-> set blacklist www.wibwaller.com
device(anti-spam:ns-profile)-> exit
device-> set policy id 2 anti-spam ns-profile
```

Defining a White List

Use the white-list commands to add or remove an IP or email address, a hostname, or a domain name from the local white list. Each entry in a white list can identify an entity that is not a suspected spammer. The following table shows some possible entries.

To define a White List, perform the following steps:

1. Initiate a profile context (**ns-profile**).
2. Give the profile a white list entry that allows connections with the hostname www.fiddwicket.com.
3. Exit the spam context and apply the profile to an existing policy (id 2).

```
device-> set anti-spam profile ns-profile
device(anti-spam:ns-profile)-> set whitelist www.fiddwicket.com
```

```
device(anti-spam:ns-profile)-> exit
device-> set policy id 2 anti-spam ns-profile
```

Defining a Default Action

Use the default commands to specify how the device handles messages deemed to be spam. The device can either drop a spam message or identify it as spam by tagging it.

You can place the tag either in the message header or the subject line.

To define the default action for spam, perform the following tasks:

1. Initiate a profile context (**ns-profile**).
2. Specify that email messages deemed to be spam will have the string "This is spam" added to the message header.
3. Exit the spam context and apply the profile to an existing policy (id 2).

```
device-> set anti-spam profile ns-profile
device(anti-spam:ns-profile)-> set default action tag header "This is spam"
device(anti-spam:ns-profile)-> exit
device-> set policy id 2 anti-spam ns-profile
```

Enabling a Spam-Blocking List Server

Use the **sbl** command to enable use of the external spam-blocking SBL service, which uses a black list to identify known spam sources. The service replies to queries from the device about whether an IP address belongs to a known spammer.

Example: These commands perform the following tasks:

1. Initiate a profile context (**ns-profile**).
2. Enable use of the default anti-spam service.
3. Exit the spam context and apply the profile to an existing policy (id 2).

```
device-> set anti-spam profile ns-profile
device(anti-spam:ns-profile)-> set sbl default-server-enable
device(anti-spam:ns-profile)-> exit
device-> set policy id 2 anti-spam ns-profile
```

Web Filtering

Web filtering enables you to manage Internet access by preventing access to inappropriate web content. ScreenOS provides two web-filtering solutions:

- Integrated

  Select security devices support an integrated web-filtering solution that employs Content Portal Authority (CPA) servers from SurfControl.

NOTE: Integrated web-filtering requires you to install a license key on your security device.

  Integrated web filtering allows you to permit or block access to a requested site by binding a web-filtering profile to a firewall policy. A web-filtering profile specifies URL categories and the action the security device takes (permit or block) when it receives a request to access a URL in each category. URL categories are predefined and maintained by SurfControl or are user-defined. For information about configuring the integrated web-filtering feature, see "Integrated Web Filtering" on page 92.

- Redirect

  Select security devices support a web-filtering solution that employs SurfControl and Websense services to a SurfControl or Websense server.

  In redirect web filtering, the security device sends the HTTP request in a TCP connection to either a Websense server or a SurfControl server, enabling you to block or permit access to different sites based on their URLs, domain names, and IP addresses. For information about configuring the redirect web-filtering feature, see "Integrated Web Filtering" on page 92.

NOTE: Use integrated web filtering to manage HTTPS traffic. Redirect web filtering does not support HTTPS traffic.

Using the CLI to Initiate Web-Filtering Modes

You can use the WebUI or CLI to configure your security device for web filtering. If you are using the CLI, then you perform the following steps to configure either of the web-filtering solutions:

1. Select the protocol.

   For example, the **set url protocol type { sc-cpa | scfp | websense }** command selects the protocol.

2. Initiate the web-filtering mode.

   Executing the **set url protocol { sc-cpa | scfp | websense }** command places the CLI in the web-filtering context. Once you initiate the web-filtering context, all subsequent command executions apply to that web-filtering mode.

Table 3 shows the commands for entering and exiting the three different web-filtering modes.

Table 3: Entering and Exiting the Web-Filtering Modes

	Integrated Web Filtering	Redirecting to SurfControl Server	Redirecting to Websense Server
1. Select the protocol	**set url protocol type sc-cpa**	**set url protocol type scfp**	**set url protocol type websense**
2. Initiate the web-filtering context	**set url protocol sc-cpa** **(url:sc-cpa)-> :**	**set url protocol scfp** **(url:scfp)-> :**	**set url protocol websense** **(url:websense)-> :**
3. Exit the web-filtering mode	**(url:sc-cpa)-> :exit**	**(url:scfp)-> :exit**	**(url:websense)-> :exit**

Integrated Web Filtering

To enable web filtering, you first bind a web-filtering profile to a firewall policy. With integrated web filtering, the Juniper Networks security device intercepts each HTTP request, determines whether to permit or block access to a requested site by categorizing its URL, then matches the URL category to a web-filtering profile. A web-filtering profile defines the action the security device takes (permit or block) when it receives a request to access a URL.

A URL category is a list of URLs organized by content. Security devices use the SurfControl predefined URL categories to determine the category of the requested URL. SurfControl Content Portal Authority (CPA) servers maintain the largest database of all types of web content classified into about 40 categories. A partial list of the URL categories is shown in “Define URL Categories (Optional)” on page 95.

For a complete list of SurfControl URL categories, visit the SurfControl website at http://www.surfcontrol.com. In addition to the SurfControl predefined URL categories, you can also group URLs and create categories based on your needs. For information about creating user-defined categories, see “Define URL Categories (Optional)” on page 95.

Following is the basic sequence of events when a host in the Trust zone tries an HTTP connection to a server in the Untrust zone:

1. The security device checks for a firewall policy that applies to the traffic:
   - If there is no firewall policy for the traffic, the device drops the traffic.
   - If there is a firewall policy and if web filtering is enabled on that policy, the device intercepts all HTTP requests.
2. The device checks for a user-defined profile bound to the firewall policy. If there is none, the device then uses the default profile, **ns-profile**.
3. The device determines if the category of the requested URL is already cached. If it is not, the device sends the URL to the SurfControl CPA server for categorization and caches the result.

4. Once the device determines the category of the URL, it checks for the category in the web-filtering profile bound to the firewall policy.

   - If the category is in the profile, the device blocks or permits access to the URL as defined in the profile.
   - If the category is not in the profile, the device performs the configured default action.

This section addresses the following integrated web-filtering topics:

- "SurfControl Servers" on this page
- "Redirect Web Filtering" on page 101
- "Web-Filtering Cache" on page 93
- "Configuring Integrated Web Filtering" on page 94
- "Example: Integrated Web Filtering" on page 99

SurfControl Servers

SurfControl has three server locations, each of which serves a specific geographic area: the Americas, Asia Pacific, and Europe/MiddleEast/Africa. The default primary server is the Americas, and the default backup server is Asia Pacific. You can change the primary server, and the security device automatically selects a backup server, based on the primary server. (The Asia Pacific server is the backup for the Americas server, and the Americas server is the backup for the other two servers.)

The SurfControl CPA server periodically updates its list of categories. Since the CPA server does not notify its clients when the list is updated, the security device must periodically poll the CPA server. By default, the device queries the CPA server for category updates every two weeks. You can change this default to support your networking environment. You can also manually update the category list by entering the web-filtering context and executing the **exec cate-list-update** command. To manually update the category list, do the following:

```
device-> set url protocol sc-cpa
device(url:sc-cpa)-> exec cate-list-update
```

Web-Filtering Cache

By default, the security device caches the URL categories. This action reduces the overhead of accessing the SurfControl CPA server each time the device receives a new request for previously requested URLs. You can configure the size and duration of the cache, according to the performance and memory requirements of your networking environment. The default cache size is platform-dependent, and the default timeout is 24 hours.

In the following example, you change the cache size to 500 kilobytes (KB) and the timeout value to 18 hours.

WebUI

Screening > Web Filtering > Protocol Selection > SC-CPA: Enter the following, then click **Apply**:

Enable Cache: (select)
Cache Size: 500 (K)
Cache Timeout: 18 (Hours)

CLI

```
device-> set url protocol sc-cpa
device(url:sc-cpa)-> set cache size 500
device(url:sc-cpa)-> set cache timeout 18
device(url:sc-cpa)-> exit
device-> save
```

Configuring Integrated Web Filtering

To configure a security device for web filtering, perform the following steps:

1. "Set Up a Domain Name Server" on this page
2. "Enable Web Filtering" on this page
3. "Define URL Categories (Optional)" on page 95
4. "Define Web-Filtering Profiles (Optional)" on page 96
5. "Enable Web-Filtering Profile and Policy" on page 98

Each step is described in detail in the following sections.

1. Set Up a Domain Name Server

The Juniper Networks security device incorporates Domain Name System (DNS) support, allowing you to use domain names as well as IP addresses for identifying locations. You must configure at least one DNS server to enable the security device to resolve the CPA server name to an address. For more information about DNS, see "Domain Name System Support" on page 2-229.

2. Enable Web Filtering

You can use the Web UI or CLI commands to enable integrated web filtering on a security device. If you use the CLI, you must enter the web-filtering context before entering the commands specific to integrated web filtering.

WebUI

Screening > Web Filtering > Protocol Selection: Select **Integrated (SurfControl)**, then click **Apply**. Then select **Enable Web Filtering via CPA Server**, and click **Apply** again.

CLI

```
device-> set url protocol type sc-cpa
device-> set url protocol sc-cpa
device(url:sc-cpa)-> set enable
device(url:sc-cpa)-> exit
device-> save
```

The *device*(url:sc-cpa)-> prompt indicates that you have entered the integrated web-filtering context and can now configure integrated web-filtering parameters.

3. Define URL Categories (Optional)

A category is a list of URLs grouped by content. There are two types of categories: predefined and user-defined. SurfControl maintains about 40 predefined categories. A partial list of the URL categories is shown in Table 4 on page 95. For a complete list and description of each URL category developed by SurfControl, visit the SurfControl website at http://www.surfcontrol.com.

To view the list of SurfControl predefined URL categories, do the following:

WebUI

Screening > Web Filtering > Profile > Predefine Category

CLI

```
device-> set url protocol type sc-cpa
device-> set url protocol sc-cpa
device(url:sc-cpa)-> get category pre
```

The URL category list displayed is similar to that shown in Table 4.

Table 4: Partial List of SurfControl URL Categories

Type	Code	Category Name
Predefine	76	Advertisements
Predefine	50	Arts & Entertainment
Predefine	3001	Chat
Predefine	75	Computing & Internet

The predefined categories list displays the categories and their SurfControl internal codes. Though you cannot list the URLs within a category, you can determine the category of a website by using the Test A Site feature on the SurfControl website at www.surfcontrol.com.

In addition to the SurfControl predefined URL categories, you can group URLs and create categories specific to your needs. Each category can have a maximum of 20 URLs. When you create a category, you can add either the URL or the IP address of a site. When you add a URL to a user-defined category, the device performs DNS lookup, resolves the host name into IP addresses, and caches this information. When a user tries to access a site with the IP address of the site, the device checks the cached list of IP addresses and tries to resolve the hostname.

Many sites have dynamic IP addresses, meaning that their IP addresses change periodically. A user attempting to access a site can type an IP address that is not in the cached list on the device. Therefore, if you know the IP addresses of sites you are adding to a category, enter both the URL and the IP address(es) of the site.

NOTE: If a URL appears in both a user-defined category and a predefined category, the device matches the URL to the user-defined category.

In the following example, you create a category named **Competitors** and add the following URLs: **www.games1.com** and **www.games2.com**

WebUI

Screening > Web Filtering > Profile > Custom List > New: Enter the following, then click **Apply**:

Category Name: Competitors
URL: www.games1.com

Enter the following, then click **OK**:

URL: www.games2.com

CLI

```
device-> set url protocol sc-cpa
device(url:sc-cpa)-> set category competitors url www.games1.com
device(url:sc-cpa)-> set category competitors url www.games2.com
device(url:sc-cpa)-> exit
device-> save
```

4. Define Web-Filtering Profiles (Optional)

A web-filtering profile consists of a group of URL categories assigned with one of the following actions:

- Permit - The security device always allows access to the websites in this category.
- Block - The security device blocks access to the websites in this category. When the device blocks access to this category of websites, it displays a message in your browser indicating the URL category.
- Black List - The security device always blocks access to the websites in this list. You can create a user-defined category or use a predefined category.
- White List - The security device always allows access to the websites in this list. You can create a user-defined category or use a predefined category.

Juniper Networks security devices provide a default profile called **ns-profile**. This profile lists the SurfControl predefined URL categories and their actions. You cannot edit the default profile. To view the predefined profile, use the following command:

WebUI

Screening > Web Filtering > Profile > Predefined Profile

CLI

```
device-> set url protocol sc-cpa
device(url:sc-cpa)-> get profile ns-profile
```

The security device displays the predefined profile as illustrated below:

```
Profile Name: ns-profile
Black-List category: None
White-List category: None
Default Action: Permit

Category                Action
 Advertisements         block
 Arts & Entertainment   permit
 Chat                   permit
 Computing & Internet   permit
 .
 .
 .
 Violence               block
 Weapons                block
 Web-based Email        permit
 other                  permit
```

If the URL in an HTTP request is not in any of the categories listed in the default profile, the default action of the security device is to permit access to the site.

You can create a custom profile by cloning an existing profile, saving it with a new name, and then editing the profile. Perform the following step in the WebUI to clone **ns-profile**.

WebUI

Screening > Web Filtering > Profile > Custom Profile: ns-profile: Select **Clone**.

NOTE: You must use the WebUI to clone **ns-profile**.

You can also create your own web-filtering profile. When you create a web-filtering profile, you can:

- Add both user-defined and SurfControl predefined URL categories
- Specify a category for the black list and/or the white list
- Change the default action

In the following example, you create a custom profile called **my-profile** with a default action of **permit**. Then, you take the category you created in the previous example and add it to my-profile with an action of **block**.

WebUI

Screening > Web Filtering > Profile > Custom Profile > New: Enter the following, then click **Apply**:

Profile Name: my-profile
Default Action: Permit
Select the following, then click **OK**:
Subscribers Identified by:
Category Name: Competitors (select)
Action: Block (select)
Configure: Add (select)

NOTE: To configure the default action using the CLI, specify the action for the Other category.

CLI

```
device-> set url protocol type sc-cpa
device-> set url protocol sc-cpa
device(url:sc-cpa)-> set profile my-profile other permit
device(url:sc-cpa)-> set profile my-profile competitors block
device(url:sc-cpa)-> exit
device-> save
```

5. Enable Web-Filtering Profile and Policy

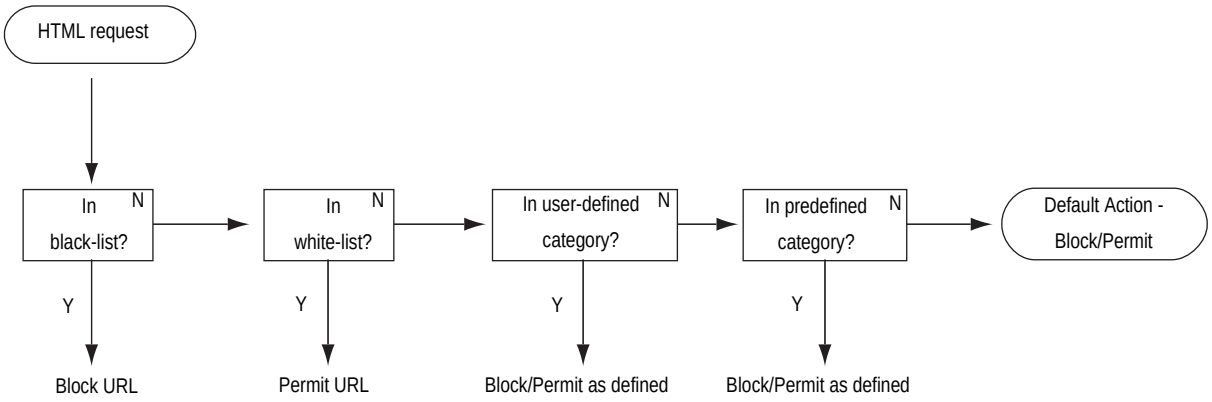
Firewall policies permit or deny specified types of unidirectional traffic between two points. (For information about firewall policies, see “Policies” on page 2-171.) You can enable both antivirus (AV) scanning and integrated web filtering in a policy. (For information about AV scanning, see “Antivirus Scanning” on page 58.)

Enable web filtering in the policy and bind the profile to the policy. When you enable integrated web filtering in a policy, the security device intercepts all HTTP requests. If there is a web-filtering profile bound to the policy, the device matches the URL in the incoming HTTP request to the categories in the profile in the following sequence:

1. Black list
2. White list
3. User-defined categories
4. SurfControl predefined URL categories

If the device is unable to determine the category of the requested URL, then it blocks or permits access based on the default configuration in the profile.

Figure 40: Web-Filtering Profiles and Policies Flowchart



If the device determines that the URL is in a permitted category, and if AV scanning is enabled for that policy, then the device scans the contents for viruses. If the device determines that the URL is in a blocked category, it closes the TCP connection, sends a message alerting the user, and does not perform AV scanning.

Example: Integrated Web Filtering

In this example, you perform the following steps to enable integrated web filtering on the security device and block access to the competitor sites.

1. Create a category called **Competitors**.
2. Add the following URLS to the category: **www.comp1.com** and **www.comp2.com**
3. Create a profile called **my-profile,** and add the **Competitors** category.
4. Apply **my-profile** to a firewall policy.

WebUI

1. **Web Filtering**
   Screening > Web Filtering > Protocol Selection: Select **Integrated (SurfControl)**, then click **Apply**. Then, select **Enable Web Filtering via CPA Server**, and click **Apply** again.

2. **URL Category**
   Screening > Web Filtering > Profile > Custom List > New: Enter the following, then click **Apply**:

   Category Name: Competitors
   URL: www.comp1.com

   Enter the following, then click **OK**:

   URL: www.comp2.com

3. **Web-Filtering Profile**
   Screening > Web Filtering > Profile > Custom Profile > New: Enter the following, then click **Apply**:

   Profile Name: my-profile
   Default Action: Permit

   Category Name: Competitors (select)
   Action: Block (select)
   Configure: Add (select)

4. **Policy**
   Policies > (From: Trust, To: Untrust) New: Enter the following, then click **OK**:

   Source Address:
   Address Book Entry: (select), Any
   Destination Address:
   Address Book Entry: (select), Any
   Service: HTTP
   Web Filtering: (select), my-profile
   Action: Permit

CLI

1. **Web Filtering**
   device->set url protocol sc-cpa
   device(url:sc-cpa)-> set enable

2. **URL Category**
   device(url:sc-cpa)-> set category competitors url www.comp1.com
   device(url:sc-cpa)-> set category competitors url www.comp2.com

3. **Web-Filtering Profile**
   device(url:sc-cpa)-> set profile my-profile other permit
   device(url:sc-cpa)-> set profile my-profile competitors block
   device(url:sc-cpa)-> exit

4. **Firewall Policy**
   device-> set policy id 23 from trust to untrust any any http permit url-filter
   device-> set policy id 23
   device(policy:23)-> set url protocol sc-cpa profile my-profile
   device(policy:23)-> exit
   device-> save

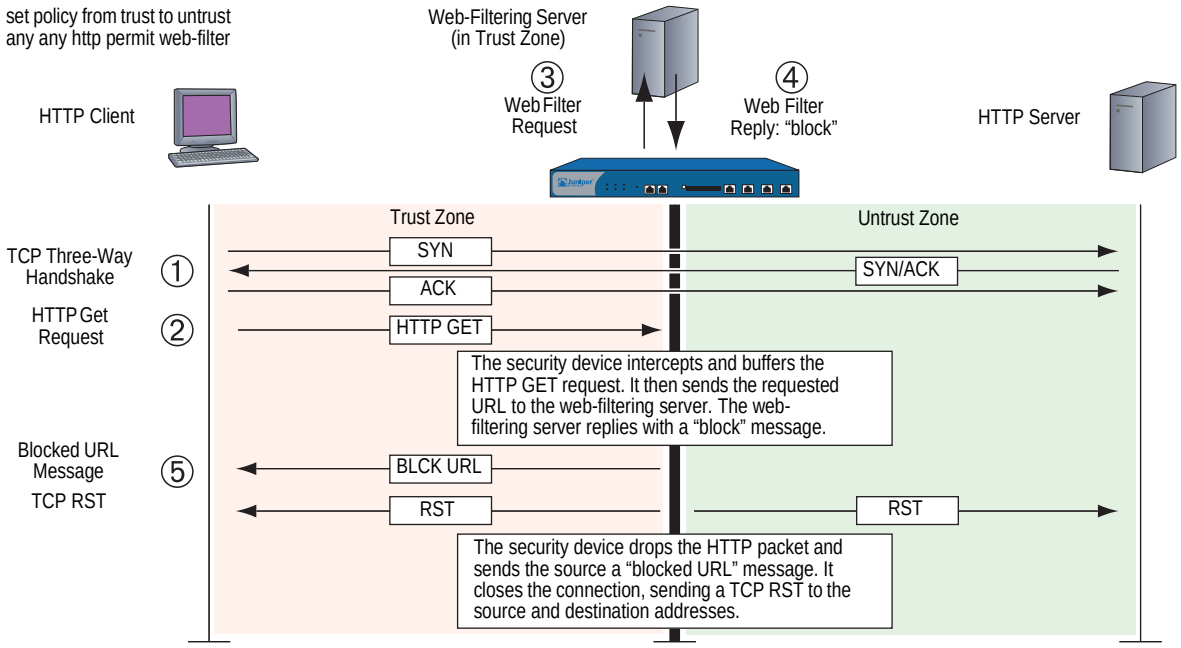
Redirect Web Filtering

Juniper Networks security devices support redirect web filtering using either the Websense Enterprise Engine or the SurfControl Web Filter, both of which enable you to block or permit access to different sites based on their URLs, domain names, and IP addresses. The security device can link directly to a Websense or SurfControl web-filtering server.

NOTE: For additional information about Websense, visit www.websense.com. For additional information about SurfControl, visit www.surfcontrol.com.

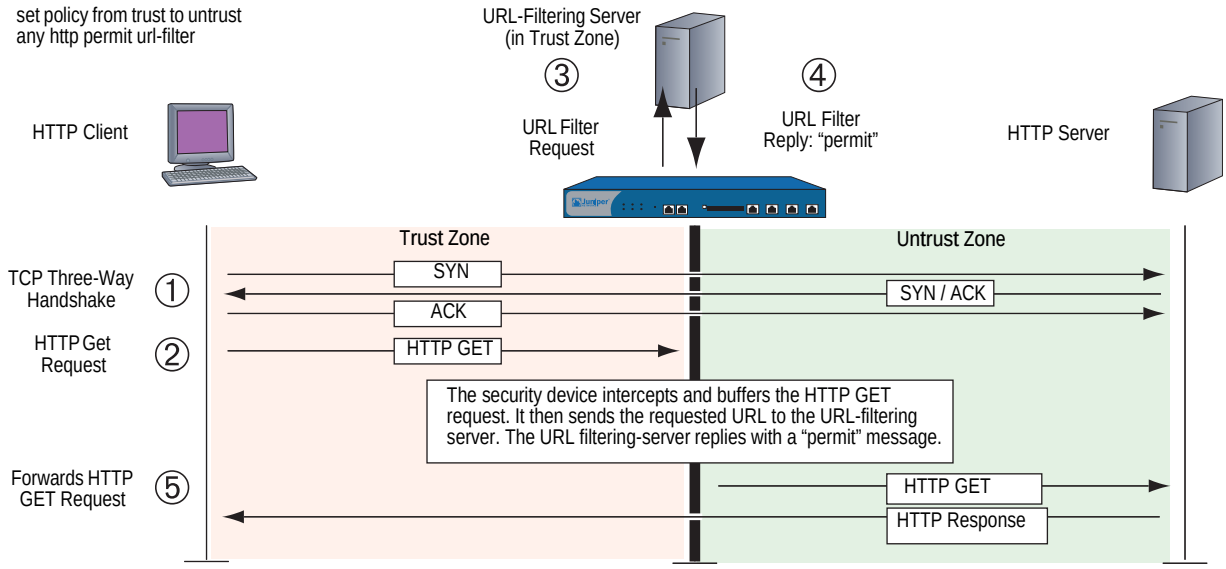
Figure 41 shows the basic sequence of events when a host in the Trust zone attempts an HTTP connection to a server in the Untrust zone. However, web filtering determines that the requested URL is prohibited.

Figure 41: A Blocked URL from Trust Zone to Untrust Zone



If the server permits access to the URL, the sequence of events in the HTTP connection attempt proceeds as shown in Figure 42.

Figure 42: A Permitted URL from Trust Zone to Untrust Zone



Refer to the following sections for more details on redirect Web filtering:

- "Virtual System Support" on this page
- "Configuring Redirect Web Filtering" on page 103
- "Example: Redirect Web Filtering" on page 106

Virtual System Support

Security devices with virtual systems (vsys) support up to eight web-filtering servers—one server reserved for the root system, which can be shared with an unrestricted number of virtual systems, and seven web-filtering servers for private use by the virtual systems. A root-level administrator can configure the web-filtering module at the root and vsys levels. A vsys-level administrator can configure the URL module for his or her own vsys if that vsys has its own dedicated web-filtering server. If the vsys-level administrator uses the root web-filtering server settings, that administrator can view—but not edit—the root-level web-filtering settings.

Alternatively, devices with virtual systems that use Websense web-filtering servers can share all eight Websense servers, not just the root server. Each Websense server can support an unrestricted number of virtual systems, allowing you to balance the traffic load among the eight servers.

To configure multiple virtual systems to connect to a Websense web-filtering server, the root-level or vsys administrator must perform the following steps:

1. Create a account name for each vsys. Use the following CLI command:

   ```
   device-> set url protocol type websense
   device-> set url protocol websense
   device(url:websense)-> set account name
   ```

   When a host in a vsys sends out a URL request, it includes the account name. This name enables the Websense server to identify which vsys sent the URL request.

2. Configure the same web-filtering server settings and system-level parameters for each vsys that shares a Websense web-filtering server. The next section contains information about configuring web-filtering settings and parameters.

Configuring Redirect Web Filtering

To configure a security device to perform redirected web filtering, follow these steps:

1. Set Up a Domain Name System (DNS) Server

The Juniper Networks security device incorporates Domain Name System (DNS) support, allowing you to use domain names as well as IP addresses for identifying locations. You must configure at least one DNS server to enable the security device to resolve the CPA server name to an address. For more information about DNS, see “Domain Name System Support” on page 2-229.

2. Set Up Communication with the Web-Filtering Servers

Configure the security device to communicate with one of the following servers:

- Websense server
- SurfControl server using the SurfControl Content Filtering Protocol (SCFP)

You can set up communications with up to eight web-filtering servers.

WebUI

Screening > Web Filtering > Protocol > Select **Redirect (Websense) or Redirect (SurfControl),** then click **Apply**.

CLI

Enter the web-filtering context for SurfContol (scfp) or Websense (websense) redirect filtering. For more information, see “Using the CLI to Initiate Web-Filtering Modes” on page 91.

```
device-> set url protocol type { websense | scfp }
device-> set url protocol { websense | scfp }
device(url:scfp)-> set server { ip_addr | dom_name } port_num timeout_num
```

Configure the following web-filtering settings at the system level for web-filtering server communication:

- **Source Interface:** The source from which the device initiates web-filter requests to a web-filtering server.
- **Server Name:** The IP address or Fully Qualified Domain Name (FQDN) of the computer running the Websense or SurfControl server.
- **Server Port:** If you have changed the default port on the server, you must also change it on the security device. (The default port for Websense is 15868, and the default port for SurfControl is 62252.) Please see your Websense or SurfControl documentation for full details.
- **Communication Timeout:** The time interval, in seconds, that the device waits for a response from the web-filtering server. If the server does not respond within the time interval, the device either blocks or allows the request. For the time interval, enter a value from 10 through 240.

If a device with multiple virtual systems connects to a Websense server, the virtual systems can share the server. To configure multiple virtual systems to share a Websense server, use the following CLI commands to create an account name for each vsys:

```
device-> set url protocol type websense
device-> set url protocol websense
device(url:websense)-> set account name
```

Once you have configured the vsys names, you define the settings for the web-filtering server and the parameters for the behavior that you want the security device to take when applying web filtering. If you configure these settings in the root system, they also apply to any vsys that shares the web-filtering configuration with the root system. For a vsys, the root and vsys administrators must configure the settings separately. Virtual systems that share the same Websense web-filtering server must have the same web-filtering settings.

3. Enable Web Filtering at the Root and Vsys Levels

You must enable web filtering at the system level. For a device that is hosting virtual systems, enable web filtering for each system that you want to apply it. For example, if you want the root system and a vsys to apply web filtering, enable web filtering in both the root system and that vsys.

To enable web filtering, do either of the following:

WebUI

Screening > Web Filtering > Protocol > Select **Redirect (Websense) or Redirect (SurfControl),** then click **Apply.**
Enable Web Filtering checkbox.

CLI

```
device-> set url protocol type { websense | scfp }
device-> set url protocol { websense | scfp }
device(url:scfp)-> set config enable
```

When web filtering is enabled at the system level, HTTP requests are redirected to a Websense or SurfControl server. This action allows the device to check all HTTP traffic for policies (defined in that system) that require web filtering. If you disable web filtering at the system level, the device ignores the web-filtering component in policies and treats the policies as "permit" policies.

4. Define the System-Level Behavioral Parameters

Define the parameters that you want the system—root or vsys—to use when applying web filtering. One set of parameters can apply to the root system and any vsys that shares the web-filtering configuration with the root system. Other sets can apply to virtual systems that have a dedicated web-filtering server.

The options are as follows:

- **If connectivity to the server is lost:** If the security device loses contact with the web-filtering server, you can specify whether to **Block** or **Permit** all HTTP requests.

- **Blocked URL Message Type:** If you select **NetPartners Websense/SurfControl**, the security device forwards the message it receives in the "block" response from the Websense or SurfControl server. When you select **Juniper Networks**, the device sends the message that you have previously entered in the **Juniper Networks Blocked URL Message** field.

NOTE: If you select **NetScreen**, some of the functions that Websense provides, such as redirection, are suppressed.

- **Juniper Networks Blocked URL Message:** This is the message the security device returns to the user after blocking a site. You can use the message sent from the Websense or SurfControl server, or you can create a message (up to 500 characters) to be sent from the device.

To configure these settings, use either of the following:

WebUI

Screening > Web Filtering > Protocol > Select **Redirect (Websense) or Redirect (SurfControl),** then click **Apply**.

CLI

```
device-> set url protocol type { websense | scfp }
device-> set url protocol { websense | scfp }
device(url:scfp)-> set fail-mode permit
device(url:scfp)-> set deny-message use-server
```

5. Enable Web Filtering in Individual Policies

Configure the device to contact the web-filtering server based on the policy.

To enable web filtering in a policy, use either of the following:

WebUI

Policies > Click **Edit** (edit the policy that you want web filtering to apply), then select the **Web Filter** checkbox.
Select the web-filtering profile from the dropdown box.

CLI

set policy from *zone* to *zone src_addr dst_addr service* permit url-filter

---

NOTE: The device reports the status of the Websense or SurfControl server. To update the status report, click the **Server Status** icon in the WebUI:

Screening > Web Filtering > Protocol > Select **Redirect (Websense) or Redirect (SurfControl),** then click **Apply**.

Example: Redirect Web Filtering

In this example, you configure the security device to do the following:

1. Set the interfaces to work with a SurfControl server at IP address 10.1.2.5, with port number 62252 (default), and have the web-filtering server in the Trust security zone.

2. Enable web filtering on all outbound HTTP traffic from hosts in the Trust zone to hosts in the Untrust zone. If the device loses connectivity with the web-filtering server, the device permits outbound HTTP traffic. When an HTTP client requests access to a prohibited URL, the device sends the following message: “We’re sorry, but the requested URL is prohibited. Contact ntwksec@mycompany.com.”

3. Set both security zones to be in the trust-vr routing domain with the interface for the Untrust zone as ethernet3, IP address 1.1.1.1/24, and the interface for the Trust zone as ethernet1, IP address 10.1.1.1/24. Because the web-filtering server is not in the immediate subnet of one of the device interfaces, a route is added to it through ethernet1 and the internal router at 10.1.1.250.

4. Configure the policy to enable web filtering so that Trust to Untrust permits HTTP service from any source address to any destination address.

WebUI

1. **Interfaces**

   Network > Interfaces > Edit (for ethernet1): Enter the following, then click **Apply**:

   Zone Name: Trust
   Static IP: (select this option when present)
   IP Address/Netmask: 10.1.1.1/24

   Enter the following, then click **OK**:

   Interface Mode: NAT

   Network > Interfaces > Edit (for ethernet3): Enter the following, then click **OK**:

   Zone Name: Untrust
   Static IP: (select this option when present)
   IP Address/Netmask: 1.1.1.1/24

2. **Web-Filtering Server**

   Screening > Web Filtering > Protocol: Select **Redirect (SurfControl)**, then click **Apply**. Then enter the following, and click **Apply** again:

   Enable Web Filtering: (select)
   Server Name: 10.1.2.5
   Server Port: 62252
   Communication Timeout: 10 (seconds)
   If connectivity to the server is lost … all HTTP requests: Permit
   Blocked URL Message Type: NetScreen
   NetScreen Blocked URL Message: We're sorry, but the requested URL is prohibited. Contact ntwksec@mycompany.com.

3. **Routes**

   Network > Routing > Routing Entries > trust-vr New: Enter the following, then click **OK**:

   Network Address/Netmask: 0.0.0.0/0
   Gateway: (select)
   Interface: ethernet3
   Gateway IP Address: 1.1.1.250

   Network > Routing > Routing Entries > trust-vr New: Enter the following, then click **OK**:

   Network Address/Netmask: 10.1.2.0/24
   Gateway: (select)
   Interface: ethernet1
   Gateway IP Address: 10.1.1.250

4. **Policy**

   Policies > (From: Trust, To: Untrust) New: Enter the following, then click **OK**:

   Source Address:
   Address Book Entry: (select), Any
   Destination Address:
   Address Book Entry: (select), Any
   Service: HTTP
   Action: Permit
   Web Filtering: (select)

CLI

1. **Interfaces**

   ```
   set interface ethernet1 zone trust
   set interface ethernet1 ip 10.1.1.1/24
   set interface ethernet3 zone untrust
   set interface ethernet3 ip 1.1.1.1/24
   ```

2. **Web-Filtering Server**

   ```
   device-> set url protocol type scfp
   device-> set url protocol scfp
   device(url:scfp)-> set server 10.1.2.5 62252 10
   device(url:scfp)-> set fail-mode permit
   device(url:scfp)-> set deny-message “We’re sorry, but the requested URL is
       prohibited. Contact ntwksec@mycompany.com.”
   device(url:scfp)-> set config enable
   ```

3. **Routes**

   ```
   set vrouter trust-vr route 0.0.0.0/0 interface ethernet3 gateway 1.1.1.250
   set vrouter trust-vr route 10.1.2.0/24 interface ethernet1 gateway 10.1.1.250
   ```

4. **Policy**

   ```
   set policy from trust to untrust any any http permit url-filter
   save
   ```

Chapter 5
Deep Inspection

You can enable Deep Inspection (DI) in policies to examine permitted traffic and take action if the DI module in ScreenOS finds attack signatures or protocol anomalies. The following sections in this chapter present the DI elements that appear in policies and explains how to configure them:

You can also enable DI at the security zone level for HTTP components. These SCREEN options are explained in the final section of this chapter:

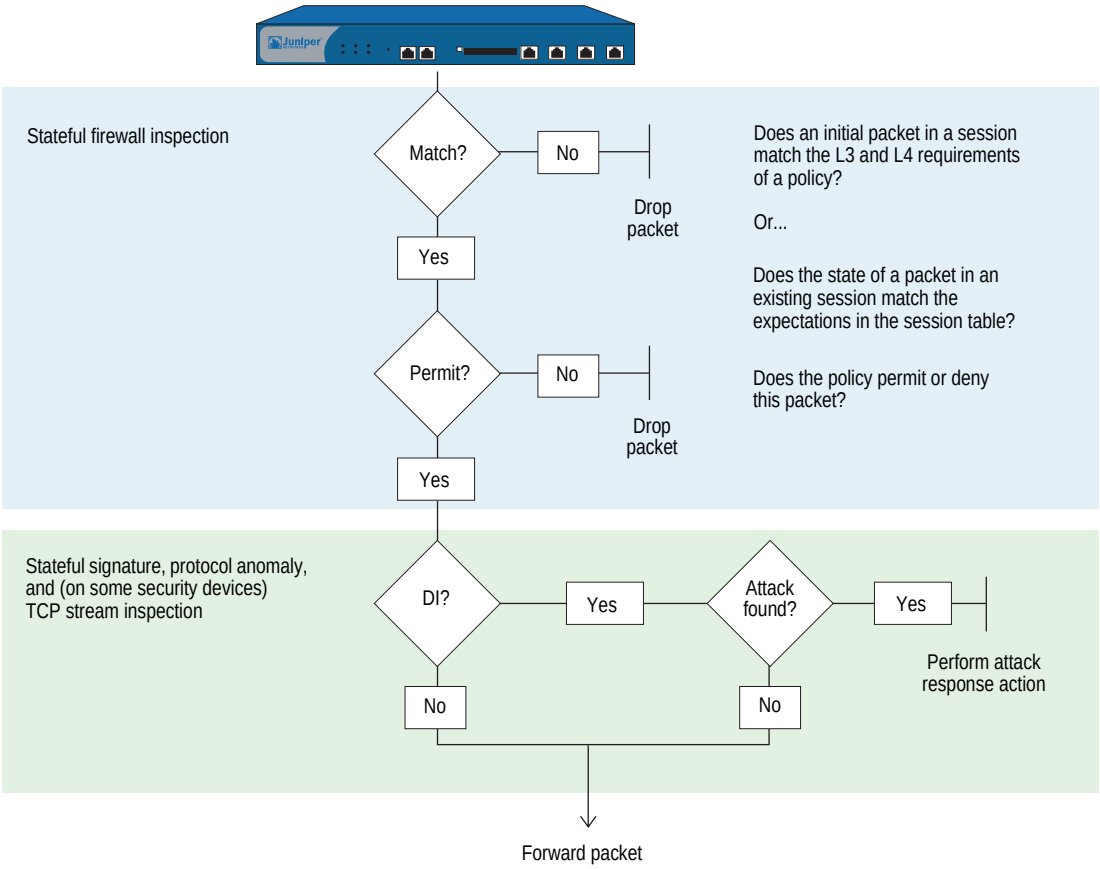
- "Granular Blocking of HTTP Components" on page 160
  - "ActiveX Controls" on page 161
  - "Java Applets" on page 161
  - "EXE Files" on page 161
  - "ZIP Files" on page 161

Overview

Deep Inspection (DI) is a mechanism for filtering the traffic permitted by the Juniper Networks firewall. DI examines Layer 3 and Layer 4 packet headers and Layer 7 application content and protocol characteristics in an effort to detect and prevent any attacks or anomalous behavior that might be present. Figure 43 shows how a packet undergoes Layer 3 inspection.

NOTE: Juniper Networks security devices detect anomalous traffic patterns at Layer 3 and Layer 4 (IP and TCP) via SCREEN options set at the zone level, not the policy level. Examples of IP and TCP traffic-anomaly detection are "IP Address Sweep" on page 8, "Port Scanning" on page 9, and the various flood attacks described in "Network DoS Attacks" on page 34.

Figure 43: Stateful Firewall Inspection



When the security device receives the first packet of a session, it inspects the source and destination IP addresses in the IP packet header (Layer 3 inspection) and the source and destination port numbers and protocol in the TCP segment or UDP datagram header (Layer 4 inspection). If the Layer 3 and 4 components match the criteria specified in a policy, the device then performs the specified action on the packet—permit, deny, or tunnel. When the device receives a packet for an established session, it compares it with the state information maintained in the session table to determine if it belongs to the session.

NOTE: If the specified action is tunnel, the notion of permission is implied. Note that if you enable DI in a policy whose action is tunnel, the security device performs the specified DI operations before encrypting an outbound packet and after decrypting an inbound packet.

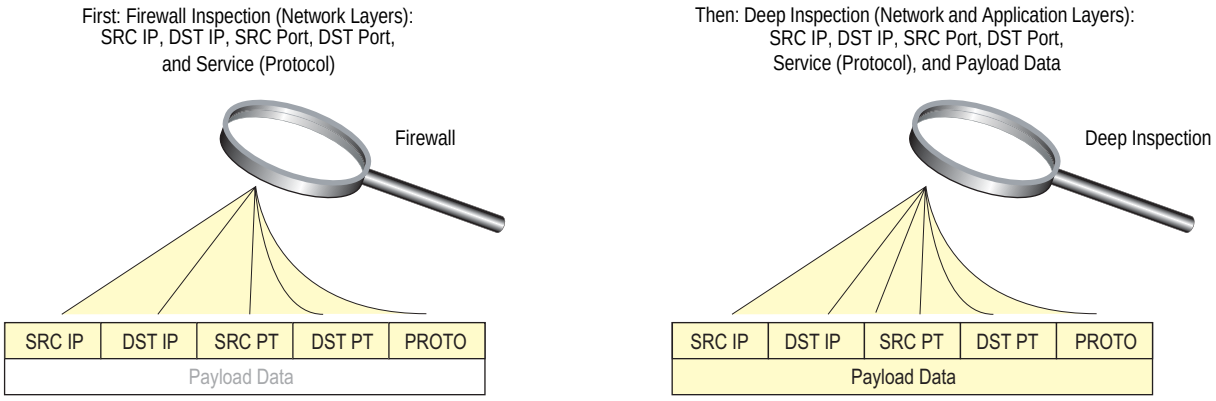
If you have enabled DI in the policy that applies to this packet and the policy action is “permit” or “tunnel,” then the security device further inspects it and its associated data stream for attacks. It scans the packet for patterns that match those defined in one or more groups of attack objects. Attack objects can be attack signatures or protocol anomalies, which you can either define yourself or download to the security device from a database server. (For more information, see “Attack Objects and Groups” on page 121 and “Customized Attack Objects and Groups” on page 149.)

NOTE: The Deep Inspection (DI) feature is available after you have obtained and loaded an advanced mode license key. (If you upgrade from a pre-5.0.0 version of ScreenOS, the mode automatically becomes "advanced." In this case, an advanced-mode license key is not required.)The ability to download signature packs from the database server requires that you first subscribe for the service. For more information, see "Registration and Activation of Subscription Services" on page **2-**264.

Based on the attack objects specified in the policy, the security device might perform the following inspections (see Figure 44):

- Examine header values and payload data for stateful attack signatures
- Compare the format of the transmitted protocol with the standards specified in the RFCs and RFC extensions for that protocol to determine if someone has altered it, possibly for malicious purposes

Figure 44: Firewall Inspection Versus Deep Inspection



If the security device detects an attack, it performs the action specified for the attack object group to which the matching attack object belongs: close, close-client, close-server, drop, drop-packet, ignore, or none. If it does not find an attack, it forwards the packet. (For more information about attack actions, see "Attack Actions" on page 132.)

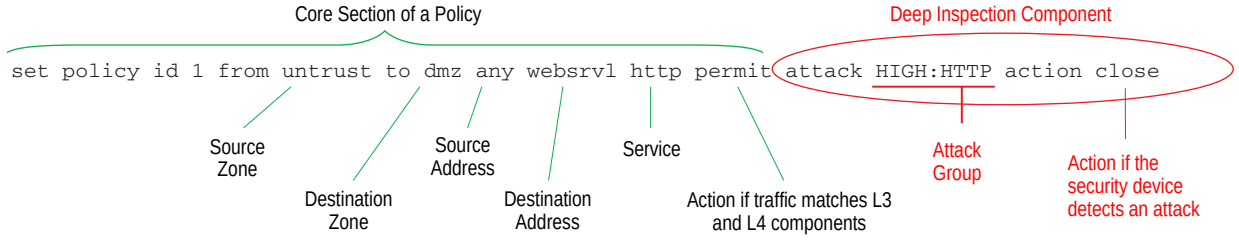
You can conceptually separate a **set policy** command into two parts—the core section and the DI component:

- The core section contains the source and destination zones, source and destination addresses, one or more services, and an action.
- The DI component instructs the security device to inspect traffic permitted by the core section of the policy for patterns matching the attack objects contained in one or more attack object groups. If the security device detects an attack object, it then performs the action defined for the corresponding group.

NOTE: You can optionally add other extensions to the core component of a **set policy** command: VPN and L2TP tunnel references, a schedule reference, address translation specifications, user authentication specifications, antivirus checking, logging, counting, and traffic management settings. Whereas these extensions are optional, the elements that constitute the core of a policy—source and destination zones, source and destination addresses, service (or services), and action—are required. (An exception to this is a global policy, in which no source and destination zones are specified: **set policy global *src_addr dst_addr service action***. For more information about global policies, see "Global Policies" on page **2-**174.)

The following **set policy** command includes a DI component:

Figure 45: DI Component in the Set Policy Command



The above command directs the security device to permit HTTP traffic from any address in the Untrust zone to the destination address "websrv1" in the DMZ zone. It also instructs the device to inspect all HTTP traffic permitted by this policy. If any pattern in the traffic matches an attack object defined in the attack object group "HIGH:HTTP:ANOM", the device closes the connection by dropping the packet and sending TCP RST notifications to the hosts at the source and destination addresses.

It is possible to enter the context of an existing policy by using its ID number. For example:

```
device-> set policy id 1
device(policy:1)->
```

NOTE: The command prompt changes to signal that the subsequent command will be within a particular policy context.

Entering a policy context is convenient if you want to enter several commands related to a single policy. For example, the following set of commands creates a policy that permits HTTP and HTTPS traffic from the any address in the Untrust to websrv1 and websrv2 in the DMZ zone and looks for high and critical HTTP stateful signature and protocol anomaly attacks:

```
device-> set policy id 1 from untrust to dmz any websrv1 http permit attack
    CRITICAL:HTTP:ANOM action close
device-> set policy id 1
device(policy:1)-> set dst-address websrv2
device(policy:1)-> set service https
device(policy:1)-> set attack CRITICAL:HTTP:SIGS action close-server
device(policy:1)-> set attack HIGH:HTTP:ANOM action drop
device(policy:1)-> set attack HIGH:HTTP:SIGS action close-server
device(policy:1)-> exit
device-> save
```

The above configuration permits both HTTP and HTTPS traffic, but only looks for attacks in HTTP traffic. To be able to add attack object groups within a policy context, you must first specify a DI attack and action in the top-level command. In the above example, you can add CRITICAL:HTTP:SIGS, HIGH:HTTP:ANOM, and HIGH:HTTP:SIGS attack object groups because you first configured the policy for DI with the CRITICAL:HTTP:ANOM group.

NOTE: You can specify a different attack action for each attack object group in a policy. If the security device simultaneously detects multiple attacks, it applies the most severe action, which in the above example is "close." For information about the seven attack actions, including their severity levels, see "Attack Actions" on page 132.

Attack Object Database Server

The attack object database server contains all the predefined attack objects, organized into attack object groups by protocol and severity level. Juniper Networks stores the attack object database on a server at https://services.netscreen.com/restricted/sigupdates.

Predefined Signature Packs

The attack object database is organized into four signature packs, base, server protection, client protection, and worm mitigation. This approach is ideal because of the limited device memory and increased protocol support in the signature packs. Table 5 describes each of the predefined signature packs and the threat coverage.

Table 5: Predefined Signature Packs

Signature Pack	Description	Threat Coverage
Base[1]	A selected set of signatures for client/server and worm protection optimized for remote and branch offices along with small/medium businesses.	Includes a sample of worm, client-to-server, and server-to-client signatures for Internet-facing protocols and services, such as HTTP, DNS, FTP, SMTP, POP3, IMAP, NetBIOS/SMB, MS-RPC, P2P, and IM (AIM, YMSG, MSN, and IRC).
Server protection	For small/medium enterprises and remote and branch offices of large enterprises needing perimeter defense and compliance for server infrastructure, such as IIS, and Exchange.	Primarily focuses on protecting a server farm. It includes a comprehensive set of server-oriented protocols, such as HTTP, DNS, FTP, SMTP, IMAP, MS-SQL, and LDAP. Also includes worm signatures that target servers.
Client protection	For small/medium enterprises and remote and branch offices of large enterprises needing perimeter defense and compliance for hosts (desktops, laptops, and so on).	Primarily focuses on protecting users from getting malware, Trojans, and so on while surfing the Internet. Includes a comprehensive set of client-oriented protocols, such as HTTP, DNS, FTP, IMAP, POP3, P2P, and IM (AIM, YMSG, MSN, and IRC). Also includes worm signatures that target clients.
Worm Mitigation	For remote and branch offices of large enterprises along with small/medium businesses to provide the most comprehensive defense against worm attacks.	Includes stream signatures and primarily focuses on providing comprehensive worm protection. Detects server-to-client and client-to-server worm attacks for all protocols.

1.Due to memory allocation required for new enhancements, only DI signatures of critical severity are provided for NS-5XT/GT devices.

Table 6 lists the predefined signatures packs with the corresponding URLs.

Table 6: URLs for Predefined Signature Packs

Signature Pack	URL
Base (default)	https://services.netscreen.com/restricted/sigupdates The security device uses this URL by default.
Server	https://services.netscreen.com/restricted/sigupdates/server
Client	https://services.netscreen.com/restricted/sigupdates/client
Worm-mitigation	https://services.netscreen.com/restricted/sigupdates/worm

Updating Signature Packs

Juniper Networks stores the four predefined signature packs on an attack object database server at https://services.netscreen.com/restricted/sigupdates. To gain access to this database server, you must first subscribe to the DI signature service for your device as described in the next section.

There are four ways to update the database:

- “Immediate Update” on page 116
- “Automatic Update” on page 117
- “Automatic Notification and Immediate Update” on page 118
- “Manual Update” on page 119

NOTE: You can also use NetScreen-Security Manager to download the signature packs. For information, see the *NetScreen-Security Manager Administration Guide*.

Before You Start Updating Attack Objects

Before you start downloading and updating attack objects, you must do the following:

1. Register your security device and obtain an authorization code.
2. Purchase a license key and activate a subscription for Deep Inspection.
3. Verify that the system clock and the Domain Name System (DNS) settings on your device are accurate.

For more information, see "Registration and Activation of Subscription Services" on page **2-**264.

WebUI

Configuration > Date/Time

Network > DNS > Host

4. Click the **Update Now** button.

   Note that this option is only available after you retrieve a Deep Inspection subscription key.

   The security device then attempts to contact the server at the default URL: https://services.netscreen.com/restricted/sigupdates; or, if you have entered a different URL in the Database Server field, it attempts to contact the URL that you entered. Table 6 on page 115 lists the predefined signatures packs and the corresponding URLs.

   After a few moments, a message appears indicating whether the update was successful. If the update was unsuccessful, then check the event log to determine the cause of the failure.

NOTE: After you download the signature pack the first time, you must reset the security device. Following each download thereafter, resetting the device is unnecessary.

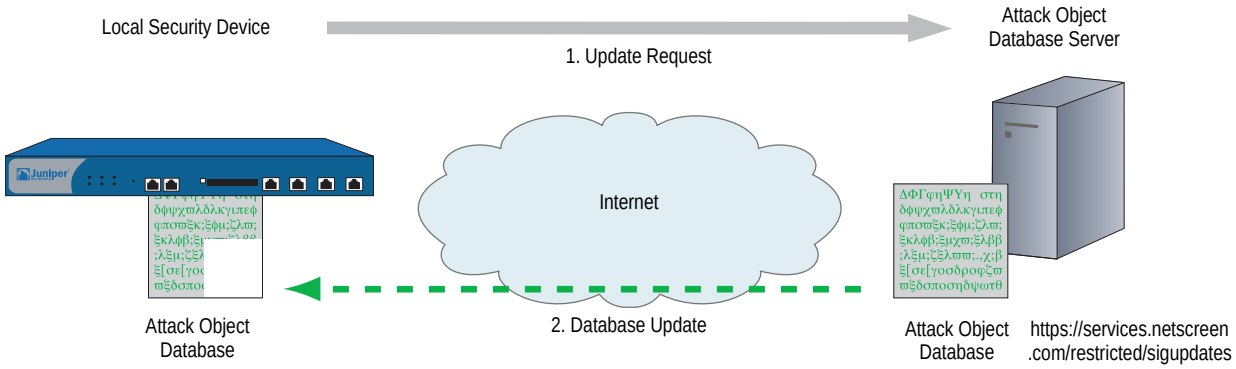
Immediate Update

The **Immediate Update** option allows you to update the signature pack on the security device immediately with the signature pack stores on the database server. For this operation to work, you must first configure the attack object database server settings.

In this example (see Figure 46), you save a predefined signature pack from the attack object database server to the security device immediately.

You do not set a schedule for updating the database on the security device. Instead, you save the database from the server to the security device immediately.

Figure 46: Updating DI Signatures Immediately



WebUI

Configuration > Update > Attack Signature:
Signature Pack: Client
Click the **Update Now** button.

CLI

```
set attack db sigpack client
exec attack-db update
Loading attack database.............
Done.
Done.
Switching attack database...Done
Saving attack database to flash...Done.
```

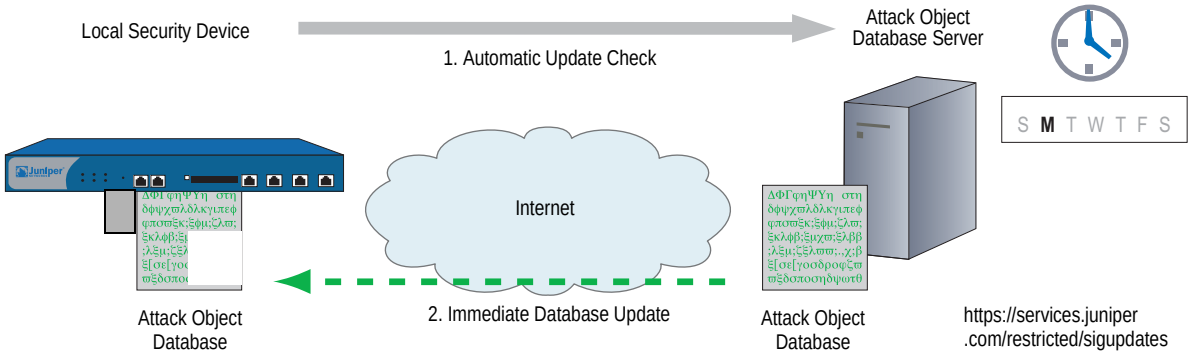
Automatic Update

The **Automatic Update** option, downloads the signature pack at user-scheduled times if the database on the server is a newer version than that previously loaded on the device. Juniper Networks updates the signature pack on a regular basis with newly discovered attack patterns. Therefore, because of its changing nature, you must update the signature pack on your security device regularly too. For this operation to work, you must first configure the attack object database server settings.

In this example (see Figure 47), you set a schedule to update the database on the security device every Monday at 4:00 AM. At that scheduled time, the device compares the version of the database on the server with that on the device. If the version on the server is more recent, the security device automatically replaces its database with the newer version.

For example, select Server to update the server signature pack. See Table 6 on page 115 for a list of predefined signatures packs and the corresponding URLs.

Figure 47: Updating DI Signatures Automatically



WebUI

Configuration > Update > Attack Signature: Enter the following, then click **OK**:

Signature Pack: Server
Update Mode: Automatic Update
Schedule:
Weekly on: Monday
Time (hh:mm): 04:00

NOTE: If you schedule updates on a monthly basis and the date you choose does not occur in a month (for example, 31 does not occur in several months), the security device uses the last possible date of the month in its place.

CLI

```
set attack db sigpack server
set attack db mode update
set attack db schedule weekly monday 04:00
save
```

Automatic Notification and Immediate Update

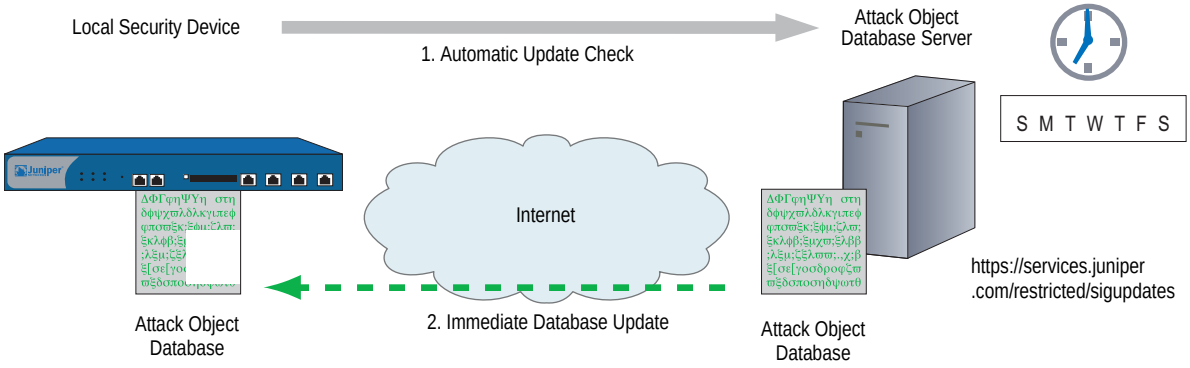
The **Automatic Notification** and I**mmediate Update** option allows you to check at user-scheduled times if the data on the database server is more recent than that on the security device. If the data on the server is more recent, a notice appears on the Home page in the WebUI, and in the CLI after you log into the security device. You can then enter the exec attack-db update command or click the Update Now button on the Configuration > Update > Attack Signature page in the WebUI to save the signature pack from the server to the device. For the server-checking operation semi-automatic procedure to work, you must first configure the attack object database server settings.

In this example (see Figure 48), you set a schedule to check the database on the security device every day at 07:00 AM.

When you receive a notice that the database on the server has been updated, you click the **Update Now** button on the Configuration > Update > Attack Signature page in the WebUI or enter the **exec attack-db update** command to save the database from the server to the device.

For example, do the following to update the Client signature pack. See Table 6 on page 115 for a list of predefined signatures packs and the corresponding URLs.

Figure 48: Notifying Signature Updates



WebUI

1. **Scheduled Database Checking**
   Configuration > Update > Attack Signature: Enter the following, then click **OK**:

   Signature Pack: Client
   Update Mode: Automatic Notification
   Schedule:
   Daily
   Time (hh:mm): 07:00

2. **Immediate Database Update**
   When you receive a notice that the attack database on the server is more current than the one on the security device, do the following:

   Configuration > Update > Attack Signature

   Signature pack: Client
   Click the **Update Now** button.

CLI

1. **Scheduled Database Checking**
   set attack db sigpack client
   set attack db mode notification
   set attack db schedule daily 07:00

2. **Immediate Database Update**
   When you receive a notice that the attack database on the server is more current than the one on the security device, do the following:

   exec attack-db update

Manual Update

The **Manual Update** option, allows you to first use a browser to download the signature pack to a local directory or TFTP server directory. You can then load the signature pack on the security device using either the WebUI (from the local directory) or CLI (from the TFTP server directory).

NOTE: Before performing an immediate database update, you can use the **exec attack-db check** command to check if the attack object database on the server is more recent than the one on the security device.

In this example (see Figure 49), you manually save the latest signature pack to the local directory “C:\netscreen\attacks-db” (if you want to use the WebUI to load the database) or C:\Program Files\TFTP Server (if you want to use the CLI to load it). You then load the database on the security device from your local directory.

NOTE: After downloading the signature pack, you can also post it on a local server and set it up for other security devices to access. The admins for the other devices must then change the database server URL to that of the new location. They can either enter the new URL in the Database Server field on the Configuration > Update > Attack Signature page or use the following CLI command: **set attack db server** *url_string*.

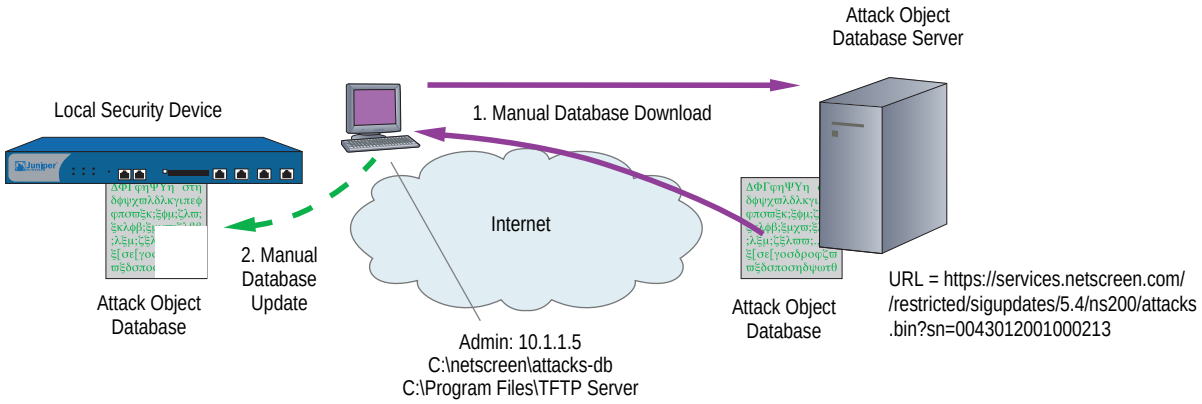
For an automatic update, the security device automatically adds the following elements to the URL:

- Serial number of the security device
- Number of the major ScreenOS version running on the device
- Platform type

When you manually update the DI Signatures, you must add these elements yourself. In this example, the serial number is 0043012001000213, the ScreenOS version is 5.4, and the platform is NetScreen-208 (ns200). Consequently, the resulting URL is:

https://services.netscreen.com/restricted/sigupdates/5.4/ns200/attacks.bin?sn=0043012001000213

Figure 49: Updating DI Signatures Manually



1. Downloading the Signature Pack

To save the signature pack to your local server, enter the following URL in the address field of your browser. See Table 6 on page 115 for a list of predefined signatures packs and the corresponding URLs.

https://services.netscreen.com/restricted/sigupdates/5.4/ns200/attacks.bin?sn=0043012001000213

Save *attacks.bin* to the local directory "C:\netscreen\attacks-db" (for loading via the WebUI) or to your TFTP server directory C:\Program Files\TFTP Server (when you want to use the CLI to load it).

2. Updating the Signature Pack

WebUI

Configuration > Update > Attack Signature: Enter the following, then click **OK**:

Deep Inspection Signature Update:
Load File: Enter **C:\netscreen\attacks-db\attacks.bin**

Or

Click **Browse** and navigate to that directory, select **attacks.bin**, then click **Open**.

If you downloaded the server, client, or worm protection signature packs, then enter the appropriate filename.

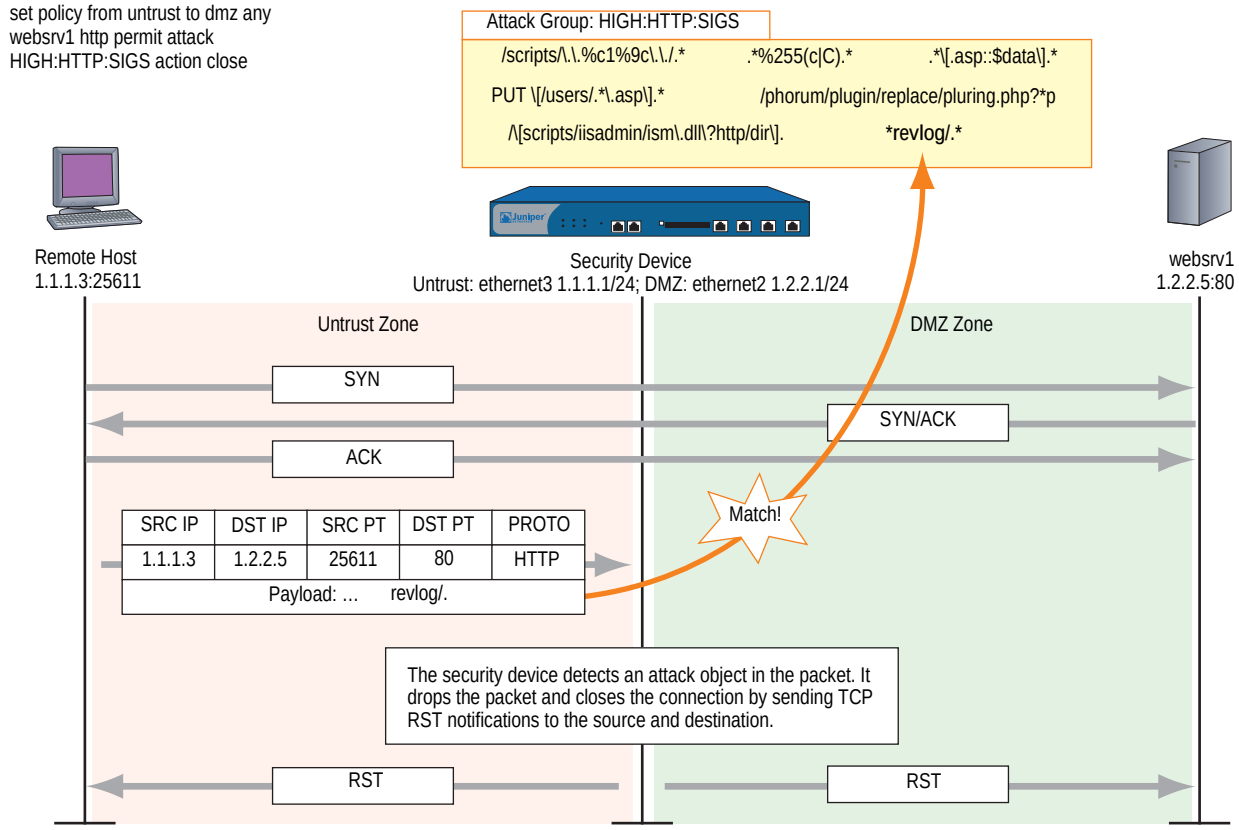
CLI

save attack-db from tftp 10.1.1.5 attacks.bin to flash

Attack Objects and Groups

Attack objects are stateful signatures, stream signatures (on the NetScreen-5000 series), and protocol anomalies that a security device uses to detect attacks aimed at compromising one or more hosts on a network. Attack objects are in groups organized by protocol type and then by severity. When you add Deep Inspection (DI) to a policy, the device inspects the traffic that the policy permits for any patterns matching those in the referenced attack object group (or groups).

Figure 50: Attack Objects and Groups



The attack object groups that you reference in the DI component of a policy must target the same service type that the policy permits. For example, if the policy permits SMTP traffic, the attack object group must aim at attacks on SMTP traffic. The following policy exemplifies a valid configuration:

set policy id 2 from trust to untrust any any **smtp** permit attack **CRIT:SMTP:SIGS** action close

The next policy is erroneous because the policy permits SMTP traffic, but the attack object group is for POP3 traffic:

set policy id 2 from trust to untrust any any **smtp** permit attack **CRIT:POP3:SIGS** action close

The second policy is configured incorrectly and, if implemented, would cause the security device to expend unnecessary resources inspecting SMTP traffic for POP3 attack objects that it could never find. If policy 2 permits both SMTP and POP3 traffic, you can configure the DI component to check for SMTP attack objects, POP3 attack objects, or for both.

set group service **grp1**
set group service **grp1** add **smtp**
set group service **grp1** add **pop3**
set policy id 2 from trust to untrust any any **grp1** permit attack **CRIT:SMTP:SIGS** action close
set policy id 2 attack **CRIT:POP3:SIGS** action close

Supported Protocols

The Deep Inspection (DI) module supports stateful signature attack objects and protocol anomaly attack objects for the following protocols and applications:

Table 7: Basic Network Protocols

Protocol	Stateful Signature	Protocol Anomaly	Definition
DNS	Yes	Yes	Domain Name System (DNS) is a database system for translating domain names to IP addresses, such as www.juniper.net = 207.17.137.68.
FTP	Yes	Yes	File Transfer Protocol (FTP) is a protocol for exchanging files between computers across a network.
HTTP	Yes	Yes	HyperText Transfer Protocol (HTTP) is a protocol primarily used to transfer information from webservers to web clients.
IMAP	Yes	Yes	Internet Mail Access Protocol (IMAP) is a protocol that provides incoming e-mail storage and retrieval services, with the option that users can either download their e-mail or leave it on the IMAP server.
NetBIOS	Yes	Yes	NetBIOS (Network Basic Input Output System) is an application interface that allows applications on users' workstations to access network services provided by network transports such as NetBEUI, SPX/IPX, and TCP/IP.
POP3	Yes	Yes	Post Office Protocol, version 3 (POP3) is a protocol that provides incoming e-mail storage and retrieval services.
SMTP	Yes	Yes	Simple Mail Transfer Protocol (SMTP) is a protocol for transferring e-mail between mail servers.
Chargen	Yes	Yes	Character generator protocol
DHCP	Yes	Yes	Dynamic Host Configuration Protocol is used to control vital networking parameters of hosts (running clients) with the help of a server. DHCP is backward compatible with BOOTP.
Discard	Yes	Yes	Discard protocol is a useful debugging and measurement tool. A discard service simply throws away any data it receives.
Echo	Yes	Yes	Echo protocol is an internet protocol intended for testing and measurement purposes. A host may connect to a server that supports the ECHO protocol, on either TCP or UDP port 7. The server then sends back any data it receives.
Finger	Yes	Yes	Finger User Information protocol is a simple protocol that provides an interface to a remote user information program.
Gopher	Yes	Yes	Gopher is an internet protocol designed for distributed document search and retrieval.
ICMP	Yes	Yes	Internet Control Message Protocol is a required protocol tightly integrated with IP. ICMP messages, delivered in IP packets, are used for out-of-band messages related to network operation.
IDENT	Yes	Yes	Identification protocol provides a means to determine the identity of a user of a particular TCP connection.
LDAP	Yes	Yes	Lightweight Directory Access Protocol is a set of protocols for accessing information directories.
LPR	Yes	Yes	Line Printer spooler
NFS	Yes	Yes	Network File System (NFS) protocol provides transparent remote access to shared files across networks. The NFS protocol is designed to be portable across different machines, operating systems, network architectures, and transport protocols.
NNTP	Yes	Yes	Network News Transfer Protocol specifies a protocol for the distribution, inquiry, retrieval, and posting of news articles using a reliable stream-based transmission of news.

Protocol	Stateful Signature	Protocol Anomaly	Definition
NTP	Yes	Yes	Network Time Protocol and Simple Network Time Protocol is used to synchronize the time of a computer client or server to another server or reference time source, such as a radio or satellite receiver or modem.
Portmapper	Yes	Yes	Port Mapper Program Protocol maps RPC program and version numbers to transport- specific port numbers.
RADIUS	Yes	Yes	Remote Authentication Dial In User Service, an authentication and accounting system used by many Internet Service Providers (ISPs).
rexec	Yes	Yes	Remote Execution
rlogin	Yes	Yes	Remote Login occurs when a user connects to an Internet host to use its native user interface.
rsh	Yes	Yes	Remote shell
RTSP	Yes	Yes	Real Time Streaming Protocol is a client-server application-level protocol for controlling the delivery of data with real-time properties. It establishes and controls either a single or several time-synchronized streams of continuous media, such as audio and video.
SNMPTRAP	Yes	Yes	Simple Network Management Protocol is an SNMP application that uses the SNMP TRAP operation to send information to a network manager.
SSH	Yes	Yes	Secure Shell Protocol is a protocol for secure remote login and other secure network services over an insecure network.
SSL	Yes	Yes	Secure Sockets Layer is a protocol used for transmitting private documents via the Internet using a cryptographic system.
syslog	Yes	Yes	System Logging Protocol is used for the transmission of event notification messages across networks.
Telnet	Yes	Yes	Telnet protocol is a terminal emulation program for TCP/IP networks. This protocol enables you to communicate with other servers on the network.
TFTP	Yes	Yes	Trivial File Transfer Protocol is a simple protocol used to transfer files. TFTP uses the User Datagram Protocol (UDP) and provides no security features.
VNC	Yes	Yes	Virtual Network Computing is a desktop protocol to remotely control another computer.
Whois	Yes	Yes	Network Directory Service Protocol is a TCP transaction based query/response server that provides network-wide directory service to internet users.

Table 8: Instant Messaging Applications

Protocol	Stateful Signature	Protocol Anomaly	Definition
AIM	Yes	Yes	America Online Instant Messaging (AIM) is the instant messaging application for America Online.
MSN Messenger	Yes	Yes	Microsoft Network Messenger (MSN Messenger) is the instant messaging service provided by Microsoft.
Yahoo! Messenger	Yes	Yes	Yahoo! Messenger is the instant messaging service provided by Yahoo!.
IRC	Yes	Yes	Internet Relay Chat is a text-based protocol, with the simplest client being any socket program capable of connecting to the server.

Table 9: Peer-to-Peer (P2P) Networking Applications

Protocol	Stateful Signature	Protocol Anomaly	Definition
BitTorrent	Yes	No	BitTorrent is a P2P file distribution tool, designed to provide an efficient way to distribute the same file to a large group by having everybody that downloads a file also upload it to others.
DC (Direct Connect)	Yes	No	DC (Direct Connect) is a P2P file-sharing application. A DC network uses hubs to connect groups of users, often with a requirement that they share a certain amount of bytes or files. Many hubs feature special areas of interest, creating small communities for connected users.
eDonkey	Yes	No	eDonkey is a decentralized P2P file-sharing application that uses the Multisource File Transfer Protocol (MFTP). The eDonkey network supports two kinds of applications: clients and servers. Clients connect to the network and share files. Servers act as meeting hubs for the clients.
Gnutella	Yes	Yes	Gnutella is a P2P file-sharing protocol and application without any centralized servers. Some other applications using the Gnutella protocol are BearShare, Limewire, Morpheus, and ToadNode.
KaZaa	Yes	No	KaZaa is a decentralized P2P file-sharing application using the FastTrack protocol. KaZaa is mainly used for sharing MP3 files.
MLdonkey	Yes	No	MLdonkey is a P2P client application that can run on multiple platforms and can access multiple P2P networks, such as BitTorrent, DC, eDonkey, FastTrack (KaZaa and others), and Gnutella and Gnutella2.
Skype	Yes	No	Skype is a free P2P Internet telephony service that allows users to talk with each other over a TCP/IP network such as the Internet.
SMB	Yes	Yes	SMB (Server Message Block) is a protocol for sharing such resources as files and printers among computers. SMB operates on top of the NetBIOS protocol.
WinMX	Yes	No	WinMX is a P2P file-sharing application that allows a client to connect to several servers simultaneously

NOTE: Many of the listed P2P applications use their own proprietary protocols.

Table 10: Application Layer Gateways (ALGs)

Protocol	Stateful Signature	Protocol Anomaly	Definition
MSRPC	Yes	Yes	MSRPC (Microsoft-Remote Procedure Call) is a mechanism for running processes on a remote computer.

If the security device has access to http://help.juniper.net/sigupdates/english, you can see the contents of all the predefined attack object groups and descriptions of the predefined attack objects. Open your browser, and enter one of the following URLs in the Address field:

http://help.juniper.net/sigupdates/english/AIM.html
http://help.juniper.net/sigupdates/english/DNS.html
http://help.juniper.net/sigupdates/english/FTP.html
http://help.juniper.net/sigupdates/english/GNUTELLA.html
http://help.juniper.net/sigupdates/english/HTTP.html
http://help.juniper.net/sigupdates/english/IMAP.html
http://help.juniper.net/sigupdates/english/MSN.html
http://help.juniper.net/sigupdates/english/NBDS.html
http://help.juniper.net/sigupdates/english/NBNAME.html

http://help.juniper.net/sigupdates/english/POP3.html
http://help.juniper.net/sigupdates/english/SMTP.html
http://help.juniper.net/sigupdates/english/MSRPC.html
http://help.juniper.net/sigupdates/english/SMB.html
http://help.juniper.net/sigupdates/english/YMSG.html

Each of the above URLs links to an HTML page containing a list of all the predefined attack objects—organized in groups by severity—for a particular protocol. To see a description of an attack object, click its name.

Stateful Signatures

An attack signature is a pattern that exists when a particular exploit is in progress. The signature can be a Layer 3 or 4 traffic pattern, such as when one address sends lots of packets to different port numbers at another address (see "Port Scanning" on page 9), or a textual pattern, such as when a malicious URL string appears in the data payload of a single HTTP or FTP packet. The string can also be a specific segment of code or a specific value in the packet header. However, when searching for a textual pattern, the Deep Inspection (DI) module in a security device looks for more than just a signature in a packet; it looks for the signature in a particular portion of the packet (even if fragmented or segmented), in packets sent at a particular time in the life of the session, and sent by either the connection initiator or the responder.

NOTE: Because the DI module supports regular expressions, it can use wildcards when searching for patterns. Thus, a single attack signature definition can apply to multiple attack pattern variations. For information about regular expressions, see "Regular Expressions" on page 150.

When the DI module checks for a textual pattern, it considers the roles of the participants as client or server and monitors the state of the session to narrow its search to just those elements relevant to the exploit for which attackers use the pattern. Using contextual information to refine packet examination greatly reduces false alarms—or "false positives"—and avoids unnecessary processing. The term "stateful signatures" conveys this concept of looking for signatures within the context of the participants' roles and session state.

To see the advantage of considering the context in which a signature occurs, note the way the DI module examines packets when enabled to detect the EXPN Root attack. Attackers use the EXPN Root attack to expand and expose mailing lists on a mail server. To detect the EXPN Root attack, the security device searches for the signature "expn root" in the control portion of a Simple Mail Transfer Protocol (SMTP) session. The device examines only the control portion because that is only where the attack can occur. If "expn root" occurs in any other portion of the session, it is not an attack.

Using a simple textual packet signature detection technique, the signature "expn root" triggers an alarm even if it appears in the data portion of the SMTP connection; that is, in the body of an e-mail message. If, for example, you were writing to a colleague about EXPN Root attacks, a simple packet signature detector would regard this as an attack. Using stateful signatures, the DI module can distinguish between text strings that signal attacks and those that are harmless occurrences.

NOTE: For a list of protocols for which there are predefined stateful signature attack objects, see “Supported Protocols” on page 123.

TCP Stream Signatures

Like a stateful signature, a TCP stream signature is a pattern that exists when an exploit is in progress. However, when the DI module examines traffic for stateful signatures, it searches only within specific contexts. When the DI module examines traffic for TCP stream signatures, it does so without regard for contexts. Another distinction between the two types of signatures is that although stateful signatures can be predefined or user-defined, TCP stream signatures must be user-defined. After you add a stream signature attack object to an attack object group, you can then reference that group in a policy that applies DI. (For more about TCP stream signatures, see “TCP Stream Signature Attack Objects” on page 153.)

NOTE: You can define TCP stream signatures on the high-end systems only.

Stream signatures are independent of protocols and are therefore more flexible in matching traffic. Stream signatures can examine traffic where protocols decoders can’t inspect. However, this flexibility affects performance and resource consumption.

Stream signatures consume resources and affect performance, so they must be used sparingly. Stream256 signatures however, operate the same way, but rather than matching over the entire stream, they only match on the first 256 bytes of the stream. Therefore, they consume fewer resources and are less of a performance hit.

Protocol Anomalies

Attack objects that search for protocol anomalies detect traffic that deviates from the standards defined in RFCs and common RFC extensions. With signature attack objects, you must use a predefined pattern or create a new one; therefore, they can only detect known attacks. Protocol anomaly detection is particularly useful for catching new attacks or those attacks that cannot be defined by a textual pattern.

NOTE: For a list of protocols for which there are predefined protocol anomaly attack objects, see "Supported Protocols" on page 123.

Attack Object Groups

Predefined attack object groups contain attack objects for a specific protocol. For each protocol, the groups are separated into protocol anomalies and stateful signatures, and then roughly organized by severity. The three attack object group severity levels are critical, high, and medium:

- **Critical:** Contains attack objects matching exploits that attempt to evade detection, cause a network device to crash, or gain system-level privileges.
- **High:** Contains attack objects matching exploits that attempt to disrupt a service, gain user-level access to a network device, or activate a Trojan horse previously loaded on a device.
- **Medium:** Contains attack objects matching exploits that detect reconnaissance efforts attempting to access vital information through directory traversal or information leaks.
- **Low:** Contains attack objects matching exploits that attempt to obtain non-critical information or scan a network with a scanning tool.
- **Info:** Contains attack objects matching normal, harmless traffic containing URLs, DNS lookup failures, SNMP public community strings, and Peer-to-Peer (P2P) parameters. You can use informational attack objects to obtain information about your network.

Changing Severity Levels

Although attack object groups are classified by protocol and severity level (critical, high, medium), each attack object has its own specific severity level: critical, high, medium, low, info. These attack object severity levels map to severity levels for event log entries as follows:

Table 11: Attack Object Severity Levels

Attack Object Severity Level	– Maps to –	Event Log Entry Severity Level
Critical		Critical
High		Error
Medium		Warning
Low		Notification
Info		Information

For example, if the security device detects an attack with the severity level "Medium," the corresponding entry that appears in the event log then has the severity level "Warning."

It is possible to override the default severity level of all attack objects in one or more attack object groups referenced in a policy. You do this at the policy level by entering the context of an existing policy and then assigning a new severity level to all the attack object groups that the policy references.

The following shows how to change the severity level of the attack object groups referenced in a policy through the WebUI and CLI:

WebUI

Policies > Edit (for an existing policy): Do the following, then click **OK**:

> Deep Inspection: Select a severity option in the Severity drop-down list, then click **OK**.

CLI

```
device-> set policy id number
device(policy:number)-> set di-severity { info | low | medium | high | critical }
```

To return the severity level for each attack object to its original setting, you again enter the context of a policy and do either of the following:

WebUI

Policies > Edit (for an existing policy): Do the following, then click **OK**:

> Deep Inspection: Select **Default** in the Severity drop-down list, then click **OK**.

CLI

```
device-> set policy id number
device(policy:number)-> unset di-severity
```

Example: Deep Inspection for P2P

In this example, you permit any host in the Trust zone to initiate a peer-to-peer (P2P) session with any host in the Untrust zone using HTTP, DNS, and Gnutella services. You then apply Deep Inspection (DI) to the permitted traffic to check for stateful signatures and protocol anomalies as defined in the following attack object groups:

- INFO:DNS:SIGS
- INFO:GNUTELLA:ANOM
- INFO:HTTP:SIGS

NOTE: For security reasons, you do not define a policy permitting any host in the Untrust zone to initiate a P2P session with a host in the Trust zone.

If the security device detects a signature or anomalous behavior, it severs the connection and sends a TCP RST to the client to close the session. You also enable the logging of any discovered attack, which is the default behavior.

NOTE: For information about the various attack actions that the security device can perform, see “Attack Actions” on page 132. For information about logging detected attacks, see “Attack Logging” on page 143.

WebUI

Policies > (From: Trust, To: Untrust) New: Enter the following, then click **OK**:

Source Address:
Address Book Entry: (select), Any
Destination Address:
Address Book Entry: (select), Any
Service: DNS

> Click **Multiple**, select **GNUTELLA** and **HTTP**, then click **OK** to return to the basic policy configuration page.

Action: Permit

> Click **Deep Inspection**, enter the following, click **Add** to enter each attack object group, then click **OK** to return to the basic policy configuration page:

Severity: Default
Group: INFO:DNS:SIGS
Action: Close Client
Log: (select)
Severity: Default
Group: INFO:GNUTELLA:ANOM
Action: Close Client
Log: (select)
Severity: Default
Group: INFO:HTTP:SIGS
Action: Close Client
Log: (select)

CLI

```
set policy id 1 from trust to untrust any any dns permit attack
INFO:DNS:SIGS action close-client
set policy id 1
device(policy:1)-> set service gnutella
device(policy:1)-> set service http
device(policy:1)-> set attack INFO:GNUTELLA:ANOM action close-client
device(policy:1)-> set attack INFO:HTTP:SIGS action close-client
device(policy:1)-> exit
save
```

NOTE: Because the logging of detected attacks is enabled by default, you do not have to specify logging through CLI commands.

Disabling Attack Objects

When you reference an attack object group in a policy, the security device checks the traffic to which the policy applies for patterns matching any of the attack objects in that group. At some point, you might not want to use a particular attack object if it repeatedly produces false-positives; that is, if it erroneously interprets legitimate traffic as an attack. If the problem stems from a custom attack object, you can simply remove it from its custom attack object group. However, you cannot remove a predefined attack object from a predefined attack object group. In that case, the best course of action is to disable the object.

Note that a predefined attack object is disabled only within the root system or virtual system (vsys) in which you disable it. For example, disabling a predefined attack object in the root system does not automatically disable it in any virtual systems. Likewise, disabling an attack object in one vsys does not affect that object in any other vsys.

NOTE: Disabling attack objects does not improve throughput performance.

To disable an attack object, do either of the following:

WebUI

Objects > Attacks > Predefined: Clear the checkbox in the **Configure** column for the attack object that you want to disable.

CLI

set attack disable *attack_object_name*

To re-enable a previously disabled attack object, do either of the following:

WebUI

Objects > Attacks > Predefined: Select the checkbox in the **Configure** column for the attack object that you want to enable.

CLI

unset attack disable *attack_object_name*

Attack Actions

When the security device detects an attack, it performs the action that you specify for the attack group containing the object that matches the attack. The seven actions are as follows, from most to least severe:

- **Close** (severs connection and sends RST to client and server)

NOTE: The client is always the initiator of a session; that is, the source address in a policy. The server is always the responder, or the destination address.

Use this option for TCP connections. The security device drops the connection and sends a TCP RST to both the client (source) and server (destination). Because the delivery of RST notifications is unreliable, by sending a RST to both client and server, there is a greater chance that at least one gets the RST and closes the session.

- **Close Server** (severs connection and sends RST to server)

Use this option for inbound TCP connections from an untrusted client to a protected server. If the client tries to launch an attack, the security device drops the connection and sends a TCP RST only to the server for it to clear its resources while the client is left hanging.

- **Close Client** (severs connection and sends RST to client)

  Use this option for outbound TCP connections from a protected client to an untrusted server. If, for example, the server sends a malicious URL string, the security device drops the connection and sends a RST only to the client for it to clear its resources while the server is left hanging.

- **Drop** (severs connection without sending anyone a RST)

  Use this option for UDP or other non-TCP connections, such as DNS. The security device drops all packets in a session, but does not send a TCP RST.

- **Drop Packet** (drops a particular packet, but does not sever connection)

  This option drops the packet in which an attack signature or protocol anomaly occurs but does not terminate the session itself. Use this option to drop malformed packets without disrupting the entire session. For example, if the security device detects an attack signature or protocol anomaly from an AOL proxy, dropping everything would disrupt all AOL service. Instead, dropping just the packet stops the problem packet without stopping the flow of all the other packets.

- **Ignore** (after detecting an attack signature or anomaly, the security device makes a log entry and stops checking—or ignores—the remainder of the connection)

  If the security device detects an attack signature or protocol anomaly, it makes an event log entry but does not sever the session itself. Use this option to tweak false positives during the initial setup phase of your Deep Inspection (DI) implementation. Also, use this option when a service uses a standard port number for nonstandard protocol activities; for example, Yahoo Messenger uses port 25 (SMTP port) for non-SMTP traffic. The security device logs the occurrence once per session (so that it does not fill the log with false positives), but takes no action.

- **None** (no action)

  It is useful when first identifying attack types during the initial setup phase of your DI implementation. When the security device detects an attack signature or protocol anomaly, it makes an entry in the event log but takes no action on the traffic itself. The security device continues to check subsequent traffic in that session and make log entries if it detects other attack signatures and anomalies.

You can create a policy referencing multiple attack object groups, each group having a different action. If the security device simultaneously detects multiple attacks that belong to different attack object groups, it applies the most severe action specified by one of those groups.

Example: Attack Actions—Close Server, Close, Close Client

In this example, there are three zones: Trust, Untrust, and DMZ. You have finished analyzing attacks and have concluded you need the following three policies:

- **Policy ID 1:** Permit HTTP, HTTPS, PING, and FTP-GET traffic from any address in the Untrust zone to the webservers (websrv1 and websrv2) in the DMZ.

  Attack Settings for Policy ID 1:

  - CRITICAL:HTTP:ANOM, CRITICAL:HTTP:SIGS
  - HIGH:HTTP:ANOM, HIGH:HTTP:SIGS
  - MEDIUM:HTTP:ANOM, MEDIUM:HTTP:SIGS
  - CRITICAL:FTP:SIGS
  - Action for all attack object groups: Close Server
  - Logging: Enabled (default setting)

  You choose to drop the connection and send a TCP RST notification only to the protected webservers so they can terminate sessions and clear resources. You anticipate attacks coming from the Untrust zone.

- **Policy ID 2:** Permit HTTP, HTTPS, PING, and FTP traffic from any address in the Trust zone to the webservers (websrv1 and websrv2) in the DMZ

  Attack Settings for Policy ID 2:

  - CRITICAL:HTTP:ANOM, CRITICAL:HTTP:SIGS
  - HIGH:HTTP:ANOM, HIGH:HTTP:SIGS
  - MEDIUM:HTTP:ANOM, MEDIUM:HTTP:SIGS
  - CRITICAL:FTP:SIGS
  - Action for all attack object groups: Close
  - Logging: Enabled (default setting)

  You choose to drop the connection and send a TCP RST notification to both the protected clients and servers so they both can terminate their sessions and clear their resources regardless of the severity level of the attack.

- **Policy ID 3:** Permit FTP-GET, HTTP, HTTPS, PING traffic from any address in the Trust zone to any address in the Untrust zone

  Attack Settings for Policy ID 3:

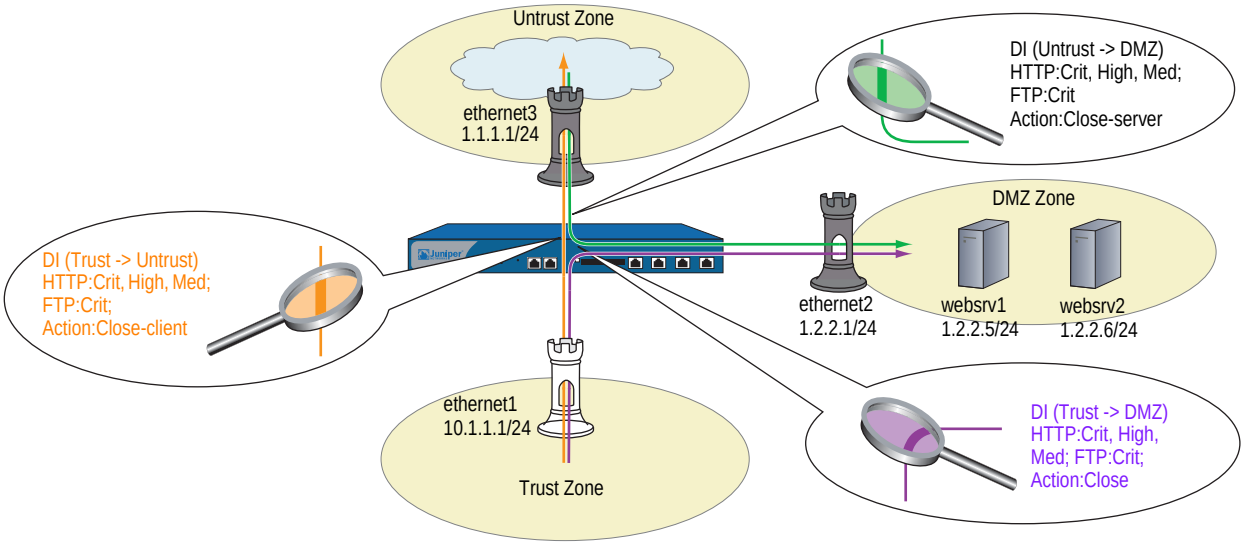
  - CRITICAL:HTTP:ANOM, CRITICAL:HTTP:SIGS
  - HIGH:HTTP:ANOM, HIGH:HTTP:SIGS

- MEDIUM:HTTP:ANOM, MEDIUM:HTTP:SIGS
- CRITICAL:FTP:SIGS
- Action for all attack object groups: Close Client
- Logging: Enabled (default setting)

You choose to drop the connection and send a TCP RST notification to the protected clients so they both can terminate their sessions and clear their resources. In this case, you anticipate an attack coming from an untrusted HTTP or FTP server.

Although the policies permit HTTP, HTTPS, Ping, and FTP-Get or FTP, the security device activates DI only for HTTP and FTP traffic. All zones are in the trust-vr routing domain.

Figure 51: DI Attack Actions



WebUI

1. **Interfaces**

   Network > Interfaces > Edit (for ethernet1): Enter the following, then click **Apply**:

   Zone Name: Trust
   Static IP: (select this option when present)
   IP Address/Netmask: 10.1.1.1/24

   Enter the following, then click **OK**:

   Interface Mode: NAT
   Service Options:
   Management Services: (select all)
   Other services: Ping

   Network > Interfaces > Edit (for ethernet3): Enter the following, then click **OK**:

Zone Name: Untrust
Static IP: (select this option when present)
IP Address/Netmask: 1.1.1.1/24

Network > Interfaces > Edit (for ethernet2): Enter the following, then click **OK**:

Zone Name: DMZ
Static IP: (select this option when present)
IP Address/Netmask: 1.2.2.1/24

2. **Addresses**
   Objects > Addresses > List > New: Enter the following, then click **OK**:

   Address Name: websrv1
   IP Address/Domain Name:
   IP/Netmask: (select), 1.2.2.5/32
   Zone: DMZ

   Objects > Addresses > List > New: Enter the following, then click **OK**:

   Address Name: websrv2
   IP Address/Domain Name:
   IP/Netmask: (select), 1.2.2.6/32
   Zone: DMZ

3. **Route**
   Network > Routing > Routing Entries > trust-vr New: Enter the following, then click **OK**:

   Network Address/Netmask: 0.0.0.0/0
   Gateway: (select)
   Interface: ethernet3
   Gateway IP Address: 1.1.1.250

4. **Policy ID 1**
   Policies > (From: Untrust, To: DMZ) New: Enter the following, then click **OK**:

   Source Address:
   Address Book Entry: (select), Any
   Destination Address:
   Address Book Entry: (select), websrv1

   > Click **Multiple**, select **websrv2**, then click **OK** to return to the basic policy configuration page.

   Service: HTTP

   > Click **Multiple**, select **FTP-GET**, **HTTPS**, **PING**, then click **OK** to return to the basic policy configuration page.

   Action: Permit

   > Click **Deep Inspection**, enter the following, click **Add** to enter each attack object group, then click **OK** to return to the basic policy configuration page:

   Group: CRITICAL:HTTP:ANOM

Action: Close Server
Log: (select)
Group: CRITICAL:HTTP:SIGS
Action: Close Server
Log: (select)
Group: HIGH:HTTP:ANOM
Action: Close Server
Log: (select)
Group: HIGH:HTTP:SIGS
Action: Close Server
Log: (select)
Group: MEDIUM:HTTP:ANOM
Action: Close Server
Log: (select)
Group: MEDIUM:HTTP:SIGS
Action: Close Server
Log: (select)
Group: CRITICAL:FTP:SIGS
Action: Close Server
Log: (select)

5. **Policy ID 2**

Policies > (From: Trust, To: DMZ) New: Enter the following, then click **OK**:

Source Address:
Address Book Entry: (select), Any
Destination Address:
Address Book Entry: (select), websrv1

> Click **Multiple**, select **websrv2**, then click **OK** to return to the basic policy configuration page.

Service: HTTP

> Click **Multiple**, select **FTP-GET**, **HTTPS**, **PING**, then click **OK** to return to the basic policy configuration page.

Action: Permit

> Click **Deep Inspection**, enter the following, click **Add** to enter each attack object group, then click **OK** to return to the basic policy configuration page:

Group: CRITICAL:HTTP:ANOM
Action: Close
Log: (select)
Group: CRITICAL:HTTP:SIGS
Action: Close
Log: (select)
Group: HIGH:HTTP:ANOM
Action: Close
Log: (select)
Group: HIGH:HTTP:SIGS
Action: Close
Log: (select)
Group: MEDIUM:HTTP:ANOM
Action: Close
Log: (select)

Group: MEDIUM:HTTP:SIGS
Action: Close
Log: (select)
Group: CRITICAL:FTP:SIGS
Action: Close
Log: (select)

6. **Policy ID 3**
   Policies > (From: Trust, To: Untrust) New: Enter the following, then click **OK**:

   Source Address:
   Address Book Entry: (select), Any
   Destination Address:
   Address Book Entry: (select), Any
   Service: HTTP

   > Click **Multiple**, select **FTP-GET**, **HTTPS**, **PING**, then click **OK** to return to the basic policy configuration page.

   Action: Permit

   > Click **Deep Inspection**, enter the following, click **Add** to enter each attack object group, then click **OK** to return to the basic policy configuration page:

   Group: CRITICAL:HTTP:ANOM
   Action: Close Client
   Log: (select)
   Group: CRITICAL:HTTP:SIGS
   Action: Close Client
   Log: (select)
   Group: HIGH:HTTP:ANOM
   Action: Close Client
   Log: (select)
   Group: HIGH:HTTP:SIGS
   Action: Close Client
   Log: (select)
   Group: MEDIUM:HTTP:ANOM
   Action: Close Client
   Log: (select)
   Group: MEDIUM:HTTP:SIGS
   Action: Close Client
   Log: (select)
   Group: CRITICAL:FTP:SIGS
   Action: Close Client
   Log: (select)

CLI

1. **Interfaces**

```
set interface ethernet1 zone trust
set interface ethernet1 ip 10.1.1.1/24
set interface ethernet1 manage
set interface ethernet3 zone untrust
set interface ethernet3 ip 1.1.1.1/24
set interface ethernet2 zone dmz
set interface ethernet2 ip 2.1.1.1/24
```

2. **Addresses**

```
set address dmz websrv1 1.2.2.5/32
set address dmz websrv2 1.2.2.6/32
```

3. **Route**

```
set vrouter trust-vr route 0.0.0.0/0 interface ethernet3 gateway 1.1.1.250
```

4. **Policy ID 1**

```
set policy id 1 from untrust to dmz any websrv1 http permit attack
    CRITICAL:HTTP:ANOM action close-server
set policy id 1
device(policy:1)-> set dst-address websrv2
device(policy:1)-> set service ftp-get
device(policy:1)-> set service https
device(policy:1)-> set service ping
device(policy:1)-> set attack CRITICAL:HTTP:SIGS action close-server
device(policy:1)-> set attack HIGH:HTTP:ANOM action close-server
device(policy:1)-> set attack HIGH:HTTP:SIGS action close-server
device(policy:1)-> set attack MEDIUM:HTTP:ANOM action close-server
device(policy:1)-> set attack MEDIUM:HTTP:SIGS action close-server
device(policy:1)-> set attack CRITICAL:FTP:SIGS action close-server
device(policy:1)-> exit
```

5. **Policy ID 2**

```
set policy id 2 from trust to dmz any websrv1 http permit attack
    CRITICAL:HTTP:ANOM action close
set policy id 2
device(policy:2)-> set dst-address websrv2
device(policy:2)-> set service ftp
device(policy:2)-> set service https
device(policy:2)-> set service ping
device(policy:2)-> set attack CRITICAL:HTTP:SIGS action close
device(policy:2)-> set attack HIGH:HTTP:ANOM action close
device(policy:2)-> set attack HIGH:HTTP:SIGS action close
device(policy:2)-> set attack MEDIUM:HTTP:ANOM action close
device(policy:2)-> set attack MEDIUM:HTTP:SIGS action close
device(policy:2)-> set attack CRITICAL:FTP:SIGS action close
device(policy:2)-> exit
```

6. **Policy ID 3**

```
set policy id 3 from trust to untrust any any http permit attack
    CRITICAL:HTTP:ANOM action close-client
set policy id 3
device(policy:3)-> set service ftp-get
device(policy:3)-> set service https
device(policy:3)-> set service ping
device(policy:3)-> set attack CRITICAL:HTTP:SIGS action close-client
```

```
device(policy:3)-> set attack HIGH:HTTP:ANOM action close-client
device(policy:3)-> set attack HIGH:HTTP:SIGS action close-client
device(policy:3)-> set attack MEDIUM:HTTP:ANOM action close-client
device(policy:3)-> set attack MEDIUM:HTTP:SIGS action close-client
device(policy:3)-> set attack CRITICAL:FTP:SIGS action close-client
device(policy:3)-> exit
save
```

Brute Force Attack Actions

A typical brute force attack is accomplished by sending lots of traffic with varying source ports or IP in an attempt to obtain network access. In order to effectively prevent future attempts, ScreenOS allows you to associate an IP action for each attack group in a policy.

Brute force attack is detected based on the threshold values set for the DI supported protocols. For example,

set di service *protocol-name value*

Apart from a DI action, brute force attack actions are configured with the **IP action** command for a configured amount of time for a specified target. If your security device detects a brute force attack, then select one of the following actions to perform:

- Notify: The security device logs the event but does not take any action against further traffic matching the target definition for the period of time specified in the timeout setting.
- Block: The security device logs the event and drops all further traffic matching the target definition for the period of time specified in the timeout setting.
- Close: The security device logs the event and drops all further traffic matching the target definition for the period of time specified in the timeout setting, and sends a Reset (RST) for TCP traffic to the source and destination addresses.

Brute Force Attack Objects

Table 12 lists the brute force attack objects in ScreenOS 5.4 and the threshold parameters that can be used with the IP actions.

Table 12: Brute Force Attack Objects

Brute Force Attack Name	Parameter
HTTP Brute Force Login Attempt	failed_logins
HTTP Brute Search Attempt	brute_search
IMAP Brute Force Login Attempt	failed_logins
LDAP Brute Force Login Attempt	failed_logins
MS-RPC IsSystemActive request flood	Not configurable—32 attempts
MS-SQL Login Brute Force	Not configurable—4 attempts
POP3 Brute Force Login Attempt	failed_login
RADIUS Brute Force Authentication Attempt	failed_auth
SMB Brute Force Directory Create/Delete	Not configurable—200 attempts
SMB Brute Force Login Attempt	failed_login
FTP Brute Force Login Attempt	failed_login
Telnet Brute Force Login Attempt	failed_login
VNC Brute Force Login Attempt	failed_login

Brute Force Attack Target

The target option specifies a set of elements that must match for the security device to consider a packet part of a brute force attack. The specified set of elements in an IP packet arriving during a specified timeout period must match that in the packet that the security device detected as part of a brute force attack for the subsequent packet to be considered part of the same attack. The default target definition is Serv. You can select any of the following target definitions shown in Table 13.

Table 13: Target Options

Target option	Matching elements
Serv	source IP, destination IP, destination port, and protocol
Src-IP	source IP address
Zone-Serv	source security zone, destination IP, destination port number, and protocol
Dst-IP	destination IP address
Zone	source security zone (The security zone to which the ingress interface is bound; that is, the source security zone from which the attacking packets originate)

Brute Force Attack Timeout

Timeout is a period of time following brute force attack detection during which the security device performs an IP action on packets matching specified target parameters. The default timeout is 60 seconds.

Example 1

In this example, you configure an IP action along with the existing DI action for each group in a policy. The following CLI commands block brute force attack object—HTTP Brute Force Login Attempt or HTTP Brute Force Search for 45 seconds. All other attacks in the HIGH:HTTP:ANOM attack group are configured with a DI action of **close**.

CLI

```
device>get attack group HIGH:HTTP:ANOM
GROUP "HIGH:HTTP:ANOM" is pre-defined. It has the following members
ID     Name
1674 HTTP:INVALID:INVLD-AUTH-CHAR
1675 HTTP:INVALID:INVLD-AUTH-LEN
1711 HTTP:OVERFLOW:HEADER
1713 HTTP:OVERFLOW:INV-CHUNK-LEN
1717 HTTP:OVERFLOW:AUTH-OVFLW
5394 HTTP:EXPLOIT:BRUTE-FORCE
5395 HTTP:EXPLOIT:BRUTE-SEARCH
device> set policy id 1 from Untrust to DMZ Any Any Any permit attack
    MEDIUM:HTTP:ANOM action none
device> set policy id 1
device(policy:1)-> set attack HIGH:HTTP:ANOM action close ip-action block
    target dst-ip timeout 45
```

If the configured attack group does not have any brute force attack protocol anomalies, IP action is not enforced.

Example 2

In this example, you associate an IP action for each attack group for a configured amount of time from a specified host.

```
set policy id 1 from trust to untrust any any any permit attack POP3 BRUTE FORCE
    Login Attempt action close ip-action notify target serv timeout 60
```

Example 3

In this example, the default threshold value of FTP brute force login attempt is 8 attempts per minute. If a user at IP address 192.168.2.2 is launching a FTP brute force login attempt to FTP server at 10.150.50.5 in order to figure out a user account name and password, the attempt is detected when the attacker makes 8 FTP login attempts within a minute.

If an IP action is configured to “Block” for 120 seconds for target of “serv”, any traffic coming from 192.168.2.2 (src IP) to 10.150.50.5 (dst IP) over TCP (protocol) port 21 (dst port) is blocked for 120 seconds.

Note that some IP action targets may affect traffic matching another policy.

Attack Logging

You can enable the logging of detected attacks per attack group per policy. In other words, within the same policy, you can apply multiple attack groups and selectively enable the logging of detected attacks for just some of them.

By default, logging is enabled. You might want to disable logging for attacks that are lower priority for you and about which you do not give much attention. Disabling logging for such attacks helps prevent the event log from becoming cluttered with entries that you do not plan to look at anyway.

Example: Disabling Logging per Attack Group

In this example, you reference the following five attack groups in a policy and enable logging only for the first two:

- HIGH:IMAP:ANOM
- HIGH:IMAP:SIGS
- MEDIUM:IMAP:ANOM
- LOW:IMAP:ANOM
- INFO:IMAP:ANOM

The policy applies to IMAP traffic from all hosts in the Trust zone to a mail server named "mail1" in the DMZ. If any of the predefined IMAP attack objects in the above five groups match an attack, the security device closes the connection. However, it only creates log entries for detected attacks matching attack objects in the first two groups.

WebUI

1. **Address**
   Objects > Addresses > List > New: Enter the following, then click **OK**:

   Address Name: mail1
   IP Address/Domain Name:
   IP/Netmask: (select), 1.2.2.10/32
   Zone: DMZ

2. Policy

Policies > (From: Trust, To: DMZ) New: Enter the following, then click **OK**:

Source Address:
Address Book Entry: (select), Any
Destination Address:
Address Book Entry: (select), mail1
Service: IMAP
Action: Permit

> Click **Deep Inspection**, enter the following, click **Add** to enter each attack object group, then click **OK** to return to the basic policy configuration page:

Group: HIGH:IMAP:ANOM
Action: Close
Log: (select)

Group: HIGH:IMAP:SIGS
Action: Close
Log: (select)

Group: MEDIUM:IMAP:ANOM
Action: Close
Log: (clear)

Group: LOW:IMAP:ANOM
Action: Close
Log: (clear)

Group: INFO:IMAP:ANOM
Action: Close
Log: (clear)

CLI

1. Address

```
set address dmz mail1 1.2.2.10/32
```

2. Policy

```
device-> set policy id 1 from trust to dmz any mail1 imap permit attack
    HIGH:IMAP:ANOM action close
device-> set policy id 1
device(policy:1)-> set attack HIGH:IMAP:SIGS action close
device(policy:1)-> set attack MEDIUM:IMAP:ANOM action close
device(policy:1)-> unset attack MEDIUM:IMAP:ANOM logging
device(policy:1)-> set attack LOW:IMAP:ANOM action close
device(policy:1)-> unset attack LOW:IMAP:ANOM logging
device(policy:1)-> set attack INFO:IMAP:ANOM action close
device(policy:1)-> unset attack INFO:IMAP:ANOM logging
device(policy:1)-> exit
device-> save
```

Mapping Custom Services to Applications

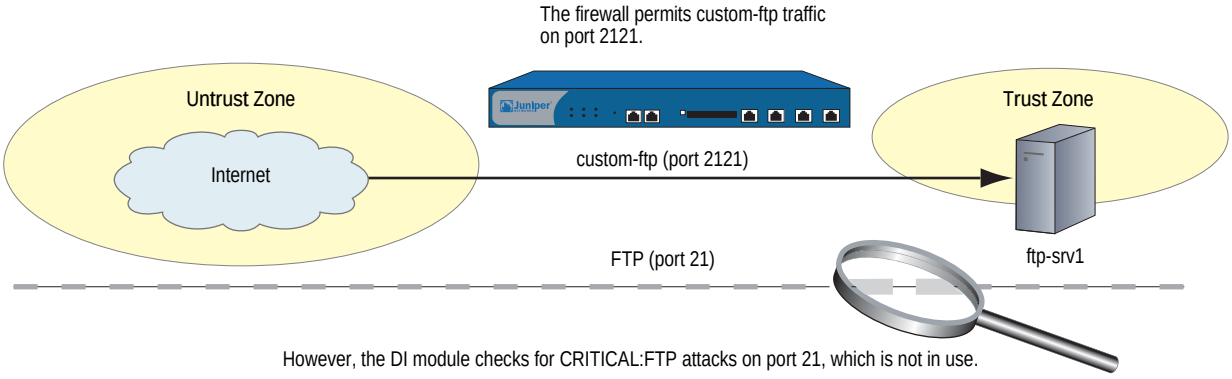
When using a custom service in a policy with a Deep Inspection (DI) component, you must explicitly specify the application that is running on that service so that the DI module can function properly. For example, if you create a custom service for FTP running on a nonstandard port number such as 2121 (see Figure 52), you can reference that custom service in a policy as follows:

```
set service ftp-custom protocol tcp src-port 0-65535 dst-port 2121-2121
set policy id 1 from untrust to trust any ftp-srv1 custom-ftp permit
```

However, if you add a DI component to a policy that references a custom service, the DI module cannot recognize the application because it is using a nonstandard port number.

```
set policy id 1 from untrust to trust any ftp-srv1 custom-ftp permit attack
    CRITICAL:FTP:SIGS action close-server
```

Figure 52: Mapping Custom Service



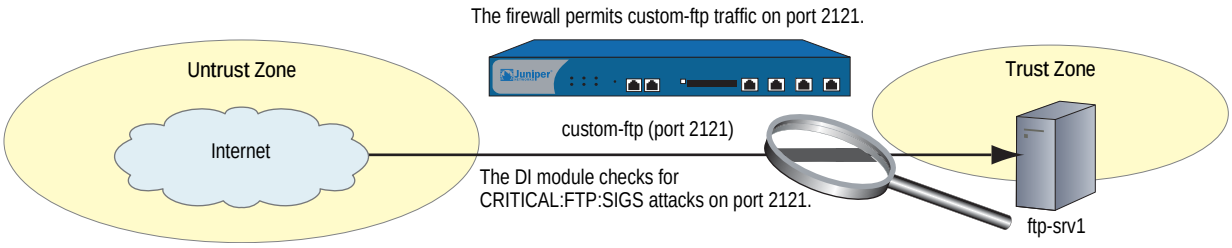
To avoid this problem, you must inform the DI module that the FTP application is running on port 2121 (see Figure 53). Essentially, you must map the protocol in the Application Layer to a specific port number in the Transport Layer. You can do this binding at the policy level:

```
set policy id 1 application ftp
```

When you map the FTP application to the custom service "custom-ftp" and configure DI to examine FTP traffic for the attacks defined in the CRITICAL:FTP:SIGS attack object group in a policy that references custom-ftp, the DI module perform its inspection on port 2121.

```
set policy id 1 from untrust to trust any ftp-srv1 custom-ftp permit attack
    CRITICAL:FTP:SIGS action close-server
set policy id 1 application ftp
```

Figure 53: Mapping Custom Service to Attack Object Group



Example: Mapping an Application to a Custom Service

In this example, you define a custom service named "HTTP1" that uses destination port 8080. You map the HTTP application to the custom service for a policy permitting HTTP1 traffic from any address in the Untrust zone to a webserver named "server1" in the DMZ zone. You then apply Deep Inspection (DI) to the permitted HTTP traffic running on port 8080. The DI settings for this policy are as follows:

- Attack Object Groups:
  - CRITICAL:HTTP:ANOM, CRITICAL:HTTP:SIGS
  - HIGH:HTTP:ANOM, HIGH:HTTP:SIGS
  - MEDIUM:HTTP:ANOM, MEDIUM:HTTP:SIGS
- Action for all attack object groups: Close Server
- Logging: Enabled (default setting)

WebUI

1. **Custom Service**
   Objects > Services > Custom > New: Enter the following, then click **OK**:

   Service Name: HTTP1
   Transport Protocol: TCP (select)
   Source Port Low: 0
   Source Port High: 65535
   Destination Port Low: 8080
   Destination Port High: 8080

2. **Address**
   Objects > Addresses > List > New: Enter the following, then click **OK**:

   Address Name: server1
   IP Address/Domain Name:
   IP/Netmask: 1.2.2.5/32
   Zone: DMZ

3. **Policy**
   Policies > (From: Untrust, To: DMZ) New: Enter the following, then click **OK**:

   Source Address:
   Address Book Entry: (select), Any
   Destination Address:
   Address Book Entry: (select), server1
   Service: HTTP1
   Application: HTTP
   Action: Permit

   > Click **Deep Inspection**, enter the following, click **Add** to enter each attack object group, then click **OK** to return to the basic policy configuration page:

   Group: CRITICAL:HTTP:ANOM
   Action: Close Server
   Log: (select)

   Group: CRITICAL:HTTP:SIGS
   Action: Close Client
   Log: (select)

   Group: HIGH:HTTP:ANOM
   Action: Close Client
   Log: (select)

   Group: HIGH:HTTP:SIGS
   Action: Close Client
   Log: (select)

   Group: MEDIUM:HTTP:ANOM
   Action: Close Client
   Log: (select)

   Group: MEDIUM:HTTP:SIGS
   Action: Close Client
   Log: (select)

CLI

1. **Custom Service**
   set service HTTP1 protocol tcp src-port 0-65535 dst-port 8080-8080

2. **Address**
   set address dmz server1 1.2.2.5/32

3. **Policy**
   device-> set policy id 1 from untrust to dmz any server1 HTTP1 permit attack CRITICAL:HTTP:ANOM action close-server
   device-> set policy id 1
   device(policy:1)-> set attack CRITICAL:HTTP:SIGS action close-server
   device(policy:1)-> set attack HIGH:HTTP:ANOM action close-server
   device(policy:1)-> set attack HIGH:HTTP:SIGS action close-server
   device(policy:1)-> set attack MEDIUM:HTTP:ANOM action close-server
   device(policy:1)-> set attack MEDIUM:HTTP:SIGS action close-server
   device(policy:1)-> exit
   device-> set policy id 1 application http
   save

Example: Application-to-Service Mapping for HTTP Attacks

Some known HTTP attacks use TCP port 8000. At the time of this writing, there are currently two such attacks in the Deep Inspection (DI) attack object database:

- 3656, App: HP Web JetAdmin Framework Infoleak

  DOS:NETDEV:WEBJET-FW-INFOLEAK (in the attack object group MEDIUM:HTTP:SIGS)

- 3638, App: HP Web JetAdmin WriteToFile Vulnerability,

  DOS:NETDEV:WEBJET-WRITETOFILE (in the attack object group CRITICAL:HTTP:SIGS)

However, by default, ScreenOS considers only TCP traffic on port 80 to be HTTP. Therefore, if the security device receives TCP traffic using port 8000, it does not recognize it as HTTP. Consequently the DI engine does not scan such HTTP traffic for these attacks and cannot detect them if they occur—unless you map HTTP as an application to a custom service using port 8000.

In this example, you associate traffic using the nonstandard port of 8000 with HTTP to detect the above attacks.

NOTE: In general, if you are running some services using nonstandard port numbers in your network and you want the DI engine to scan that traffic, you must associate the nonstandard port number to the service.

WebUI

1. **Custom Service**
   Objects > Services > Custom > New: Enter the following, then click **OK**:

   Service Name: HTTP2
   Transport Protocol: TCP (select)
   Source Port Low: 0
   Source Port High: 65535
   Destination Port Low: 8000
   Destination Port High: 8000

2. **Policy**
   Policies > (From: Untrust, To: DMZ) New: Enter the following, then click **OK**:

   Source Address:
   Address Book Entry: (select), Any
   Destination Address:
   Address Book Entry: (select), Any
   Service: HTTP2
   Application: HTTP
   Action: Permit

> Click **Deep Inspection**, enter the following, click **Add** to enter each attack object group, then click **OK** to return to the basic policy configuration page:

Group: CRITICAL:HTTP:SIGS
Action: Close
Log: (select)

Group: MEDIUM:HTTP:SIGS
Action: Close
Log: (select)

CLI

1. **Custom Service**
   set service HTTP2 protocol tcp src-port 0-65535 dst-port 8000-8000
2. **Policy**
   device-> set policy id 1 from untrust to dmz any any HTTP2 permit attack CRITICAL:HTTP:SIGS action close
   device-> set policy id 1
   device(policy:1)-> set attack MEDIUM:HTTP:SIGS action close
   device(policy:1)-> exit
   device-> set policy id 1 application http
   save

Customized Attack Objects and Groups

You can define new attack objects and object groups to customize the Deep Inspection (DI) application to best meet your needs. User-defined attack objects can be stateful signatures or—on the NetScreen-5000—TCP stream signatures. You can also adjust various parameters to modify predefined protocol anomaly attack objects.

User-Defined Stateful Signature Attack Objects

You can create a stateful signature attack object for FTP, HTTP, and SMTP. When creating an attack object, you perform the following steps:

- Name the attack object. (All user-defined attack objects must begin with "CS:".)
- Set the context for the DI search. (For a complete list of all the contexts that you can use when creating attack objects, see "Contexts for User-Defined Signatures" on page I)
- Define the signature. ("Regular Expressions" on page 150 examines the regular expressions that you can use when defining signatures.)
- Assign the attack object a severity level. (For information about severity levels, see "Changing Severity Levels" on page 128.)

You must then put a user-defined attack object in a user-defined attack object group for use in policies.

NOTE: A user-defined attack object group can only contain user-defined attack objects. You cannot mix predefined and user-defined attack objects in the same attack object group.

Regular Expressions

When entering the text string for a signature, you can enter an alphanumeric string of ordinary characters to search for an exact character-to-character match, or you can use regular expressions to broaden the search for possible matches to sets of characters. ScreenOS supports the following regular expressions as shown in Table 14.

Table 14: ScreenOS Supported Regular Expressions

Purpose	Meta characters	Example	Meaning
Direct binary match (octal)[1]	\O*octal_number*	\0162 Matches: 162	Exactly match this octal number: 162
Direct binary match (hexadecimal)[2]	\X*hexadecimal_number*\X	\X01 A5 00 00\X Matches: 01 A5 00 00	Exactly match these four hexadecimal numbers: 01 A5 00 00
Case-insensitive matches	\[*characters*\]	\[cat\] Matches: ■ Cat, cAt, caT ■ CAt, CaT, CAT ■ cat, cAt	Match the characters in cat regardless of the case of each character
Match any character	.	c . t Matches: ■ cat, cbt, cct, … czt ■ cAt, cBt, cCt, … cZt ■ c1t, c2t, c3t, … c9t	Match c-any character-t
Match the previous character zero or more times, instead of only once	*	a*b + c Matches: ■ bc ■ bbc ■ abc ■ aaabbbbc	Match zero, one, or multiple occurrences of a, followed by one or more occurrences of b, followed by one occurrence of c
Match the previous character one or more times	+	a + b + c Matches: ■ abc ■ aabc ■ aaabbbbc	Match one or more occurrences of a, followed by one or more occurrences of b, followed by one occurrence of c
Match the previous character zero times or one time	?	drop-?packet Matches: ■ drop-packet ■ droppacket	Match either drop-packet or droppacket
Group expressions	( )		

Purpose	Meta characters	Example	Meaning
Either the previous or the following character—typically used with ( )	|	(drop | packet) Matches: ■ drop ■ packet	Match either drop or packet
Character range	[*start-end*]	[c-f]a(d | t) Matches: ■ cad, cat ■ dad, dat ■ ead, eat ■ fad, fat	Match everything that begins with c, d, e, or f and that has the middle letter a and the last letter d or t
Negation of the following character	[^*character*]	[^0-9A-Z] Matches: a, b, c, d, e, … z	Match lowercase letters

1.Octal is a base-8 number system that uses only the digits 0–7.
2. Hexadecimal is a base-16 number system that uses digits 0–9 as usual, and then the letters A–F representing hexadecimal digits with decimal values of 10–15.

Example: User-Defined Stateful Signature Attack Objects

In this example, you have an FTP server, a webserver, and a mail server in the DMZ zone. You define the following attack objects for the use-defined signature objects as shown in Table 15.

Table 15: User-Defined Stateful Signature Attack Objects

Object Name	Usage
cs:ftp-stor	Block someone from putting files on an FTP server
cs:ftp-user-dm	Deny FTP access to the user with the login name dmartin
cs:url-index	Block HTTP packets with a defined URL in any HTTP request
cs:spammer	Block e-mail from any e-mail address at “spam.com”

You then organize them into a user-defined attack object group named “DMZ DI”, which you reference in a policy permitting traffic from the Untrust zone to the servers in the DMZ zone.

WebUI

1. **Attack Object 1: ftp-stor**
   Objects > Attacks > Custom > New: Enter the following, then click **OK**:

   Attack Name: cs:ftp-stor
   Attack Context: FTP Command
   Attack Severity: Medium
   Attack Pattern: STOR

2. **Attack Object 2: ftp-user-dm**
   Objects > Attacks > Custom > New: Enter the following, then click **OK**:

   Attack Name: cs:ftp-user-dm
   Attack Context: FTP User Name
   Attack Severity: Low

Attack Pattern: dmartin

3. **Attack Object 3: url-index**
   Objects > Attacks > Custom > New: Enter the following, then click **OK**:

   Attack Name: cs:url-index
   Attack Context: HTTP URL Parsed
   Attack Severity: High
   Attack Pattern: .*index.html.*

4. **Attack Object 4: spammer**
   Objects > Attacks > Custom > New: Enter the following, then click **OK**:

   Attack Name: cs:spammer
   Attack Context: SMTP From
   Attack Severity: Info
   Attack Pattern: .*@spam.com

5. **Attack Object Group**
   Objects > Attacks > Custom Groups > New: Enter the following group name, move the following custom attack objects, then click **OK**:

   Group Name: CS:DMZ DI

   Select **cs:ftp-stor** and use the < < button to move the address from the Selected Members column to the Selected Members column.

   Select **cs:ftp-user-dm** and use the < < button to move the address from the Available Members column to the Selected Members column.

   Select **cs:url-index** and use the < < button to move the address from the Available Members column to the Selected Members column.

   Select **cs:spammer** and use the < < button to move the address from the Available Members column to the Selected Members column.

6. **Policy**
   Policies > (From: Untrust, To: DMZ) New: Enter the following, then click **OK**:

   Source Address:
   Address Book Entry: (select), Any
   Destination Address:
   Address Book Entry: (select), Any
   Service: HTTP

   > Click **Multiple**, select **FTP**, then click **OK** to return to the basic policy configuration page.

   Action: Permit

> Click **Deep Inspection**, enter the following, click **Add** to enter each attack object group, then click **OK** to return to the basic policy configuration page:

Group: CS:DMZ DI
Action: Close Server
Log: (select)

CLI

1. **Attack Object 1: ftp-stor**
   set attack cs:ftp-stor ftp-command STOR severity medium
2. **Attack Object 2: ftp-user-dm**
   set attack cs:ftp-user-dm ftp-username dmartin severity low
3. **Attack Object 3: url-index**
   set attack cs:url-index http-url-parsed index.html severity high
4. **Attack Object 4: url-index**
   set attack cs:spammer smtp-from .*@spam.com severity info
5. **Attack Object Group**
   set attack group "CS:DMZ DI"
   set attack group "CS:DMZ DI" add cs:ftp-stor
   set attack group "CS:DMZ DI" add cs:ftp-user-dm
   set attack group "CS:DMZ DI" add cs:url-index
   set attack group "CS:DMZ DI" add cs:spammer
6. **Policy**
   set policy id 1 from untrust to dmz any any http permit attack "CS:DMZ DI" action close-server
   set policy id 1
   device(policy:1)-> set service ftp
   device(policy:1)-> exit
   save

TCP Stream Signature Attack Objects

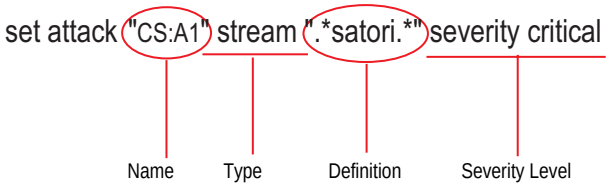
The stateful signatures are context-based within specific applications, such as an FTP username or an SMTP header field. TCP stream signatures look for patterns anywhere in any kind of TCP traffic regardless of the application protocol in use.

NOTE: You can define TCP stream signatures on NetScreen-5000 series systems only.

Because there are no predefined TCP stream signature attack objects, you must define them. When creating a signature attack object, you define the following components:

- Attack object name (All user-defined attack objects must begin with "CS:".)
- Object type ("stream")
- Pattern definition
- Severity level

Figure 54: Example of a TCP Stream Signature Attack Object



Example: User-Defined Stream Signature Attack Object

In this example, you define a stream signature object ".*satori.*". You name it "CS:A1" and define its severity level as critical. Because a policy can reference only attack object groups, you create a group named "CS:Gr1", and then add this object to it. Finally, you define a policy that references CS:Gr1 and that instructs the security device to sever the connection and send TCP RST to the client if the pattern appears in any traffic to which the policy applies.

WebUI

1. **Stream Signature Attack Object**
   Objects > Attacks > Custom > New: Enter the following, then click **OK**:

   Attack Name: CS:A1
   Attack Context: Stream
   Attack Severity: Critical
   Attack Pattern: .*satori.*

2. **Stream Signature Attack Object Group**
   Objects > Attacks > Custom Groups > New: Enter the following, then click **OK**:

   Group Name: CS:Gr1

   Select **CS:A1** in the Available Members column and then click < < to move it to the Selected Members column.

3. **Policy**
   Policies > (From: Trust, To: Untrust) New: Enter the following, then click **OK**:

   Source Address:
   Address Book Entry: (select), Any
   Destination Address:
   Address Book Entry: (select), Any
   Service: ANY
   Action: Permit

   > Click **Deep Inspection**, enter the following, click **Add** to enter each attack object group; and then click **OK** to return to the basic policy configuration page:

   Group: CS:Gr1
   Action: Close Client
   Log: (select)

CLI

1. **Stream Signature Attack Object**
   set attack "CS:A1" stream ".*satori.*" severity critical
2. **Stream Signature Attack Group**
   set attack group "CS:Gr1"
   set attack group "CS:Gr1" add "CS:A1"
3. **Policy**
   set policy from trust to untrust any any any permit attack CS:Gr1 action close-client save

Configurable Protocol Anomaly Parameters

You can modify certain parameters of a protocol anomaly attack object. Although Juniper defines protocol anomaly attack objects to find deviations from protocol standards defined in RFCs and common RFC extensions, not all implementations adhere to these standards. If you find that the application of a certain protocol anomaly attack object is producing numerous false positives, you can modify its parameters to better match the accepted use of that protocol in your network.

NOTE: For a complete list of all configurable parameters, see the **di** command in *ScreenOS CLI Reference Guide: IPv4 Command Descriptions*.

Example: Modifying Parameters

In this example, you set higher values for the following parameters to reduce the number of false positives that occurred with the default settings:

Protocol Parameter	Default	New
SMB—Maximum number of login failures per minute	4 failures	8 failures
Gnutella—Maximum number of time-to-live (TTL) hops	8 hops	10 hops

For the following parameters, you set lower values to detect anomalous behavior that the security device missed with the default settings:

Protocol Parameter	Default	New
AOL Instant Messenger (AIM)—Maximum OSCAR File Transfer (OFT) file name length. OSCAR = Open System for Communication in Real-time, the protocol that AIM clients use.	10,000 bytes	5,000 bytes
AOL Instant Messenger—Maximum length of a FLAP frame (FLAP header, which is always 6 bytes, plus data). OSCAR makes use of a the FLAP protocol to make connections and open channels between AIM clients.	10,000 bytes	5,000 bytes

WebUI

NOTE: You must use the CLI to modify protocol anomaly parameters.

CLI

```
set di service smb failed_logins 8
set di service gnutella max_ttl_hops 10
set di service aim max_flap_length 5000
set di service aim max_oft_frame 5000
save
```

Negation

Typically, you use attack objects to match patterns that are indicative of malicious or anomalous activity. However, you can also use them to match patterns indicative of benign or legitimate activity. WIth this approach, something is amiss only if a type of traffic does *not* match a particular pattern. To use attack objects in this way, you apply the concept of negation.

A useful application of attack object negation would be to block all login attempts other than those with the correct username and password. It would be difficult to define all invalid usernames and passwords, but quite easy to define the correct ones and then apply negation to reverse what the security device considers an attack; that is, everything except the specified attack object.

Example: Attack Object Negation

In this example (see Figure 55), you define two attack objects: one specifying the correct username required to log in to an FTP server, and another the correct password. You then apply negation to both attack objects, so that the security device blocks any login attempt to that server that uses any other username or password than those defined in the attack objects.

The example uses the following settings:

- The correct username and password are *admin1* and *pass1*.
- The FTP server is at 1.2.2.5 in the DMZ zone. Its address name is *ftp1*.
- You apply DI on FTP traffic to the server from all hosts in the Untrust and Trust zones.
- All security zones are in the trust-vr routing domain.

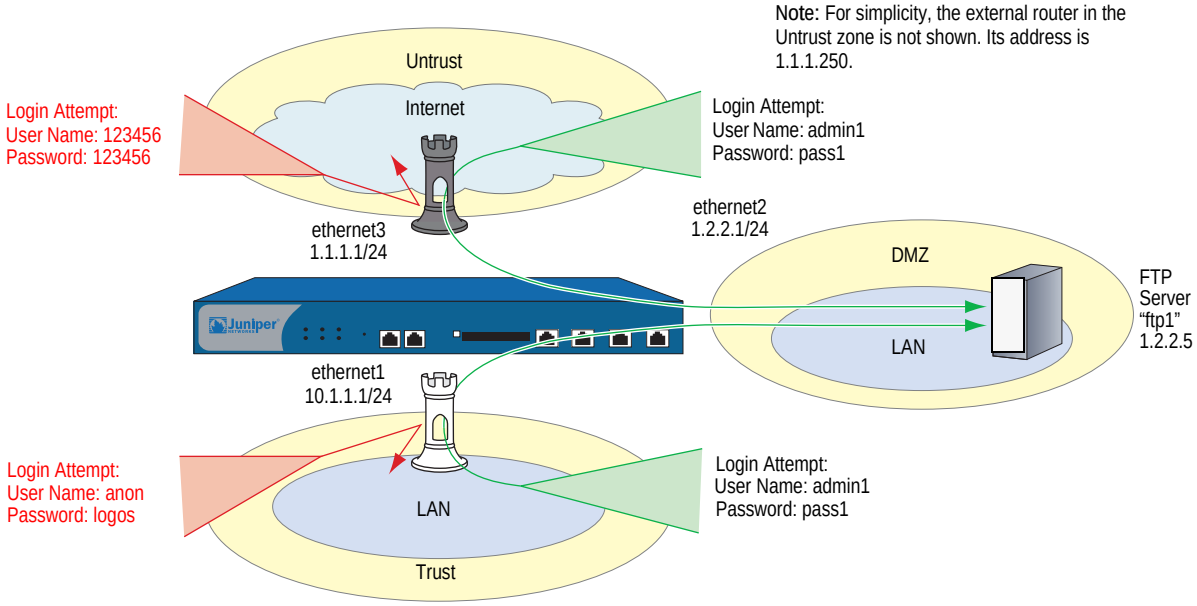
You create the following two attack objects:

- Attack Object #1:
  - Name: CS:FTP1_USR_OK
  - Negation: enabled
  - Context: ftp-username
  - Pattern: admin1
  - Severity: high

- Attack Object #2:
  - Name: CS:FTP1_PASS_OK
  - Negation: enabled
  - Context: ftp-password
  - Pattern: pass1
  - Severity: high

You then put both objects into an attack object group named *CS:FTP1_LOGIN* and reference that attack object group in two policies permitting FTP traffic from the Trust and Untrust zones to ftp1 in the DMZ.

Figure 55: Attack Object Negation



WebUI

1. **Interfaces**

   Network > Interfaces > Edit (for ethernet1): Enter the following, then click **Apply**:

   Zone Name: Trust
   Static IP: (select this option when present)
   IP Address/Netmask: 10.1.1.1/24

   Select the following, then click **OK**:

   Interface Mode: NAT (select)

NOTE: By default, any interface that you bind to the Trust zone is in NAT mode. Consequently, this option is already enabled for interfaces bound to the Trust zone.

Network > Interfaces > Edit (for ethernet2): Enter the following, then click **OK**:

Zone Name: DMZ
Static IP: (select this option when present)
IP Address/Netmask: 1.2.2.1/24

Network > Interfaces > Edit (for ethernet3): Enter the following, then click **OK**:

Zone Name: Untrust
Static IP: (select this option when present)
IP Address/Netmask: 1.1.1.1/24

2. **Address**
   Objects > Addresses > List > New: Enter the following, then click **OK**:

   Address Name: ftp1
   IP Address/Domain Name:
   IP/Netmask: (select), 1.2.2.5/32
   Zone: DMZ

3. **Attack Object 1: CS:FTP1_USR_OK**
   Objects > Attacks > Custom > New: Enter the following, then click **OK**:

   Attack Name: CS:FTP1_USR_OK
   Attack Context: ftp-username
   Attack Severity: High
   Attack Pattern: admin1
   Pattern Negation: (select)

4. **Attack Object 2: CS:FTP1_PASS_OK**
   Objects > Attacks > Custom > New: Enter the following, then click **OK**:

   Attack Name: CS:FTP1_PASS_OK
   Attack Context: ftp-password
   Attack Severity: High
   Attack Pattern: pass1
   Pattern Negation: (select)

5. **Attack Object Group**
   Objects > Attacks > Custom Groups > New: Enter the following group name, move the following custom attack objects, then click **OK**:

   Group Name: CS:FTP1_LOGIN

   Select **CS:FTP1_USR_OK** and use the < < button to move the address from the Available Members column to the Selected Members column.

   Select **CS:FTP1_PASS_OK** and use the < < button to move the address from the Available Members column to the Selected Members column.

6. **Route**

   Network > Routing > Routing Entries > trust-vr New: Enter the following, then click **OK**:

   Network Address/Netmask: 0.0.0.0/0
   Gateway: (select)
   Interface: ethernet3
   Gateway IP Address: (select) 1.1.1.250

7. **Policies**

   Policies > (From: Untrust, To: DMZ) New: Enter the following, then click **OK**:

   Source Address:
   Address Book Entry: (select), Any
   Destination Address:
   Address Book Entry: (select), ftp1
   Service: FTP
   Action: Permit

   > Click **Deep Inspection**, enter the following, click **Add** to enter each attack object group; and then click **OK** to return to the basic policy configuration page:

   Group: CS:FTP1_LOGIN
   Action: Drop
   Log: (select)

   Policies > (From: Trust, To: DMZ) New: Enter the following, then click **OK**:

   Source Address:
   Address Book Entry: (select), Any
   Destination Address:
   Address Book Entry: (select), ftp1
   Service: FTP
   Action: Permit

   > Click **Deep Inspection**, enter the following, click **Add** to enter each attack object group, then click **OK** to return to the basic policy configuration page:

   Group: CS:FTP1_LOGIN
   Action: Drop
   Log: (select)

CLI

1. **Interfaces**
   set interface ethernet1 zone trust
   set interface ethernet1 ip 10.1.1.1/24
   set interface ethernet1 nat
   set interface ethernet2 zone dmz
   set interface ethernet2 ip 1.2.2.1/24
   set interface ethernet3 zone untrust
   set interface ethernet3 ip 1.1.1.1/24
2. **Address**
   set address dmz ftp1 1.2.2.5/32
3. **Attack Objects**
   set attack CS:FTP1_USR_OK ftp-username not admin1 severity high
   set attack CS:FTP1_PASS_OK ftp-password not pass1 severity high
   set attack group CS:FTP1_LOGIN
   set attack group CS:FTP1_LOGIN add CS:FTP1_USR_OK
   set attack group CS:FTP1_LOGIN add CS:FTP1_PASS_OK
4. **Route**
   set vrouter trust-vr route 0.0.0.0/0 interface ethernet3 gateway 1.1.1.250
5. **Policies**
   set policy from untrust to dmz any ftp1 ftp permit attack CS:FTP1_LOGIN action drop
   set policy from trust to dmz any ftp1 ftp permit attack CS:FTP1_LOGIN action drop
   save

Granular Blocking of HTTP Components

A Juniper Networks security device can selectively block ActiveX controls, Java applets, .zip files, and .exe files sent via HTTP. The danger that these components pose to the security of a network is that they provide a means for an untrusted party to load and then control an application on hosts in a protected network.

When you enable the blocking of one or more of these components in a security zone, the security device examines every HTTP header that arrives at an interface bound to that zone. It checks if the content type listed in the header indicates that any of the targeted components are in the packet payload. If the content type is Java, .exe, or .zip and you have configured the security device to block those HTTP component types, the device blocks the packet. If the content type lists only "octet stream" instead of a specific component type, then the device examines the file type in the payload. If the file type is Java, .exe, or .zip and you have configured the device to block those component types, the device blocks the packet.

When you enable the blocking of ActiveX controls, the device blocks all HTTP packets containing any type of HTTP component in its payload—ActiveX controls, Java applets, .exe files, or .zip files.

NOTE: When ActiveX-blocking is enabled, the security device blocks Java applets, .exe files, and .zip files whether or not they are contained within an ActiveX control.

ActiveX Controls

Microsoft ActiveX technology provides a tool for web designers to create dynamic and interactive web pages. ActiveX controls are components that allow different programs to interact with each other. For example, ActiveX allows your browser to open a spreadsheet or display your personal account from a backend database. ActiveX components might also contain other components such as Java applets, or files such as .exe and .zip files.

When you visit an ActiveX-enabled website, the site prompts you to download ActiveX controls to your computer. Microsoft provides a pop-up message displaying the name of the company or programmer who authenticated the ActiveX code that is offered for download. If you trust the source of the code, you can proceed to download the controls. If you distrust the source, you can refuse them.

If you download an ActiveX control to your computer, it can then do whatever its creator designed it to do. If it is malicious code, it can now reformat your hard drive, delete all your files, send all your personal e-mail to your boss, and so on.

Java Applets

Serving a similar purpose as ActiveX, Java applets also increase the functionality of web pages by allowing them to interact with other programs. You download Java applets to a Java Virtual Machine (VM) on your computer. In the initial version of Java, the VM did not allow the applets to interact with other resources on your computer. Starting with Java 1.1, some of these restrictions were relaxed to provide greater functionality. As a result, Java applets can now access local resources outside the VM. Because an attacker can program Java applets to operate outside the VM, they pose the same security threat as ActiveX controls do.

EXE Files

If you download and run an executable file (that is, a file with a .exe extension) obtained off the Web, you cannot guarantee that the file is uncontaminated. Even if you trust the site from which you downloaded it, it is possible that somebody sniffing download requests from that site has intercepted your request and responded with a doctored .exe file that contains malicious code.

ZIP Files

A zip file (that is, a file with a .zip extension) is a type of file containing one or more compressed files. The danger of downloading a .exe file presented in the previous section about .exe files applies to .zip files, because a .zip file can contain one or more .exe files.

Example: Blocking Java Applets and .exe Files

In this example, you block any HTTP traffic containing Java applets and .exe files in packets arriving at an Untrust zone interface.

WebUI

Screening > Screen (Zone: Untrust): Select **Block Java Component** and **Block EXE Component**, then click **Apply**.

CLI

```
set zone untrust screen component-block jar
set zone untrust screen component-block exe
save
```

Chapter 6

Intrusion Detection and Prevention

An Intrusion Prevention System (IPS), more commonly known as a *firewall*, is used to detect and prevent attacks in network traffic. While firewalls provide perimeter and boundary protection, allowed traffic can hide attacks that firewalls are not designed to detect.

Juniper Networks Intrusion Detection and Prevention (IDP) technology can detect and then stop attacks when deployed inline to your network. Unlike an IPS alone, IDP uses multiple methods to detect attacks against your network and prevent attackers from gaining access and doing damage. IDP can drop malicious packets or connections before the attacks can enter your network. It is designed to reduce false positives and ensure that only actual malicious traffic is detected and stopped. You can also deploy IDP as a passive sniffer, similar to a traditional IPS but with greater accuracy and manageability.

This chapter contains the following sections:

IDP-Capable Security Devices

ScreenOS supports IDP capabilities on some security devices only. The security module, an optional component installed in some security devices provides IDP functionality.

If you purchased a security device with only firewall or virtual private network (VPN) capabilities, you can upgrade the device to an IDP-capable system by doing the following:

- Installing the Advanced and IDP license keys
- Upgrading the boot loader
- Installing an IDP-capable version of ScreenOS
- Upgrading the system memory
- Installing security module (s)

NOTE: Installing the IDP license key disables the ScreenOS Deep Inspection (DI) feature.

Refer to the *ISG 2000 and ISG 1000 Field Upgrade* documents for instructions on how to upgrade the devices to include IDP capabilities.

You can use the IDP-capable security device as a fully integrated firewall/VPN/IDP security system that not only screens traffic between the Internet and your private network but also provides application-level security. You can also use this device as a standalone IDP system to protect critical segments of your private network. For more information, see “Configuring the Device as a Standalone IDP Device” on page 219.

Configuring Basic Intrusion Detection and Prevention

This section presents three basic examples for configuring IDP on your security device:

- "Example 1: Basic IDP Configuration" on page 166
- "Example 2: Configuring IDP for Active–Passive Failover" on page 168
- "Example 3: Configuring IDP for Active–Active Failover" on page 170

Preconfiguration Tasks

Before you start configuring IDP on the device, you need to ensure the following:

- Your security device is IDP-capable. For more information, see "IDP-Capable Security Devices" on page 164.
- You have installed and configured a NetScreen-Security Manager (NSM) system on a management station.

NOTE: Although you can perform basic device configuration using the ScreenOS CLI or WebUI, you need NSM to configure and manage IDP on the security device.

  NetScreen-Security Manager provides integrated policy management, where each security device is linked to one security policy that contains rules defining the types of traffic permitted on the network and the way that traffic is treated inside the network.

- You have a security policy for the device. You can use the default security policy provided in NetScreen-Security Manager, or you can create a custom security policy for the firewall/VPN functions on the device.

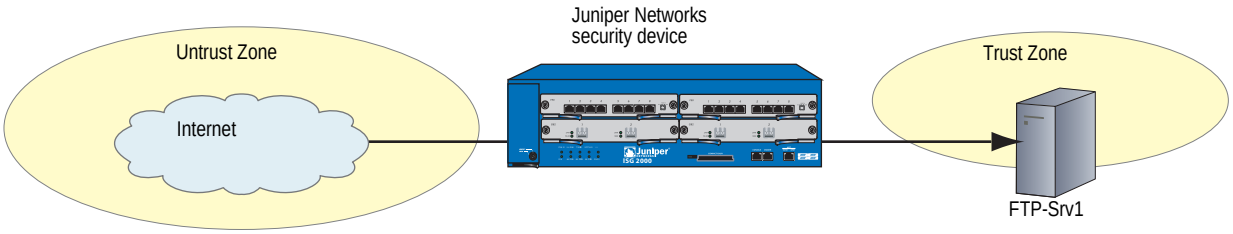
Example 1: Basic IDP Configuration

In this example, a Juniper Networks device is deployed with firewall/VPN/IDP functionality. Before you start configuring, make sure your device is IDP-capable as described in "IDP-Capable Security Devices" on page 164. Set up the device as shown in Figure 56, then do the following:

1. Physically connect the network components.

2. Add the network components that you want IDP to protect using the CLI, WebUI, or NetScreen-Security Manager UI.

Figure 56: Setting Up the Device for Basic IDP



These components can be routers, servers, workstations, subnetworks, or any other objects connected to your network. In NetScreen-Security Manager, these network components are represented as *address objects*. You can also create address object *groups*, which represent multiple address objects. For more information about creating address objects, refer to the *NetScreen-Security Manager Administrator's Guide*.

3. Enable IDP (the default is inline mode) in the appropriate firewall rule for the device.

   This step can be performed using the CLI, Web UI, or NetScreen-Security Manager UI. The CLI commands are shown below (to configure using NSM, see "Enabling IDP in Firewall Rules" on page 177):

```
device-> get policy
Total regular policies 5, Default deny.
ID From     To       Src-address  Dst-address  Service  Action State   ASTLCB
9 Trust     Untrust  Any          Any          MAIL     Permit  enabled --X-X
4 blade1    dmz2     Any          Any          ANY      Permit  enabled --X-X
6 dmz2      blade1   Any          Any          ANY      Permit  enabled --X-X
10 Untrust  Trust    Any          MIP(172.24.~ HTTP     Permit  enabled --X-X
                                               HTTPS
device-> get policy id 4
name:"none" (id 4), zone blade1 -> dmz2,action Permit, status "enable
src "Any", dst "Any", serv "ANY"
Policies on this vpn tunnel: 0
nat off, Web filtering : disabled
vpn unknown vpn, policy flag 00010000, session backup: on
policy IDP mode : disable
traffic shapping off, scheduler n/a, serv flag 00
log yes, log count 0, alert no, counter no(0) byte rate(sec/min) 0/0
total octets 0, counter(session/packet/octet) 0/0/0
No Authentication
No User, User Group or Group expression set
```

```
device-> set policy id 4
device (policy:4)-> set idp
device (policy:4)-> exit
device-> get policy id 4
policy IDP mode : inline
```

4. Add the device using the Netscreen-Security Manager UI.

   To **add the device**, do the following:

   a. Select **Device Manager > Security Devices > + (Device)**.

   b. Enter a device name, then and click **Next**.

   c. Enter the management IP address of the device, then click **Finish**.

   The new device appears in the list of security devices.

5. Validate the security policy on your device.

   Make sure you have a security policy for the device. You can use the default security policy; or, if the device is deployed as an integrated firewall/VPN/IDP device, you can create a custom security policy for the firewall/VPN functions on the device. For more information, see "Configuring Security Policies" on page 173. For more information on configuring security polices using NSM, refer to the *NetScreen-Security Manager Administrator's Guide*.

6. Import the device.

   To **import the device**, right-click on the device that you added, then select **Import device**. Importing the device copies the security-policy information from the device to the NSM server so that the device can be managed. The imported policy is displayed in NSM under **Security Policy**.

   For more information about adding and configuring devices using NSM, refer to the *NetScreen-Security Manager Administrator's Guide*. Other configuration settings include operational mode, administrative password, zone interface assignments, and default route configurations.

7. Add and configure IDP rules in the security policy for the device.

   You configure a security policy on the device to include IDP rules. When you update the configuration on the device, the entire security policy, including IDP rule additions or changes, is installed on the device. For more information about enabling and configuring IDP rules, see "Configuring IDP Rules" on page 178.

NOTE: If you are using the device as a standalone IDP system, you need to configure a simple firewall rule that directs all traffic to be checked against the IDP rules. For more information, see "Configuring the Device as a Standalone IDP Device" on page 219.

---

8. Assign the security policy to the security device.

9. Allow traffic to flow and view the IDP sessions with the following command:

```
device->get sm status
SM CPU aval ena Sess_cnt
1   1     1        10
2   1     1        8
3   0     1        0
4   0     1        0
5   0     1        0
6   0     1        0
```

Security Module 1

Security Module 2

Security Module 3

The above command shows one security module (SM1 and SM2) installed in the device. The CPU column indicates that security modules 2 and 3 are not installed in the device. The status on the two CPUs on each security module is displayed in separate rows.

The management module in the device processes the traffic and then forwards it for IDP inspection to the security modules. The traffic is load-balanced between the two CPUs in the security module (see the **Sess_Cnt** column). If your device has more than one security module, then the management module load-balances the traffic between the security modules.

NOTE: When you have more than one security module installed in the device and one module fails, then the IDP sessions are automatically transferred to the next security module.

---

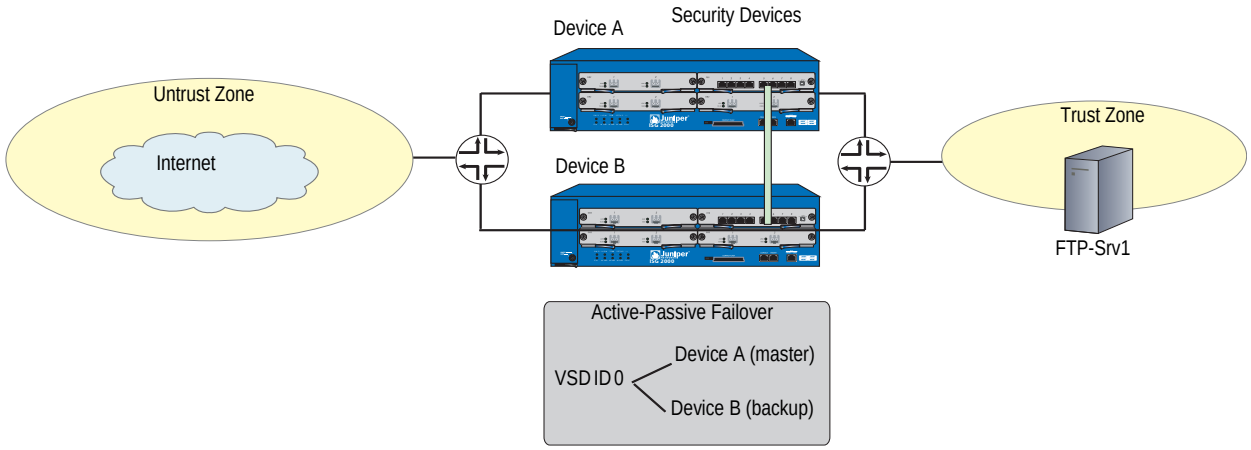
10. Periodically update the attack object database on the NetScreen-Security Manager server.

    See “Managing IDP” on page 222 for more information.

Example 2: Configuring IDP for Active–Passive Failover

In this example, set up your security device in high availability (HA) pairs to remove a potential point of failure from your network design. Figure 57 illustrates device setup for configuring IDP for active-passive failover. The two devices are in an active-passive failover configuration; that is, the primary device is active, handling all firewall and VPN activities; and the backup device is passive, waiting to take over when the primary device fails.

Figure 57: Configuring IDP for Active-Passive Failover



Set up the device as shown in Figure 57, then do the following:

1. Configure Device A and Device B for IDP as described in “Example 1: Basic IDP Configuration” on page 166.

2. To ensure continuous traffic flow, cable and configure two security devices in a redundant cluster with Device A acting as a primary device and Device B acting as its backup.

   Cable e1/x on Device A to e1/x on Device B. Similarly cable the e2/x interfaces. For more information about cabling the two devices together, setting up managed IP addresses to manage a backup device, or removing other potential points of failure by setting up redundant switches on either side of the HA pair, see *Volume 11: High Availability Concepts & Examples ScreenOS Reference Guide*.

3. Configure the HA interfaces.

   Specify the zones with HA interfaces. Bind e1/x and e2/x to the HA zone. Set manage IP addresses for the Trust zone interfaces on both devices.

4. Configure an active-passive NetScreen Redundancy Protocol (NSRP) cluster.

   Assign each device to NSRP cluster ID 0. When the devices become members of the NSRP cluster, the IP addresses of their physical interfaces automatically become the IP addresses of the Virtual Security Interfaces (VSIs) for Virtual Security Device (VSD) group ID 0. Each VSD member has a default priority of 100. The device with the higher unit ID becomes the VSD group’s primary device. For more information about VSDs, see *Volume 11: High Availability Concepts & Examples ScreenOS Reference Guide*.

   For example, enter the following on each of the devices to configure an NSRP cluster:

   a. Add the device to an NSRP cluster and a VSD group.

      ```
      set nsrp cluster id 0
      ```

b. Enable automatic Run-Time Object (RTO) synchronization.

```
set nsrp rto sync all
```

c. Select the ports that you want the devices to monitor.

If the device detects a loss of network connectivity on one of the monitored ports, then it triggers a device failover.

```
set nsrp rto-mirror sync
set nsrp monitor interface ethernet1
set nsrp monitor interface ethernet3
set nsrp cluster id 0
save
```

Upon initial NSRP configuration, the VSD group members with the priority number closest to 0 becomes the primary device. (The default is 100.) If Device A and B have the same priority value, the device with the highest MAC address becomes primary device.

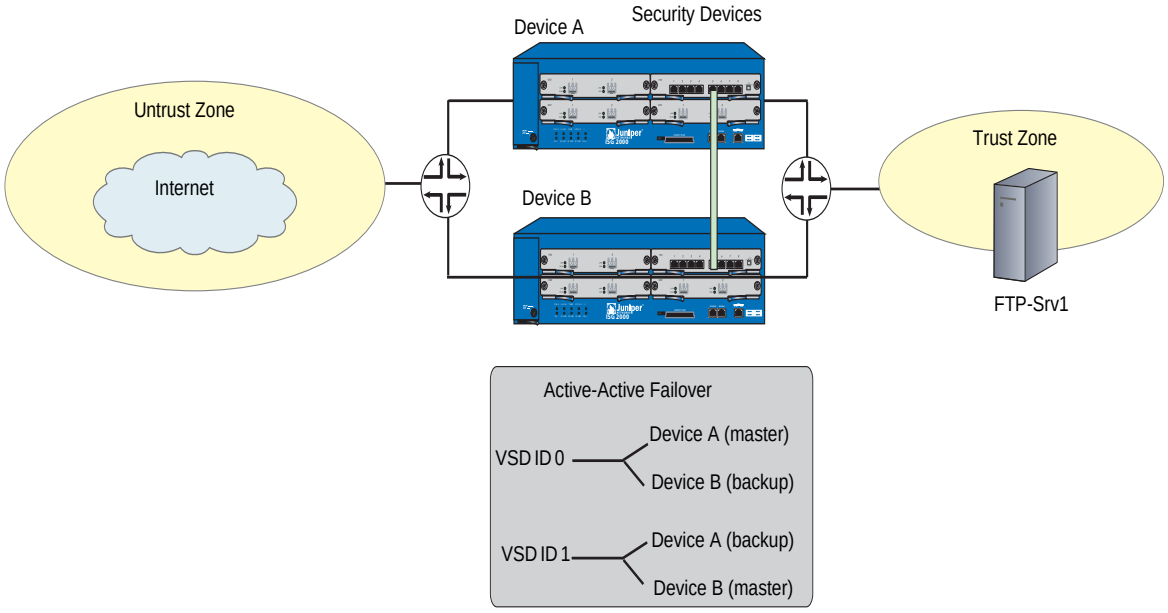
The primary device propagates all its network and configuration settings and the current session information to the backup device. Although the backup device is passive, it is maintaining its synchronization with the information it receives from the primary device. If the primary device fails, the backup device is promoted to primary and takes over the traffic processing.

NOTE: Synchronization is maintained for firewall sessions only. Stateful failover does not occur for IDP sessions.

Example 3: Configuring IDP for Active–Active Failover

In this example, set up your security devices in Route or Network Address Translation (NAT) mode and configure them in a redundant cluster to be active, sharing the traffic distributed between them. This is accomplished using NSRP to create two VSD groups as shown in Figure 58. Device A acts as the primary device in VSD group 1 and as the backup of VSD group 2. Device B acts as the primary device in VSD group 2 and as the backup of VSD group 1. No single point of failure exists in an active-active setup.

Figure 58: Configuring IDP for Active-Active Failover



Set up the device as shown in Figure 58, then do the following:

NOTE: We recommended that the same number of security modules be installed on both devices.

---

1. Configure Device A and Device B for IDP as described in "Example 1: Basic IDP Configuration" on page 166.

2. To ensure continuous traffic flow, cable and configure two security devices in a redundant cluster.

   Cable e1/x on Device A to e1/x on Device B. Similarly cable the e2/x interfaces. For more information on how to cable the two devices together, setting up managed IP addresses to manage a backup device, or to remove other potential points of failure by setting up redundant switches on either side of the HA pair, see *Volume 11: High Availability Concepts & Examples ScreenOS Reference Guide*.

3. Configure the HA interfaces.

   Specify the zones with HA interfaces. Bind e1/x and e2/x to the HA zone. Set manage IP addresses for the trust zone interfaces on both devices.

4. Configure an active-active NSRP cluster.

   Devices A and B are members of the same NSRP cluster and VSD group 0. For active-active failover, create a second VSD group—group 1.

   1. Assign Device A priority 1 in group 0 and the default priority (100) in group 1.

2. Assign Device B priority 1 in group 1 and the default priority (100) in group 0.

In both VSD groups, enable the preempt option on the primary device and set the preempt hold-down time to 10 seconds. If both devices are active, Device A is always the primary device of group 1 and Device B is always the primary device of group 2.

Device A

```
set nsrp vsd-group id 0 priority 1
set nsrp vsd-group id 0 preempt hold-down 10
set nsrp vsd-group id 0 preempt
set nsrp vsd-group id 1
save
```

Device B

```
set nsrp vsd-group id 1 priority 1
set nsrp vsd-group id 1 preempt hold-down 10
set nsrp vsd-group id 1 preempt
save
```

For more information about creating two VSD groups, see *Volume 11: High Availability Concepts & Examples ScreenOS Reference Guide*.

Devices A and B each receive 50 percent of the network and VPN traffic. When Device A fails, Device B becomes the primary device of VSD group 1, as well as continuing to be the primary device of VSD group 2, and handles 100 percent of the traffic.

NOTE: Synchronization is maintained for firewall sessions only. Stateful failover does not occur for IDP sessions.

Configuring Security Policies

A security policy defines how your managed devices handle network traffic. You can configure multiple security policies in NetScreen-Security Manager, but a device can only have one active security policy at a time. You can install the same security policy on multiple devices, or you can install a unique security policy on each device in your network.

About Security Policies

Each instruction in a security policy is called a *rule*. Security policies can contain multiple rules. You create rules in *rulebases*, sets of rules that combine to define a security policy. Each security policy contains the Zone and Global firewall rulebases, which cannot be deleted. You can add or delete any other rulebase—Multicast, IDP, Exempt, and Backdoor—in a security policy; however, a single policy can only contain one instance of any type of rulebase. Each security policy (all rulebases combined) can contain a maximum of 40,000 rules.

This section describes the IDP, Exempt, and Backdoor rulebases. For more information about Zone and Global firewall rulebases and the Multicast rulebase, refer to the information about configuring security policies in the *NetScreen-Security Manager Administrator's Guide*.

NOTE: In the ScreenOS WebUI and CLI, a security policy is a single statement that defines a source, destination, zone, direction, and service. In NetScreen-Security Manager, those same statements are known as *rules*, and a security policy is a collection of rules.

Managing Security Policies

Within security policies, you can manage individual rules in each rulebase, including:

- Determining the order in which rules are applied to network traffic
- Disabling a rule
- Negating source or destination addresses (ScreenOS 5.x devices only)
- Verifying the security policy
- Merging security policies

NOTE: The IDP, Exempt, and Backdoor rulebases are not included when you merge two policies into a single policy.

For detailed information about managing your security policy, refer to the *NetScreen-Security Manager Administrator's Guide*.

Installing Security Policies

After you create a security policy by building rules in one or more rulebases, you can assign, validate, and install that policy on specific managed devices. For detailed information about installing security policies, refer to the *NetScreen-Security Manager Administrator's Guide*.

Using IDP Rulebases

After a firewall rule (intrazone or global) has permitted the network traffic, you can direct the device to further inspect the traffic for known attacks. NetScreen-Security Manager supports the following IDP rulebases:

- **IDP:** This rulebase protects your network from attacks by using attack objects to detect known and unknown attacks. Juniper Networks provides predefined attack objects that you can use in IDP rules. You can also configure your own custom attack objects. For more information, see "Configuring IDP Attack Objects" on page 201.

NOTE: Juniper Networks regularly updates predefined attack objects to keep current with newly discovered attacks. For more information about updating attack objects, see "Managing IDP" on page 222.

- **Exempt:** This rulebase works in conjunction with the IDP rulebase to prevent unnecessary alarms from being generated. You configure rules in this rulebase to exclude known false positives or to exclude a specific source, destination, or source/destination pair from matching an IDP rule. If traffic matches a rule in the IDP rulebase, IDP attempts to match the traffic against the Exempt rulebase before performing the action specified.

- **Backdoor Detection:** This rulebase protects your network from mechanisms installed on a host computer that facilitate unauthorized access to the system. Attackers who have already compromised a system typically install backdoors (such as Trojans) to make future attacks easier. When attackers send information to and retrieve information from a backdoor program, they generate interactive traffic that IDP can detect.

The rules in all rulebases, including the Zone, Global, and Multicast rulebases, combine to create a security policy. To direct the device to process and execute rules in the IDP rulebases, you need to enable IDP in a firewall rule. See "Enabling IDP in Firewall Rules" on page 177.

NOTE: If you import the device into NetScreen-Security Manager, the imported device configuration does not include the IDP, Exempt, or Backdoor rulebases.

Role-Based Administration of IDP Rulebases

NetScreen-Security Manager's role-based administration (RBA) allows you to create custom roles for individual administrators to give them authority to view or edit IDP rulebases. For more information about RBA, refer to the *NetScreen-Security Manager Administrator's Guide*. You can assign view or edit capabilities for a role based on a IDP rulebase. For example, an administrator who can view and edit a firewall rulebase may be able to only view IDP and Backdoor rulebases.

By default, the predefined roles System Administrator and Domain Administrator can view and edit all rulebases, and the Read-Only System Administrator and Read-Only Domain Administrator can only view rulebases. When you create a new role, the New Role dialog box allows you to specify whether an administrator can view or edit IDP or Backdoor rulebases.

Configuring Objects for IDP Rules

Objects are reusable logical entities that you can apply to rules. Each object that you create is added to a database for the object type. You can use the following types of objects:

- **Address objects** represent components of your network, such as host machines, servers, and subnets. You use address objects in security policy rules to specify the network components that you want to protect.

NOTE: You must create each object in the Address Object database. There are no default address objects.

For information about creating address objects, refer to the *NetScreen-Security Manager Administrator's Guide*.

- **Service objects** represent network services that use Transport layer protocols such as TCP, UDP, RPC, and ICMP. You use service objects in rules to specify the service an attack uses to access your network. NetScreen-Security Manager includes predefined service objects, a database of service objects that are based on industry-standard services. If you need to add service objects that are not included in the predefined service objects, you can create custom service objects. For more information about creating service objects, refer to the *NetScreen-Security Manager Administrator's Guide*.

- **IDP attack objects** represent known and unknown attacks. IDP includes a predefined attack object database that is periodically updated by Juniper Networks (see "Managing IDP" on page 222). You can also add custom attack objects to detect attacks that are unique to your network (see "Configuring IDP Attack Objects" on page 201.)

Using Security Policy Templates

When you create a new security policy, you have the following options:

- Create a security policy that contains a default firewall rule.
- Select a predefined template.
- Copy an existing security policy into a new policy, which you can then modify.

A template is a set of rules of a specific rulebase type that you can use as a starting point when creating a security policy. For a list of templates, refer to the *NetScreen-Security Manager Administrator's Guide*.

Enabling IDP in Firewall Rules

The rules in all rulebases combine to create a security policy. Security devices process and execute rules in each rulebase in the following order:

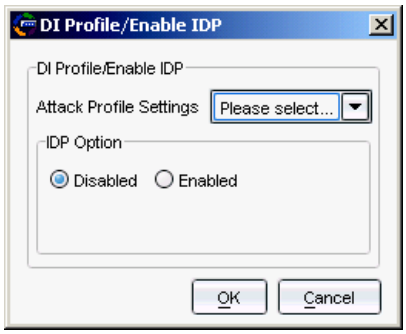
1. Zone-based firewall
2. Global firewall
3. Multicast
4. IDP
5. Exempt
6. Backdoor

Enabling IDP in a zone-based or global firewall rule directs traffic that matches the firewall rule to be checked against the IDP rulebases.

NOTE: The firewall action must be **permit**. You cannot enable IDP for traffic that the device denies or rejects.

To enable IDP in a firewall rule, right-click in the Rule Options column for the zone-based or global firewall rule, then select **DI Profile/Enable IDP**. The DI Profile/Enable IDP dialog box appears, as shown in Figure 59.

Figure 59: DI Profile/Enable IDP Dialog Box



NOTE: These attack-profile settings apply only to the Deep Inspection (DI) feature on firewall/VPN devices. When you install the IDP license on the device, DI is disabled on the device.

Enabling IDP

By default, the IDP option is disabled. Select **Enable** to enable IDP for traffic that matches the firewall rule. When you enable IDP, you can also select whether the IDP function is to operate inline or in inline tap mode on the device on which the security policy is installed.

NOTE: If you do not enable IDP in a firewall rule for a target device, you can still configure IDP rules for the device. However, you will not be able to apply the IDP rules when you update the security policy on the device.

Specifying Inline or Inline Tap Mode

IDP on the target device can operate in one of two modes:

- In **inline** mode, IDP is directly in the path of traffic on your network and can detect and block attacks. For example, you can deploy the security device with integrated firewall/VPN/IDP capabilities between the Internet and an enterprise LAN, WAN, or special zone such as the DMZ. This is the default mode.
- In **inline tap** mode, IDP receives a copy of a packet while the original packet is forwarded on the network. IDP examines the copy of the packet and flags any potential problems. IDP's inspection of packets does not affect the forwarding of the packet on the network.

NOTE: You must deploy the IDP-capable device inline. You cannot connect a device that is in inline tap mode to an external TAP or SPAN port on a switch.

You specify the IDP mode as part of the security policy for the device.

Configuring IDP Rules

The IDP rulebase protects your network from attacks by using attack objects to identify malicious activity and then by taking action to thwart the attacks. Caution against configuring a large number of IDP rules and having performance issues.

When you create an IDP rule, you must specify the following:

- The type of network traffic you want IDP to monitor for attacks, using the following characteristics:
  - **From Zone/To Zone:** All traffic flows from a source to a destination zone. You can select any zone for the source or destination; however, the zone must be valid for the security devices you select in the Install On column of the rule. You can also use zone exceptions to specify unique **to** and **from** zones for each device.
  - **Source IP:** The source IP address from which the network traffic originates. You can set this to "any" to monitor network traffic originating from any IP address. You can also specify "negate" to specify all sources except the specified addresses.

- **Destination IP:** The destination IP address to which the network traffic is sent. You can set this to "any" to monitor network traffic sent to any IP address. You can also specify "negate" to specify all destinations except the specified addresses.

- **Service:** The Application Layer protocols supported by the destination IP address.

- **Terminate Match:** By default, rules in the IDP rulebase are *nonterminal*, meaning that IDP examines all rules in the rulebase and executes all matches. You can specify that a rule is *terminal*; if IDP encounters a match for the source, destination, and service specified in a terminal rule, it does not examine any subsequent rules for that connection. Note that the traffic does not need to match the attacks specified in the terminal rule. Terminal rules should appear near the top of the rulebase, before other rules that would match the same traffic. Use caution when specifying terminal rules.

  See Figure 69 on page 186 and not that if you check Terminate Match, rules below the Terminate Match Rule (Rule Shadowing) are not evaluated.

  If you do not check Terminate Match, multi-event logging/matching occur, which results in one attack creating multiple entries in the logs and multiple actions.

- The attack(s) you want IDP to match in the monitored network traffic. Each attack is defined as an *attack object*, which represents a known pattern of attack. Whenever this known pattern of attack is encountered in the monitored network traffic, the attack object is matched. You can add attack objects by category, operating system, severity, or individually.

- The action you want IDP to take when the monitored traffic matches the rule's attack objects. You can specify the following:

  - **Action:** The action you want IDP to perform against the current connection.

  - **IP Actions:** The action you want IDP to perform against future connections that use the same IP address.

  - **Notification:** Choose **none**; or enable logging, then select the appropriate logging options for your network.

  - **Severity:** Use the default severity settings of the selected attack objects, or choose a specific severity for your rule.

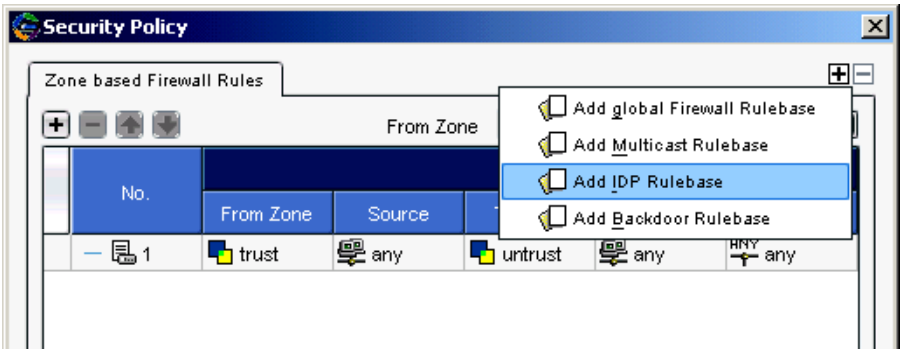
Adding the IDP Rulebase

Before you can configure a rule in the IDP rulebase, you need to add the IDP rulebase to a security policy using the following steps:

1. In the main navigation tree, select **Security Policies**. Open a security policy either by double-clicking on the policy name in the Security Policy window or by clicking on the policy name and then selecting the Edit icon.

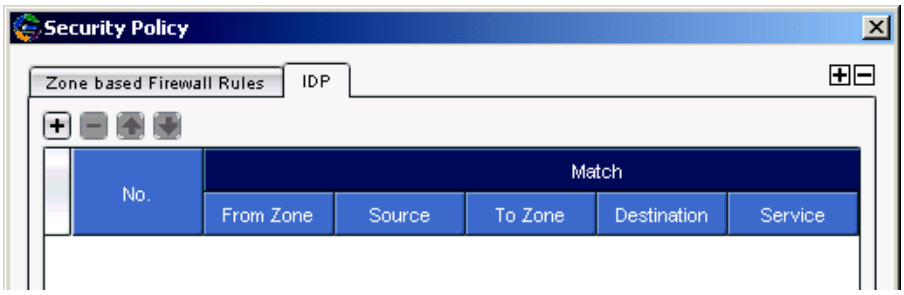
2. Click the Add icon in the upper right corner of the Security Policy window, then select **Add IDP Rulebase.** See Figure 60.

Figure 60: Adding an IDP Rulebase



The IDP rulebase tab appears. See Figure 61.

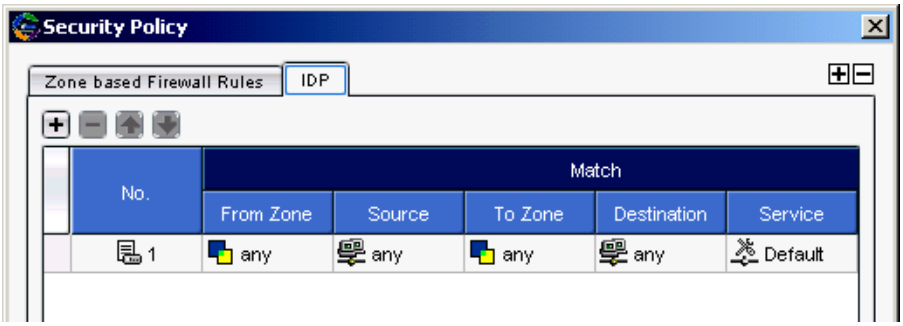
Figure 61: IDP Rulebase Added



3. To configure an IDP rule, click the Add icon on the left side of the Security Policy window.

   A default IDP rule appears. You can modify this rule as needed. See Figure 62.

Figure 62: IDP Rule Added



Matching Traffic

When creating your IDP rules, you must specify the type of network traffic that you want IDP to monitor for attacks. These characteristics include the network components that originate and receive the traffic and the firewall zones the traffic passes through.

The Match columns From Zone, Source, To Zone, Destination, and Service are required for all rules in the IDP rulebase. The Terminate Match selection allows you to designate a rule as terminal; if IDP encounters a match for the other Match columns in a terminal rule, no other rules in the rulebase are examined. The matching traffic does not need to match the attacks specified in a terminal rule. (For more information, see "Terminal Rules" on page 185.)

The following sections detail the Match columns of an IDP rule.

Source and Destination Zones

You can select multiple zones for the source and destination; however, these zones must be available on the security devices on which you will install the policy. You can specify "any" for the source or destination zones to monitor network traffic originating from or destined for any zone.

NOTE: You can create custom zones for some security devices. The list of zones from which you can select source and destination zones includes the predefined and custom zones that have been configured for all devices managed by NetScreen-Security Manager. Therefore, you should only select zones that are applicable for the device on which you will install the security policy.

Source and Destination Address Objects

In the NetScreen-Security Manager system, address objects are used to represent components on your network: hosts, networks, servers, and so on. Typically, a server or other device on your network is the destination IP for incoming attacks and can sometimes be the source IP for interactive attacks (see "Configuring Backdoor Rules" on page 197 for more information about interactive attacks). You can specify "any" to monitor network traffic originating from any IP address. You can also "negate" the address object(s) listed in the Source or Destination column to specify all sources or destinations except the excluded object(s).

You can create address objects either before you create an IDP rule (refer to the *NetScreen-Security Manager Administrator's Guide)* or while creating or editing an IDP rule. To select or configure an address object, right-click on either the Source or the Destination column of a rule, then select **Select Address**. In the **Select Source Addresses** dialog box, you can either select an already created address object or click the Add icon to create a new host, network, or group object.

Example: Setting Source and Destination

You want to detect incoming attacks that target your internal network. Set the From Zone to **Untrust** and the Source IP to **any**. Set the To Zone to **dmz** and **trust**. Select the address object that represents the host or server you want to protect from attacks as the Destination IP.

Your rule looks similar to the example shown in Figure 63.

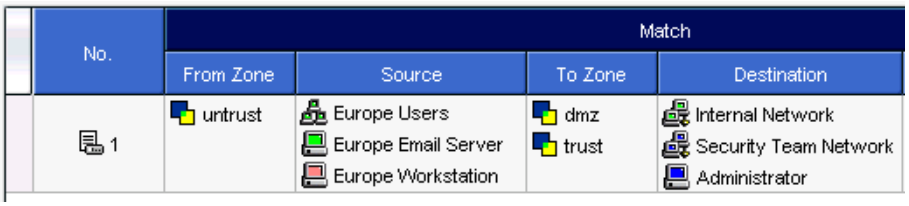
Figure 63: Set Source and Destination

Example: Setting Multiple Sources and Destinations

You want to detect attacks between two networks. Select multiple address objects for the Source and Destination.

Your rule looks similar to the example in Figure 64.

Figure 64: Set Multiple Source and Destination Networks



The more specific you are in defining the source and destination of an attack, the more you reduce false positives.

Services

Services are Application Layer protocols that define how data is structured as it travels across the network. Because the services you support on your network are the same services that attackers must use to attack your network, you can specify which services are supported by the destination IP to make your rule more efficient.

NOTE: All services rely on a Transport Layer protocol to transmit data. IDP includes services that use the TCP, UDP, RPC, and ICMP Transport Layer protocols.

Service objects represent the services running on your network. NetScreen-Security Manager includes predefined service objects that are based on industry-standard services. You use these service objects in rules to specify the service an attack uses to access your network. You can also create custom service objects to represent protocols that are not included in the predefined services. For more information about configuring service objects, refer to the information about object configuration in the *NetScreen-Security Manager Administrator's Guide*.

In the Service column, you select the service of the traffic you want IDP to match:

- Select **Default** to accept the service specified by the attack object you select in the Attacks column. When you select an attack object in the Attack column, the service associated with that attack object becomes the default service for the rule. To see the exact service, view the attack object details.

- Select **Any** to set any service.
- Select **Select Service** to choose specific services from the list of defined service objects.

Example: Setting Default Services

You want to protect your FTP server from FTP attacks. Set the service to Default, and add an attack object that detects FTP buffer overflow attempts. The Service column in the rule still displays “Default”, but the rule actually uses the default service of TCP-FTP, which is specified in the attack object.

Your rule looks similar to the example shown in Figure 65.

Figure 65: Set Default Services

Match						Action	Attacks
From Zone	Source	To Zone	Destination	Service	Terminate Ma...		
untrust	any	dmz trust	FTP Server	Default	☑	None	FTP - Critical

Example: Setting Specific Services

Your mail server supports POP3 and SMTP connections but does not support IMAP. Set POP3 and SMTP service objects as services that can be used to attack the mail server. Because IMAP is not supported, you do not need to add the IMAP service object.

Your rule looks similar to the example in Figure 66.

Figure 66: Set Specific Services

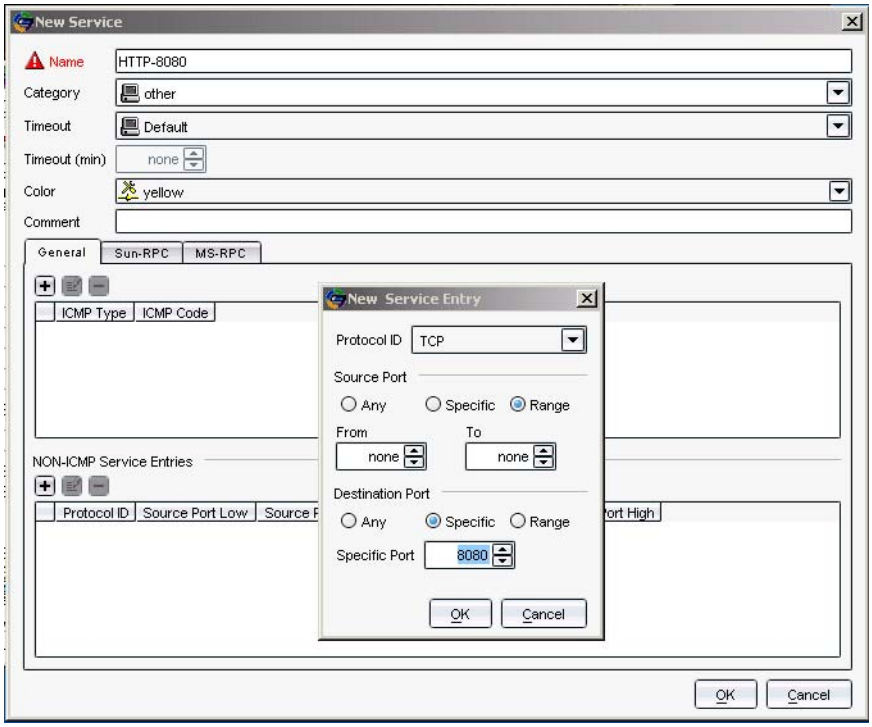
Match				
From Zone	Source	To Zone	Destination	Service
untrust	any	dmz trust	Web Server	SMTP POP3

If you are supporting services on nonstandard ports, you should choose a service other than the default.

Example: Setting Nonstandard Services

You use a nonstandard port (8080) for your HTTP services. Use the Object Manager to create a custom service object on port 8080.

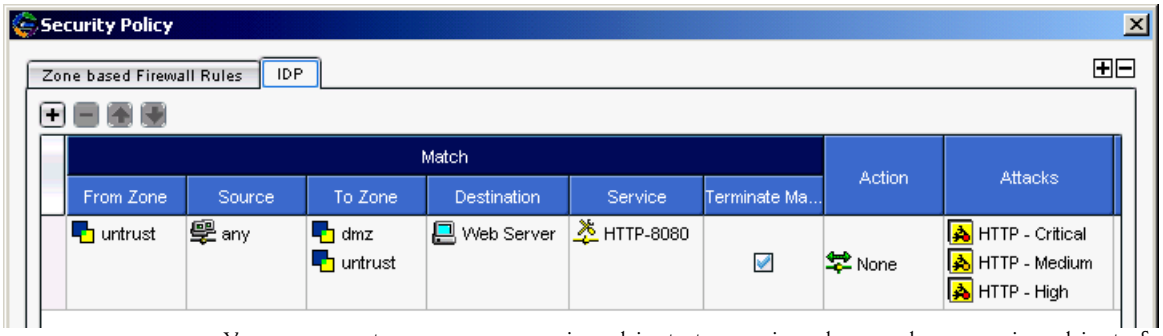
Figure 67: Add Nonstandard Services Object



Add this service object to your rule, then add several HTTP attack objects, which have a default service of TCP/80. IDP uses the specified service, HTTP-8080, instead of the default and looks for matches to the HTTP attacks in TCP traffic on port 8080.

Your rule looks similar to the example in Figure 68.

Figure 68: Set Nonstandard Service



You can create your own service objects to use in rules, such as service objects for protocols that use nonstandard ports. However, you cannot match attack objects to protocols they do not use.

Terminal Rules

The normal IDP rule-matching algorithm starts from the top of the rulebase and checks traffic against all rules in the rulebase that match the source, destination, and service. A terminal rule is an exception to this normal rule-matching algorithm. When a match is discovered in a terminal rule for the source, destination, and service, IDP does not continue to check subsequent rules for the same source, destination, and service. It does not matter whether or not the traffic matches the attack objects in the matching rule.

You can use a terminal rule for the following purposes:

- To set different actions for different attacks for the same Source and Destination. This is illustrated by rules 3 and 6 in the following section, "Example: Setting Terminal Rules".
- To disregard traffic that originates from a known trusted source. Typically, the action is None for this type of terminal rule. This is illustrated by rule 1 in the following section, "Example: Setting Terminal Rules".
- To disregard traffic sent to a server that is only vulnerable to a specific set of attacks. Typically, the action is Drop Connection for this type of terminal rule.

Use caution when defining terminal rules. An inappropriate terminal rule can leave your network open to attacks. Remember that traffic matching the source, destination, and service of a terminal rule is not compared to subsequent rules, even if the traffic does not match an attack object in the terminal rule. Use a terminal rule only when you want to examine a certain type of traffic for one specific set of attack objects. Be particularly careful about terminal rules that use "any" for both the source and destination.

Terminal rules should appear near the top of the rulebase before other rules that would match the same traffic. You set a rule as terminal by selecting the box in the Terminate Match column of the Security Policy window when you create or modify the rule.

NOTE: In many cases, you can use an exempt rule instead of a terminal rule. You might find it easier and more straightforward to configure an exempt rule than a terminal rule. See "Configuring Exempt Rules" on page 192.

Example: Setting Terminal Rules

In the example IDP rulebase shown below, rules 1, 3, 4, and 5 are configured as terminal rules:

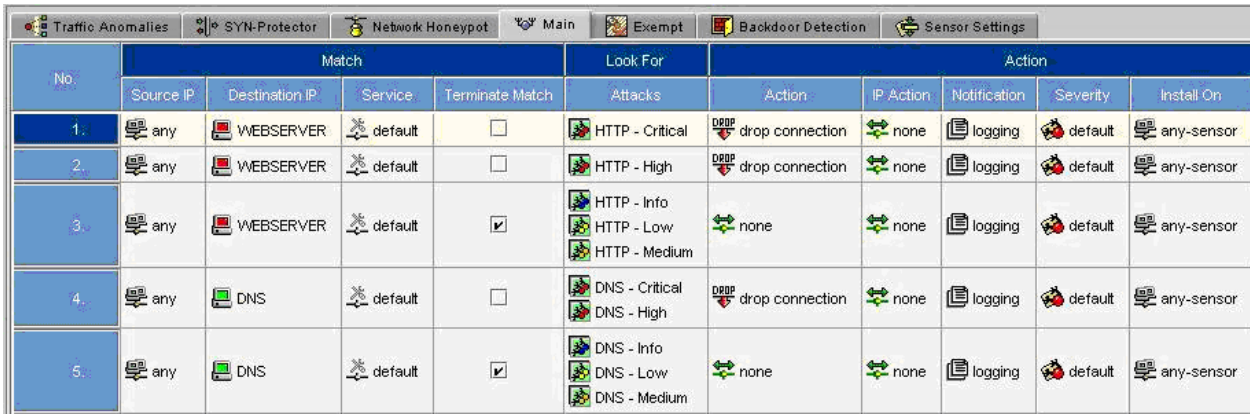
- Rule 1 terminates the match algorithm if the source IP of the traffic originates from the security network, a known trusted network. If this rule is matched, IDP disregards traffic from the security network and does not continue monitoring the session for malicious data.
- Rules 3 and 6 set different actions for different attacks when the destination IP is the Corporate or Europe email server. Rule 3 terminates the match algorithm when the attack is an email that uses the SMTP context Confidential. Rule 6 closes the server when the attack is an SMTP attack.

- Rule 4 terminates the match algorithm when the destination is the webserver and the attack is a Critical or High HTTP attack. The rule ensures that IDP drops the most important HTTP attacks against the webserver and does not continue to match the connection.

- Rule 5 terminates the match algorithm when the source is the internal network and the attack is a critical, high, or medium trojan backdoor. The rule ensures that IDP closes both the client and server and does not continue to match the connection.

The default in the Service Column (see Figure 69 on page 186) means the rule is dynamically built based on the service bindings of the attack objects of that rule. To see the service bindings for a rule, right click on the attacks and select **View Services**. Even if you select a broad category like HTTP Critical, use the View Services for more details.

Figure 69: Set Terminal Rules



Defining Actions

You can specify which actions IDP is to perform against attacks that match rules in your security policy. For each attack that matches a rule, you can choose to ignore, drop, or close the current attacking packets or connection. If the rule is triggered, IDP can perform actions against the connection.

IDP drops traffic only when it is running in inline mode; when IDP is running in inline tap mode, it cannot perform drop or close actions. However, IDP running in inline tap mode and configured for Drop Packet finds a TCP attack, then the Security Module informs the Management Module that successive packets are attacks and consequently the IDP Action is updated to a higher severity, Drop Connection.

Table 16 shows the actions you can specify for IDP rules.

Table 16: IDP Rule Actions

Action	Description
None	IDP takes no action against the connection. If a rule that contains an action of None is matched, the corresponding log record displays "accept" in the action column of the Log Viewer.
Ignore	IDP ignores a flow for future inspection if an attack match was found. Generally, avoid selecting this action. **Note:** This action does not mean ignore an attack.
Drop Packet	IDP drops a matching packet before it can reach its destination but does not close the connection. Use this action to drop packets for attacks in traffic that is prone to spoofing, such as UDP traffic. Dropping a connection for such traffic could result in a denial of service that prevents you from receiving traffic from a legitimate source-IP address.
Drop Connection	IDP drops all packets associated with the connection. Use this action to drop connections for traffic that is not prone to spoofing.
Close Client and Server	IDP closes the connection and sends an RST packet to both the client and the server. If IDP is operating in inline tap mode, IDP sends an RST packet to both the client and server but does not close the connection.
Close Client	IDP closes the connection to the client but not to the server.
Close Server	IDP closes the connection to the server but not to the client.

Setting Attack Objects

Attack objects represent specific patterns of malicious activity within a connection, and are a method for detecting attacks. Each attack object detects a known or an unknown attack that can be used to compromise your network. From more information about attack objects, see "Configuring IDP Attack Objects" on page 201.

You can add attack objects to your rule individually or in groups. Attack objects are organized as follows:

- **Attack List** is an alphabetical list of all attack objects, including custom attack objects.

- **Dynamic Attack Group** contains predefined and custom attack groups.

To add attack objects for a rule, right-click the Attacks column of the rule, then select **Select Attacks**. The Add Attacks dialog box appears.

Adding Attack Objects Individually

The Attack List allows you to select one or more specific attack objects for your rule. The Attack List contains attack objects displayed in alphabetical order. You can also use the integrated search function in NetScreen-Security Manager to locate a specific word or string in the attack object name. For more information about using the search feature, refer to the *NetScreen-Security Manager Administrator's Guide*.

For more information about attack objects and creating custom attack objects and groups, see "Configuring IDP Attack Objects" on page 201.

Adding Attack Objects by Category

IDP groups attack objects into predefined service category groups. Services are Application Layer protocols which define how data is structured as it travels across the network.

To attack a system, an attacker must use a protocol supported on that system. Therefore, When you create a rule to protect a system, you must select only the categories that are used by the address objects you are protecting with the rule.

Example: Adding Attack Objects by Service

You rely on FTP and HTTP for extensive file transfer on your webserver. Choose the FTP and HTTP category groups to carefully monitor all traffic that uses these services.

If you do not want to choose an entire category group for a rule, you can select your attack objects by severity.

Adding Attack Objects by Operating System

IDP groups attack objects for several predefined operating systems to help you choose the attack objects that are the most dangerous to specific devices on your network. You can choose BSD, Linux, Solaris, or Windows.

If you do not want to choose an entire operating system group for a rule, you can select your attack objects by severity.

Adding Attack Objects by Severity

IDP defines five severity levels, each with a recommended set of IDP actions and notifications (see Table 17). You can add a severity level to the Attacks column in your rule, then choose the recommended actions for the severity level in the Action column. (For more information about the actions you can select, see "Defining Actions" on page 187.) You can also choose the recommended notifications for the severity level in the Notifications column. (For more information about the notifications you can select, see "Setting Notification" on page 190.)

NOTE: To protect critical address objects or popular targets for attack, such as your mail server, use multiple severity levels to ensure maximum protection.

Table 17 shows the IDP severity levels, along with their recommended actions and notifications.

Table 17: Severity Levels with Recommended Actions and Notifications

Severity Level	Description	Recommended Action	Recommended Notification
Critical	Attacks attempt to evade an IPS, crash a machine, or gain system-level privileges.	Drop Packet Drop Connection	Logging Alert
Major	Attacks attempt to crash a service, perform a denial-of-service, install or use a Trojan, or gain user-level access to a host.	Drop Packet Drop Connection	Logging Alert
Minor	Attacks attempt to obtain critical information through directory traversal or information leaks.	None	Logging
Warning	Attacks attempt to obtain noncritical information or scan the network. They can also be obsolete attacks (but probably harmless) traffic.	None	Logging
Info	Attacks are normal, harmless traffic containing URLs, DNS lookup failures, and SNMP public community strings. You can use informational attack objects to obtain information about your network.	None	None

Setting IP Action

The IP Action column appears only when you view the security policy in Expanded mode. To change the security policy view from Compact to Expanded mode, select **View > Expanded Mode**.

If the current network traffic matches a rule, IDP can perform an IP action against future network traffic that uses the same IP address. IP actions are similar to other actions; they direct IDP to drop or close the connection. However, because you now also have the attacker's IP address, you can choose to block the attacker for a specified time. If attackers cannot immediately regain a connection to your network, they might try to attack easier targets.

Use IP actions in conjunction with actions and logging to secure your network. In a rule, first configure an action to detect and prevent current malicious connections from reaching your address objects. Then, right-click in the IP Action column of the rule and select **Configure** to bring up the Configure IP Action dialog box. Enable and configure an IP action to prevent future malicious connections from the attacker's IP address.

Choosing an IP Action

For each IP action option, an IP action is generated by the IDP system. The IP action instructs IDP to perform the specified task. Select from the following options:

- **IDP Notify**. IDP does not take any action against future traffic, but logs the event. This is the default.

- **IDP Drop**. IDP drops the matching connection and blocks future connections that match the criteria in the Blocking Options box.
- **IDP Close**. IDP closes future connections that match the criteria in the Blocking Options box.

Choosing a Blocking Option

Each blocking option follows the criteria you set in the Actions box. Blocking options can be based on the following matches of the attack traffic:

- **Source, Destination**, **Destination Port** and **Protocol**. IDP blocks future traffic based on the source, destination, destination port, and protocol of the attack traffic. This is the default.
- **Source**. IDP blocks future traffic based on the source of the attack traffic.
- **Destination**. IDP blocks future traffic based on the destination of the attack traffic.
- **From Zone**, **Destination**, **Destination Port** and **Protocol**. IDP blocks future traffic based on the source zone, destination, destination port, and protocol of the attack traffic.
- **From Zone**. IDP blocks future traffic based on the source zone of the attack traffic.

Setting Logging Options

When IDP detects attack traffic that matches a rule and triggers an IP action, IDP can log information about the IP action or create an alert in the Log Viewer. By default, no logging options are set.

Setting Timeout Options

You can set the number of seconds that you want the IP action to remain in effect after a traffic match. For permanent IP actions, the default timeout value is 0.

Setting Notification

The first time you design a security policy, you might be tempted to log all attacks and let the policy run indefinitely. Don't do this! Some attack objects are informational only, and others can generate false positives and redundant logs. If you become overloaded with data, you can miss something important. Remember that security policies that generate too many log records are hazardous to the security of your network, as you might discover an attack too late or miss a security breach entirely if you have to sift through hundreds of log records. Excessive logging can also affect IDP throughput, performance, and available disk space. A good security policy generates enough logs to fully document only the important security events on a network.

Setting Logging

In the Configure Notification dialog box, select **Logging**, then click **OK**. Each time the rule is matched, the IDP system creates a log record that appears in the Log Viewer.

Logging an attack creates a log record that you can view in realtime in the Log Viewer. For more critical attacks, you can also set an alert flag to appear in the log record.

To log an attack for a rule, right-click the Notification column of the rule, then select **Configure**. The Configure Notification dialog box appears.

Setting an Alert

In the Configure Notification dialog box, select **Alert**, then click **OK**. If **Alert** is selected and the rule is matched, IDP places an alert flag in the Alert column of the Log Viewer for the matching log record.

Logging Packets

You can record individual packets in network traffic that match a rule by capturing the packet data for the attack. Viewing the packets used in an attack on your network can help you determine the extent of the attempted attack and its purpose, whether or not the attack was successful, and any possible damage to your network.

NOTE: To improve IDP performance, log only the packets received after the attack.

If multiple rules with packet capture enabled match the same attack, IDP captures the maximum specified number of packets. For example, you configure rule 1 to capture 10 packets before and after the attack, and you configure rule 2 to capture 5 packets before and after the attack. If both rules match the same attack, IDP attempts to capture 10 packets before and after the attack.

NOTE: Packet captures are restricted to 256 packets before and after an attack.

Setting Severity

The Severity column appears only when you view the security policy in Expanded mode. To change the security policy view from Compact to Expanded mode, from the menu bar, select **View > Expanded Mode**.

You can override the inherent severity for an attack on a per-rule basis within the IDP rulebase. You can set the severity level to Default, Info, Warning, Minor, Major, or Critical.

To change the severity for a rule, right-click the Severity column of the rule, then select a severity.

Setting Targets

For each rule in the IDP rulebase, you can select the security device that will use that rule to detect and prevent attacks. When you install the security policy to which the rule belongs, the rule becomes active only on the device(s) you selected in the Install On column of the rulebase.

Entering Comments

You can enter notations about the rule in the Comments column. The information in the Comments column is not pushed to the target device(s). To enter a comment, right-click on the Comments column, then select **Edit Comments**. The Edit Comments dialog box appears. You can enter a comment of up to 1024 characters.

Configuring Exempt Rules

The Exempt rulebase works in conjunction with the IDP rulebase. Before you can create exempt rules, you must first create rules in the IDP rulebase. If traffic matches a rule in the IDP rulebase, IDP attempts to match the traffic against the Exempt rulebase before performing the specified action or creating a log record for the event.

NOTE: If you delete the IDP rulebase, the Exempt rulebase is also deleted.

You might want to use an exempt rule under the following conditions:

- When an IDP rule uses an attack object group that contains one or more attack objects that produce false positives or irrelevant log records.
- When you want to exclude a specific source, destination, or source/destination pair from matching an IDP rule. This prevents IDP from generating unnecessary alarms.

You can also use an exempt rule if the IDP rulebase uses static or dynamic attack-object groups containing one or more attack objects that produce false positives or irrelevant log records.

When you create an exempt rule, you must specify the following:

- Source and destination for traffic you want to exempt. You can set the source or destination to “any” to exempt network traffic originating from any source or sent to any destination. You can also specify “negate” to specify all sources or destinations except specified addresses.
- The attack(s) you want IDP to exempt for the specified source/destination addresses. You must include at least one attack object in an exempt rule.

NOTE: The Exempt rulebase is a nonterminal rulebase. That is, IDP attempts to match traffic against all rules in the Exempt rulebase and executes all matches.

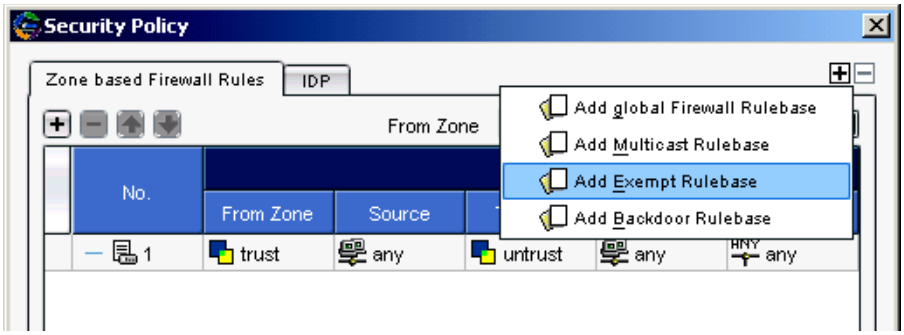
Adding the Exempt Rulebase

Before you can configure a rule in the Exempt rulebase, you need to add the Exempt rulebase to a security policy with the following steps:

1. In the main navigation tree, select **Security Policies**. Open a security policy either by double-clicking on the policy name in the Security Policies window, or by clicking on the policy name, then selecting the Edit icon.

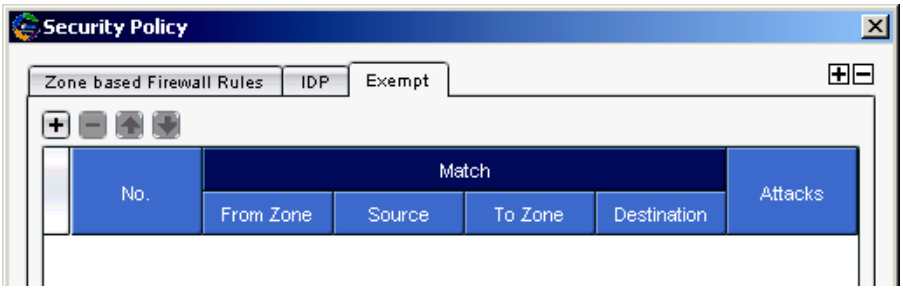
2. Click the Add icon in the upper right corner of the Security Policy window, then select **Add Exempt Rulebase**.

Figure 70: Adding an Exempt Rulebase



The Add Exempt Rulebase tab appears.

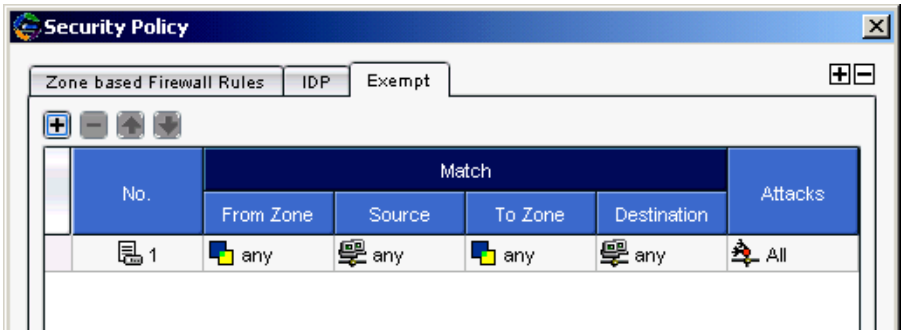
Figure 71: Exempt Rulebase Added



3. To configure an exempt rule, click the Add icon on the left side of the Security Policy window.

   A default exempt rule appears. You can modify this rule as needed.

Figure 72: Exempt Rule Added



Defining a Match

Specify the traffic you want to exempt from attack detection. The Match columns From Zone, Source, To Zone, and Destination are required for all rules in the exempt rulebase.

The following sections detail the Match columns of an exempt rule.

Source and Destination Zones

You can select multiple zones for the source and destination, however these zones must be available on the devices on which you will install the policy. You can specify “any” for the source or destination zones to monitor network traffic originating from or destined for any zone.

NOTE: You can create custom zones for some security devices. The list of zones from which you can select source and destination zones includes the predefined and custom zones that have been configured for all devices managed by NetScreen-Security Manager. Therefore, you should only select zones that are applicable for the device on which you will install the security policy.

Source and Destination Address Objects

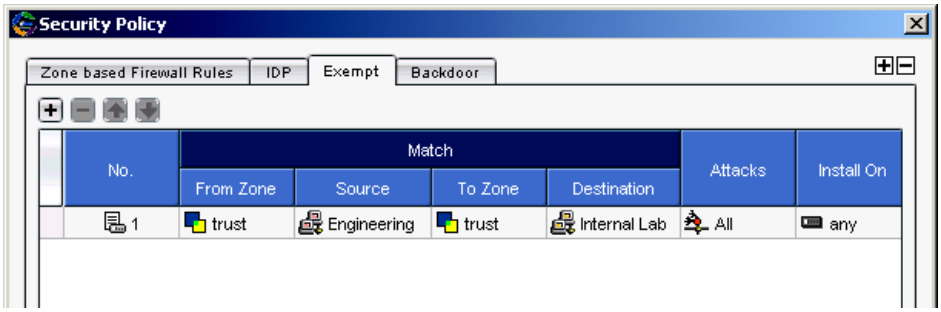
In the NetScreen-Security Manager system, address objects are used to represent components on your network: hosts, networks, servers, and so on. You can specify “any” to monitor network traffic originating from any IP address. You can also negate the address object(s) listed in the Source or Destination column of a rule to specify all sources or destinations except the excluded object.

You can create address objects either before you create an exempt rule (refer to the *NetScreen-Security Manager Administrator’s Guide)* or while creating or editing an exempt rule. To select or configure an address object, right-click on either the Source or Destination column of a rule, then select **Select Address**. In the Select Source Addresses dialog box, you can either select an already created address object, or you can click the Add icon to create a new host, network, or group object.

Example: Exempting a Source/Destination Pair

To improve performance and eliminate false positives between your Internal Lab devices and your Engineering desktops, you want to exempt attack detection. Your exempt rule looks similar to Figure 73:

Figure 73: Exempting Source and Destination



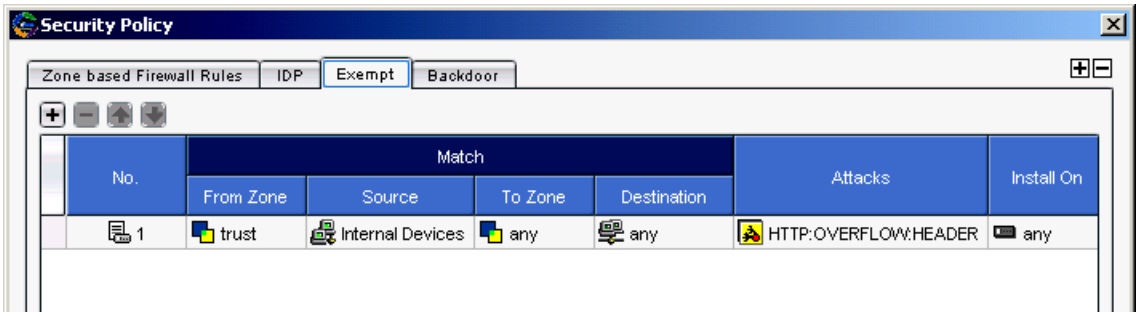
Setting Attack Objects

You specify the attack(s) you want IDP to exempt for the specified source/destination addresses. You must include at least one attack object in an exempt rule.

Example: Exempting Specific Attack Objects

You consistently find that your security policy generates false positives for the attack HTTP Buffer Overflow: Header on your internal network. You want to exempt attack detection for this attack when the source IP is from your internal network. Your exempt rule looks similar to Figure 74:

Figure 74: Exempting Attack Object



Setting Targets

For each rule in the Exempt rulebase, you can select the IDP-capable device that will use that rule to detect and prevent attacks. When you install the security policy to which the rule belongs, the rule becomes active only on the device(s) you select in the Install On column of the rulebase.

Entering Comments

You can enter notations about the rule in the Comments column. The information in the Comments column is not pushed to the target device(s). To enter a comment, right-click on the Comments column, then select **Edit Comments**. The Edit Comments dialog box appears. You can enter a comment of up to 1024 characters.

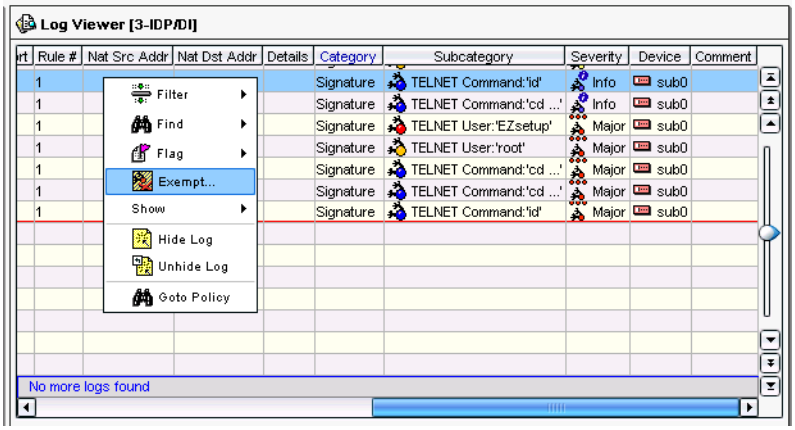
Creating an Exempt Rule from the Log Viewer

You can also create a rule in the Exempt rulebase directly from the NetScreen-Security Manager Log Viewer. You might want to use this method to quickly eliminate rules that generate false positive log records. (For more information about viewing IDP logs, see “Managing IDP” on page 222. For more information about using the Log Viewer, refer to the *NetScreen-Security Manager Administrator’s Guide*.)

To create an exempt rule from the Log Viewer, perform the following steps:

1. View the IDP/DI logs in the Log Viewer.
2. Right-click a log record that contains an attack you want to exempt, then select **Exempt**.

Figure 75: Exempting a Log Record Rule



The Exempt rulebase for the security policy that generated the log record is displayed, with the exempt rule that is associated with the log entry. The source, destination, and attack settings for the rule are automatically filled in based on the information in the log record.

NOTE: If the Exempt rulebase does not already exist when you create an exempt rule from the Log Viewer, the rulebase is automatically created and the rule is added.

You can modify, reorder, or merge an exempt rule created from the Log Viewer in the same manner as any other exempt rule that you create directly in the Exempt rulebase.

Configuring Backdoor Rules

A backdoor is a mechanism installed on a host computer that facilitates unauthorized access to a system. Attackers who have already compromised a system can install a backdoor to make future attacks easier. When attackers type commands to control a backdoor, they generate interactive traffic.

Unlike antivirus software, which scans for known backdoor or executable files on the host system, IDP detects the interactive traffic that is produced when backdoors are used. If interactive traffic is detected, IDP can perform IP actions against the connection to prevent an attacker from further compromising your network.

When you configure a backdoor rule, you must specify the following:

- Source and destination addresses for traffic you want to monitor. To detect incoming interactive traffic, set the Source to "any" and the Destination to the IP address of network device you want to protect. To detect outgoing interactive traffic, set the Source to the IP address of the network device you want to protect and the Destination to "any."

- Services that are offered by the Source or Destination as well as interactive services that can be installed and used by attackers.

NOTE: Do not include Telnet, SSH, RSH, NetMeeting, or VNC, as these services are often used to legitimately control a remote system and their inclusion might generate false positives.

- Action that the IDP is to perform if interactive traffic is detected. Set the Operation to "detect." If you are protecting a large number of network devices from interactive traffic, you can create a rule that ignores accepted forms of interactive traffic from those devices, then create another rule that detects all interactive traffic from those devices.

NOTE: The Backdoor rulebase is a terminal rulebase. That is, when IDP finds a match on a rule in the Backdoor rulebase, it does not execute successive rules.

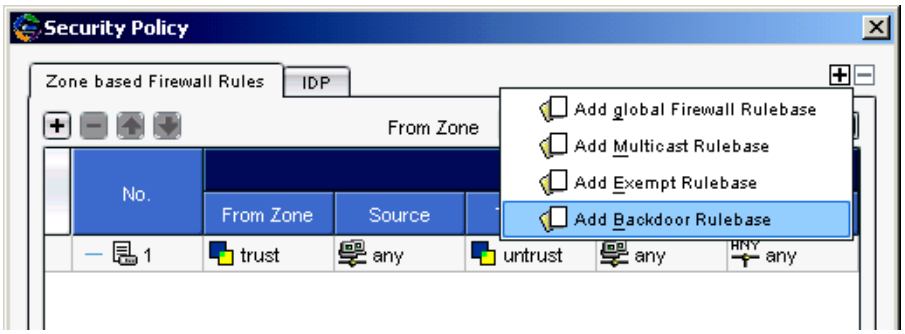
Adding the Backdoor Rulebase

Before you can configure a rule in the Backdoor rulebase, you need to add the Backdoor rulebase to a security policy with the following steps:

1. In the main navigation tree, select **Security Policies**. Open a security policy either by double-clicking on the policy name in the Security Policy window, or by clicking on the policy name, then selecting the Edit icon.

2. To configure a backdoor rule, click the Add icon in the upper right corner of the Security Policy window (see Figure 76).

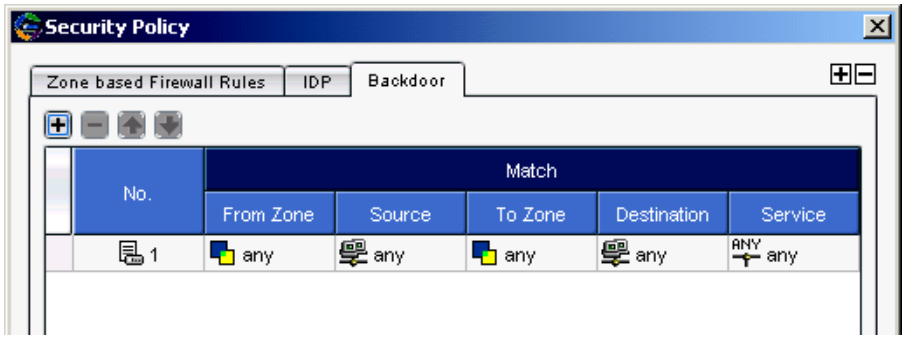
Figure 76: Adding the Backdoor Rulebase



3. Select **Add Backdoor Rulebase**.

   A default backdoor rule appears as shown in Figure 77. You can modify this rule as needed.

Figure 77: Backdoor Rule Added



Defining a Match

You specify the traffic you want IDP to monitor for indications of backdoors or Trojans. The Match columns From Zone, Source, To Zone, Destination, and Service are required for all rules in the Backdoor rulebase.

The following sections detail the Match columns of a backdoor rule.

Source and Destination Zones

You can select multiple zones for the source and destination. However, these zones must be available on the security devices on which you will install the policy. You can specify “any” for the source or destination zones to monitor network traffic originating from or destined for any zone.

NOTE: You can create custom zones for some security devices. The list of zones from which you can select source and destination zones includes the predefined and custom zones that have been configured for all devices managed by NetScreen-Security Manager. Therefore, you should only select zones that are applicable for the device on which you will install the security policy.

Source and Destination Address Objects

In the NetScreen-Security Manager system, address objects are used to represent components on your network: hosts, networks, servers, and so on. Typically, a server or other device on your network is the destination IP for incoming attacks and it can sometimes be the source IP for interactive attacks. You can specify "any" to monitor network traffic originating from any IP address. You can also negate the address object(s) listed in the Source or Destination column to specify all sources or destinations except the excluded address object.

You can create address objects either before you create a backdoor rule (refer to the *NetScreen-Security Manager Administrator's Guide)* or while creating or editing a backdoor rule. To select or configure an address object, right-click on either the Source or Destination column of a rule, then select **Select Address**. In the Select Source Addresses dialog box, you can either select an already created address object or you can click the **Add** icon to create a new host, network, or group object.

Services

Select interactive service objects. Be sure to include services that are offered by the source or destination IP as well as interactive services that are not; attackers can use a backdoor to install any interactive service. Do not include Telnet, SSH, RSH, NetMeeting, or VNC, as these services are often used to control a remote system legitimately, and their inclusion might generate false positives.

Setting the Operation

Set the Operation to **detect** or **ignore**. If you select **detect**, choose an action to perform if backdoor traffic is detected. If you are protecting a large number of address objects from interactive traffic, you can create a rule that ignores accepted forms of interactive traffic from those objects, then create a succeeding rule that detects all interactive traffic from those objects.

Setting Actions

Use the following steps to configure an action to perform if IDP detects interactive traffic:

Table 18: Actions for Backdoor Rule

Action	Description
Accept	IDP accepts the interactive traffic.
Drop Connection	IDP drops the interactive connection without sending an RST packet to the sender, preventing the traffic from reaching its destination. Use this action to drop connections for traffic that is not prone to spoofing.
Close Client and Server	IDP closes the interactive connection and sends an RST packet to both the client and the server. If the IDP is in sniffer mode, IDP sends an RST packet to both the client and server but does not close the connection.
Close Client	IDP closes the interactive connection to the client but not to the server.
Close Server	IDP closes the interactive connection to the server but not to the client.

Setting Notification

The first time you design a security policy, you might be tempted to log all attacks and let the policy run indefinitely. Don't do this! Some attack objects are informational only, and others can generate false positives and redundant logs. If you become overloaded with data, you might miss something important. Remember that security policies that generate too many log records are hazardous to the security of your network, as you might discover an attack too late or miss a security breach entirely if you have to sift through hundreds of log records. Excessive logging can also affect IDP throughput, performance, and available disk space. A good security policy generates enough logs to fully document only the important security events on a network.

Setting Logging

In the Configure Notification dialog box, select **Logging**, then click **OK**. Each time the rule is matched, the IDP system creates a log record that appears in the Log Viewer.

Logging an attack creates a log record that you can view real time in the Log Viewer. For more critical attacks, you can also set an alert flag to appear in the log record, notify you immediately by email, have IDP run a script in response to the attack, or set an alarm flag to appear in the log record. Your goal is to fine tune the attack notifications in your security policy to your individual security needs.

To log an attack for a rule, right-click the Notification column of the rule, then select **Configure**. The Configure Notification dialog box appears.

Setting an Alert

In the Configure Notification dialog box, select **Alert**, then click **OK**. If Alert is selected and the rule is matched, IDP places an alert flag in the Alert column of the Log Viewer for the matching log record.

Logging Packets

You can record the individual packets in network traffic that matched a rule by capturing the packet data for the attack. Viewing the packets used in an attack on your network can help you determine the extent of the attempted attack and its purpose, whether or not the attack was successful, and any possible damage to your network.

NOTE: To improve IDP performance, log only the packets following the attack.

If multiple rules with packet capture enabled match the same attack, IDP captures the maximum specified number of packets. For example, you configure rule 1 to capture 10 packets before and after the attack, and you can configure rule 2 to capture 5 packets before and after the attack. If both rules match the same attack, IDP attempts to capture 10 packets before and after the attack.

NOTE: Packet captures are restricted to 256 packets before and after the attack.

Setting Severity

You can override the inherent attack severity on a per-rule basis within the Backdoor rulebase. You can set the severity to Default, Info, Warning, Minor, Major, or Critical.

To change the severity for a rule, right-click the Severity column, then select a severity.

Setting Targets

For each rule in the Backdoor rulebase, you can select the security device that will use that rule to detect and prevent attacks. When you install the security policy to which the rule belongs, the rule becomes active only on the devices you select in the Install On column of the rulebase.

Entering Comments

You can enter notations about the rule in the Comments column. The information in the Comments column is not pushed to the target device(s). To enter a comment, right-click on the Comments column, then select **Edit Comments**. The Edit Comments dialog box appears. You can enter a comment of up to 1024 characters.

Configuring IDP Attack Objects

Attack objects contain patterns for known attacks that attackers can use to compromise your network. Attack objects do not work on their own—they need to be part of a rule before they can start detecting known attacks and preventing malicious traffic from entering your network. To use attack objects in your IDP rules, add the IDP rulebase in your security policy, then add an IDP rule to the rulebase. See “Configuring Security Policies” on page 173.

NOTE: IDP is supported only on security devices with IDP capabilities.

About IDP Attack Object Types

In a security policy, you can select the attack object that a device uses to detect attacks against your network. If an attack is detected, the device generates an attack log entry that appears in the Log Viewer. For more information, see “Configuring IDP Rules” on page 178.

NetScreen-Security Manager supports three types of IDP attack objects:

- Signature attack objects
- Protocol anomaly attack objects
- Compound attack objects

The following sections detail each attack object type.

Signature Attack Objects

An attack *signature* is a pattern that always exists within an attack; if the attack is present, so is the attack signature. IDP uses *stateful signatures* to detect attacks. Stateful signatures are more specific than regular signatures. With stateful signatures, IDP can look for the specific protocol or service used to perpetrate the attack, the direction and flow of the attack, and the context in which the attack occurs. ScreenOS combines the attack pattern with service, context, and other information into the attack object. Stateful signatures produce few false positives because the context of the attack is defined, eliminating huge sections of network traffic in which the attack would not occur.

Protocol Anomaly Attack Objects

Protocol anomaly attack objects detect abnormal or ambiguous messages within a connection according to the set of rules for the particular protocol being used. Protocol anomaly detection works by finding deviations from protocol standards, most often defined by RFCs and common RFC extensions. Most legitimate traffic adheres to established protocols. Traffic that does not produces an anomaly, which may be created by attackers for specific purposes, such as evading an IPS.

Compound Attack Objects

A compound attack object combines multiple signatures and/or protocol anomalies into a single object. Traffic must match all of the combined signatures and/or protocol anomalies to match the compound attack object; you can specify the order in which signatures or anomalies must match. Use compound attack objects to refine your security policy rules, reduce false positives, and increase detection accuracy.

A compound attack object enables you to be very specific about the events that need to occur before IDP identifies traffic as an attack. For example, you might want to take action only if an FTP session includes a failed login attempt for specific users.

Viewing Predefined IDP Attack Objects and Groups

Juniper Networks provides predefined attack objects and attack object groups that you can use in security policies to match traffic against known attacks. Juniper Networks regularly updates the predefined attack objects and groups. While you cannot create, edit, or delete predefined attack objects, you can update the list of attack objects that you can use in security policies. The revised attack objects and groups are available as part of an attack database update, which is downloaded to the NetScreen-Security Manager GUI Server. See “Managing IDP” on page 222 for information about attack-database updates.

To view predefined attack objects and groups perform the following steps:

1. In the Object Manager, click **Attack Objects > IDP Objects**. The IDP Objects dialog box appears.

2. Click the Predefined Attacks or Predefined Attack Groups tab to view predefined attack objects or groups.

Viewing Predefined Attacks

The Predefined Attacks tab displays all attacks in a table format and includes the following information:

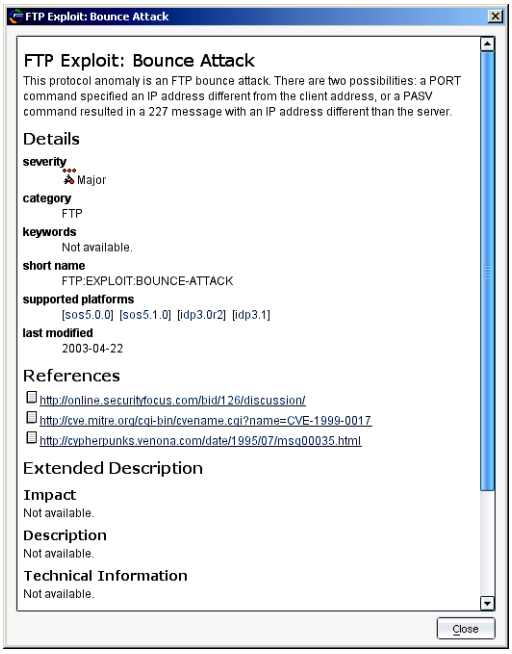
- Name of the attack object
- Severity of the attack: critical, major, minor, warning, info
- Category
- Keywords for the attack
- CVE number, which identifies the attack's number in the Common Vulnerabilities and Exposures database
- Bugtraq number, which identifies the equivalent attack in the Security Focus Bugtraq database

Initially, attack objects are listed alphabetically by Category name but you can view attacks in different orders by clicking on a column heading.

To locate all rules that use a predefined attack object, right-click the attack object, then select **View Usages**.

To display a detailed description of an attack object, double-click on the attack.

Figure 78: Attack Viewer



Viewing Predefined Groups

The Predefined Attack Group tab displays the following predefined attack groups:

- **Category** groups attack objects by predefined categories. Within each category, attack objects are grouped by severity.
- **Operating System** groups attack objects by the operating system to which they apply: BSD, Linux, Solaris, or Windows. Within each operating system, attack objects are grouped by services and severity.
- **Severity** groups attack objects by the severity assigned to the attack. IDP has five severity levels: Info, Warning, Minor, Major, Critical. Within each severity, attack objects are grouped by category.

To locate all rules that use a predefined attack object group, right-click the attack object group, then select **View Usages**.

To display a detailed description of an attack object, double-click on the attack.

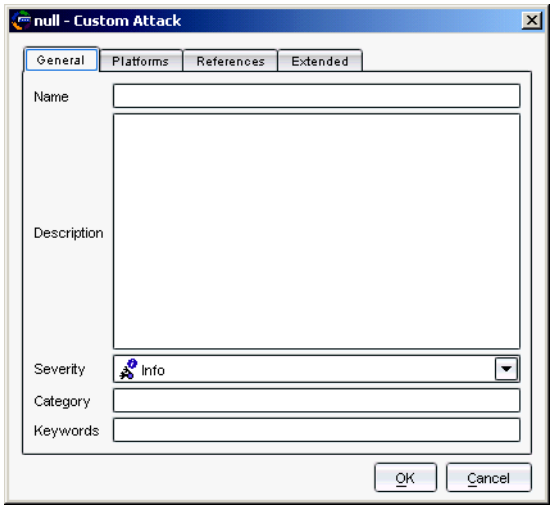
Creating Custom IDP Attack Objects

You can create custom attack objects to detect new attacks or otherwise meet the unique needs of your network.

To create a custom attack object, perform the following steps:

1. In the Object Manager, click **Attack Objects > IDP Objects**. The IDP Objects dialog box appears.
2. Click the **Custom Attacks** tab.
3. Click the **Add** icon. The Custom Attack dialog box appears with the General tab selected.

Figure 79: Custom Attack Dialog Box



   a. Enter a name for the attack. The name is used to display the attack object in the UI. You might want to include the protocol the attack uses as part of the attack name.

b. Enter a description for the attack. The description provides details about the attack. Entering a description is optional when creating a new attack object, but it can help you remember important information about the attack. View the attack descriptions for predefined attacks for examples. To display details about a predefined attacks, see "Viewing Predefined Attacks" on page 203.

c. Select a severity for this attack: Info, Warning, Minor, Major, or Critical. Critical attacks are the most dangerous—typically these attacks attempt to crash your server or gain control of your network. Informational attacks are the least dangerous and typically are used by network administrators to discover holes in their own security system.

d. Enter a category for this attack. You can use a predefined category or define a new category

e. Enter one or more keywords for this attack that can help you find it later. A keyword is a unique identifier used to display the attack object in log records. Keywords indicate the important words that relate to the attack and the attack object.

4. Click the Platforms tab in the Custom Attack dialog box to specify the security platform on which the attack detection is to occur.

   a. Click the Add icon. The Custom Attack dialog box appears.

   b. Select the platform on which the attack detection is to occur.

   c. Select the type of attack—Signature, Protocol Anomaly, or Compound Attack—then click **Next**.

If you are configuring a new attack object, the attack object editor leads you through the screens to configure the specific type of attack:

- For signature attack objects, see the following section, "Creating a Signature Attack Object".
- For protocol anomaly attack objects, see "Protocol Anomaly Attack Objects" on page 202.
- For compound attack objects, see "Compound Attack Objects" on page 202.

Creating a Signature Attack Object

To configure a signature attack in the Custom Attack dialog box, perform the following steps:

1. Configure general parameters for the attack perform the following steps:

   - **False-Positives** indicates the frequency (Unknown, Rarely, Occasionally, Frequently) that the attack object produces a false positive when used in a security policy. By default, all compound attack objects are set to Unknown, as you fine tune your IDP system to your network traffic, you can change this setting to help you track false positives.

- **Service Binding** allows you to select a protocol that the attack uses to enter your network. Depending upon the protocol you select, additional fields might appear. You can select the following protocol types:
  - **Any** allows IDP to match the signature in all services (attacks can use multiple services to attack your network).
  - **IP** (specify protocol number) allows IDP to match the signature for a specified IP protocol type.
  - **TCP** (specify port ranges) allows IDP to match the signature for specified TCP port(s).
  - **UDP** (specify port ranges) allows IDP to match the signature for specified UDP port(s).
  - **ICMP** (specify ID) allows IDP to match the signature for specified ICMP ID.
  - **RPC** (specify program number) allows IDP to match the signature for a specified remote procedure call program number.
  - **Service** (specify service) allows IDP to match the signature for a specified service.
- **Time Binding** allows IDP to detect a sequence of the same attacks over a period of time. If you select **Time Binding**, you can specify the following attributes which are bound to the attack object for one minute:
  - **Scope** specifies whether the counting of the attack is from the same source IP address, the same destination IP address, or a peer. If you select **Source**, IDP detects attacks from a given source IP address for the specified number of times, regardless of the destination IP address. If you select **Destination**, IDP detects attacks to a given destination IP address for the specified number of times, regardless of the source IP address. If you select **Peer**, IDP detects attacks between source and destination IP addresses of the sessions for the specified number of times.
  - **Count** specifies the number of times that IDP detects the attack within the specified scope before triggering an event.

2. Click **Next**.
3. Configure detection parameters for the signature attack:
   - The attack pattern is the signature of the attack you want to detect. The signature is a pattern that always exists within an attack; if the attack is present, so is the signature. When creating a new signature attack object, you must analyze the attack to detect a pattern (such as a segment of code, a URL, or a value in a packet header) and then use that pattern to create a signature. You can also negate an entered pattern.

     Table 19 shows the regular expressions you can use in the attack pattern:

Table 19: Attack Pattern Expressions

Regular Expression	Description
\0<*octal-number*>	For a direct binary match
\X<*hexadecimal-number*>\X	For a direct binary match
\[<*character-set*>\]	For case-insensitive matches
.	To match any symbol
*	To match 0 or more symbols
+	To match 1 or more symbols
?	To match 0 or 1 symbols
( )	Grouping of expressions
|	Denotes alternate, typically used with ( )
[<*start*>-<*end*>]	Character range
[^<*start*>-<*end*>]	Negation of range

- **Offset** is the starting place from the specified service context where IDP should look for the attack. If there is no offset, you can specify **None**; otherwise, you can specify a decimal value.

- **Context** defines the location where IDP should look for the attack in a specific Application-Layer protocol. When creating a signature attack object, you should choose a service context, if possible. Because the service context is very specific, your chances of detecting a false positive are greatly reduced. However, choosing a service context overrides any protocol you previously specified in the Service Binding general parameter.

  Table 20 lists the service contexts you can use for the attacks.

Table 20: Service Context for Signature Attacks

Service	Description	RFC
AIM	AOL Instant Messenger	
DHCP	Dynamic Host Configuration Protocol	2131, 2132
DNS	Domain Name System	1034, 1035
Finger	Finger Information Protocol	1128
FTP	File Transfer Protocol	959
Gnutella	Gnutella	
Gopher	Gopher	1436
HTTP	Hypertext Transfer Protocol	2616
IMAP	Internet Message Access Protocol	2060
IRC	Internet Relay Chat	2810, 2811, 2812, 2813
LDAP	Lightweight Directory Access Protocol	2251, 2252, 2253, 3377
LPR	Line Printer Protocol	1179
MSN	Microsoft Instant Messenger	
NBNAME/ NBDS	NetBios Name Service	1001, 1002
NFS	Network File System	
NNTP	Network News Transfer Protocol	977
NTP	Network Time Protocol	1305
POP3	Post Office Protocol, version 3	1081
RADIUS	Remote Authentication Dial-In User Service	2865, 2866, 2867, 2868, 3575
REXEC		
RLOGIN	Remote Login (rlogin)	1258, 1282
RSH	Remote Shell (rsh)	
RUSERS		
SMB	Server Message Block	
SMTP	Simple Mail Transfer Protocol	821
SNMP	Simple Network Management Protocol	1067
SNMPTRAP	SNMP trap	1067
SSH	Secure Shell	Proprietary
SSL	Secure Sockets Layer	
Telnet	Telnet TCP protocol	854
TFTP	Trivial File Transfer Protocol	783
VNC	Virtual Network Computing	
YMSG	Yahoo! Messenger	

- **Direction** defines the connection direction of the attack:
  - **Client to Server** detects the attack only in client-to-server traffic.

- **Server to Client** detects the attack only in server-to-client traffic.
- **Any** detects the attack in either direction.

- **Flow** defines the connection flow of the attack: Control, Auxiliary, or Both.

4. Click **Next**.

5. Configure the header match information for the signature attack. The header match configuration allows you to specify that IDP search a packet for a pattern match only for certain header information for the following protocols:

   - Internet Protocol (IP)
     - **Type-of-service**. Specify an operand (none, =, !, >, <) and a decimal value.
     - **Total length**. Specify an operand (none, =, !, >, <) and a decimal value.
     - **ID**. Specify an operand (none, =, !, >, <) and a decimal value.
     - **Time-to-live**. Specify an operand (none, =, !, >, <) and a decimal value.
     - **Protocol**. Specify an operand (none, =, !, >, <) and a decimal value.
     - **Source**. Specify the IP address of the attacking device.
     - **Destination**. Specify the IP address of the attack target.
     - **Reserved Bi**t. Specifies that IDP looks for a pattern match whether or not the IP flag is set (**none**), only if the flag is set (**set**), or only if the flag is not set (**unset**).
     - **More Fragments**. Specifies that IDP looks for a pattern match whether or not the IP flag is set (**none**), only if the flag is set (**set**), or only if the flag is not set (**unset**).
     - **Don't Fragment**. Specifies that IDP looks for a pattern match whether or not the IP flag is set (**none**), only if the flag is set (**set**), or only if the flag is not set (**unset**).
   - Transmission Control Protocol (TCP)
     - **Source Port**. The port number on the attacking device.
     - **Destination Port**. The port number of the attack target.
     - **Sequence Number**. The sequence number of the packet. This number identifies the location of the data in relation to the entire data sequence.

- **ACK Number**. The ACK number of the packet. This number identifies the next sequence number; the ACK flag must be set to activate this field.
- **Header Length**. The number of bytes in the TCP header.
- **Window Size**. The number of bytes in the TCP window size.
- **Urgent Pointer**. Indicates that the data in the packet is urgent; the URG flag must be set to activate this field.
- **Data Length**. The number of bytes in the data payload. For SYN, ACK, and FIN packets, this field should be empty.

You can also specify the following TCP flag options as **none**, **set**, or **unset**:

- **URG**. When set, the urgent flag indicates that the packet data is urgent.
- **ACK**. When set, the acknowledgment flag acknowledges receipt of a packet.
- **PSH**. When set, the push flag indicates that the receiver should push all data in the current sequence to the destination application (identified by the port number) without waiting for the remaining packets in the sequence.
- **RST**. When set, the reset flag resets the TCP connection, discarding all packets in an existing sequence.
- **FIN**. When set, the final flag indicates that the packet transfer is complete and the connection can be closed.

- User Datagram Protocol (UDP)
  - **Source Port**. The port number on the attacking device. Specify an operand (none, =, !, >, <) and a decimal value.
  - **Destination Port**. The port number of the attack target. Specify an operand (none, =, !, >, <) and a decimal value.
  - **Data Length**. The number of bytes in the data payload. Specify an operand (none, =, !, >, <) and a decimal value.
- Internet Control Message Protocol
  - **ICMP Type**. The primary code that identifies the function of the request/reply.
  - **ICMP Code**. The secondary code that identifies the function of the request/reply within a given type.
  - **Sequence Number**. The sequence number of the packet. This number identifies the location of the request/reply in relation to the entire sequence.

- **ICMP ID**. The identification number is a unique value used by the destination system to associate requests and replies.
- **Data Length**. The number of bytes in the data payload.

6. Click **Finish**.

Creating a Protocol Anomaly Attack

Perform the following steps to configure a protocol anomaly attack in the Custom Attack dialog box:

1. Configure general parameters for the attack:
   - **False-Positives** indicates the frequency (**Unknown**, **Rarely**, **Occasionally**, **Frequently**) that the attack object produces a false positive when used in a security policy. As you finetune your IDP system to your network traffic, you can change this setting to help you track false positives.
   - **Anomaly** allows you to select a protocol anomaly from a list of known protocol anomalies. NetScreen-Security Manager detects anomalies for the following protocols:

AIM	DHCP	IDENT	RUSERS	TFTP
FINGER	CHARGEN	IMAP	Gnutella	RLOGIN
FTP	DISCARD	IP Packet	Gopher	RPC
HTTP	DNS	POP3	IRC	RSH
ICMP	ECHO	REXEC	MSN	RTSP
MSN	LPR	NFS	VNC	NNTP
SNMP	SMTP	SMB	SNMP TRAP	YMSG
TCP segment	SYSLOG	SSH	TELNET	

   - **Time Binding** allows IDP to detect a sequence of the same attacks over a specified period. If you select **Time Binding**, you can specify the following attributes that are bound to the attack object for one minute:
     - **Scope** specifies whether the counting of the attack is from the same source IP address, the same destination IP address, or a peer. If you select **Source**, IDP detects attacks from a given source IP address for the specified number of times, regardless of the destination IP address. If you select **Destination**, IDP detects attacks to a given destination IP address for the specified number of times, regardless of the source IP address. If you select **Peer**, IDP detects attacks between source and destination IP addresses of the sessions for the specified number of times.
     - **Count** specifies the number of times that IDP detects the attack within the specified scope before an event is triggered.

2. Click **Finish**.

Creating a Compound Attack

Note the following when creating a custom compound attack object:

- All members of the compound attack object must use the same service setting or service binding, such as FTP, Telnet, YMSG, TCP/80, and so on.
- You cannot add predefined or custom signature attack objects to a compound attack object. Instead, you specify the signature directly within the compound attack object, including such details as service (or service binding), service context, attack pattern, and direction. You can add protocol anomaly attack objects to a compound attack object.
- You can add between 2 and 32 protocol anomaly attack objects and/or signatures as members of the compound attack object. However, all members must use the same service setting or service binding.

Perform the following steps to configure a compound attack in the Custom Attack dialog box:

1. Configure general parameters for the attack:
    - **False-Positives** indicates the frequency (**Unknown**, **Rarely**, **Occasionally**, **Frequently**) that the attack object produces a false positive when used in a security policy. By default, all compound attack objects are set to **Unknown**. As you finetune IDP to your network traffic, you can change this setting to help you track false positives.
    - **Service Binding** allows you to select a protocol that the attack uses to enter your network. Depending upon the protocol you select, additional fields may appear. You can select the following protocol types:
        - **Any** allows IDP to match the signature in all services (attacks can use multiple services to attack your network).
        - **IP** (specify protocol number) allows IDP to match the signature in a specified IP protocol type.
        - **TCP** (specify port ranges) allows IDP to match the signature for specified TCP port(s).
        - **UDP** (specify port ranges) allows IDP to match the signature for specified UDP port(s).
        - **ICMP** (specify ID) allows IDP to match the signature for specified ICMP ID.
        - **RPC** (specify program number) allows IDP to match the signature for a specified remote procedure call program number.
        - **Service** (specify service) allows IDP to match the signature for a specified service.
    - **Time Binding** allows IDP to detect a sequence of the same attacks over a specified period. If you select **Time Binding**, you can specify the following attributes that are bound to the attack object for one minute:

- **Scope** specifies whether the counting of the attack is from the same source IP address, the same destination IP address, or a peer. If you select **Source**, IDP detects attacks from a given source IP address for the specified number of times, regardless of the destination IP address. If you select **Destination**, IDP detects attacks to a given destination IP address for the specified number of times, regardless of the source IP address. If you select **Peer**, IDP detects attacks between source and destination IP addresses of the sessions for the specified number of times.

- **Count** specifies the number of times that IDP detects the attack within the specified scope before an event is triggered.

2. Click **Next**.

3. Perform the following steps to configure the compound attack members:

   - **Scope** specifies whether the match should occur over a single session or can be made across multiple transactions within a session:

     - Select **Session** to allow multiple matches for the object within the same session.

     - Select **Transaction** to match the object across multiple transactions that occur within the same session.

   - Select **Reset** if the compound attack should be matched more than once within a single session or transaction. If **Reset** is selected, multiple matches can be made within a single session or transaction.

   - Select **Ordered Match** to create a compound attack object that must match each member signature or protocol anomaly in the order you specify. If you do not specify an ordered match, the compound attack object still must match all members, but the attacks or protocol anomalies can appear in random order.

You can now add signature or protocol anomaly attack objects to the compound attack, as described in the following sections.

Adding a Signature to the Compound Attack Object

1. To add an attack pattern to the compound attack object, click the Add icon, then select **Signature**. The New Member dialog box appears.

2. Double-click the newly created signature member of the compound attack object. Configure the attack pattern settings:

   - **DFA Pattern**. Specify the pattern IDP should match. You construct the attack pattern just as you would when creating a new signature attack object.

     To exclude the specified pattern from being matched, select the Negate checkbox.

   - **Context.** Specify the context in which the IDP should look for the pattern. The context displays only contexts that are appropriate for the specified service. If you selected a service binding of **any**, you are restricted to the service contexts packet and first packet.

   - **Direction**. Specify whether IDP should match the pattern in traffic flowing in any direction, from client-to-server, or from server-to-client.

     Examine the traffic before you determine the direction. Juniper Networks recommends client-to-server direction for better performance. There is a performance hit on the device if you select server-to-client and the risk of attack objects is lower with client-to-server.

3. Click **OK**.

Adding a Protocol Anomaly to the Compound Attack Object

1. To add a protocol anomaly to the compound attack object, click the Add icon, then select **Anomaly**. The New Member dialog box appears.

2. Select an anomaly.

3. Click **OK**.

Deleting a Member from the Compound Attack Object

To remove a member signature or an anomaly, select the member in the list, then click the Delete icon. A confirmation window asks you to verify that you want to delete the item. Click **OK**.

Editing a Custom Attack Object

To modify a custom attack object, double-click the object in the Custom Attacks tab in the IDP Objects dialog box. The Custom Attacks dialog box appears with the previously configured information in the General and Platforms tabs. You can enter optional information in the References and Extended tabs. Enter any changes you want to make, then click **Apply**. To close the dialog box, click **OK**.

Deleting a Custom Attack Object

To delete a custom attack object, right-click the object in the Custom Attacks tab in the IDP Objects dialog box, then select **Delete**. A confirmation window asks you to verify that you want to delete the item. Click **OK**.

Creating Custom IDP Attack Groups

The IDP system contains hundreds of predefined attack objects, and you can create additional custom attack objects. When you create your security policy rules, you can add attack objects individually or by the predefined or the custom attack group. To help keep your security policies organized, you can organize attack objects into groups.

You can create *static groups*, which contain only the groups or attack objects you specify, or *dynamic groups*, which contain attack objects based on criteria you specify.

Configuring Static Groups

A static group contains a specific, finite set of attack objects or groups. There are two types of static groups: *predefined* static groups and *custom* static groups.

A predefined static group can include the following members:

- Predefined attack objects
- Predefined static groups
- Predefined dynamic groups

A custom static group can include the same members as a predefined static group, plus the following members:

- Custom attack objects
- Custom dynamic groups
- Other custom static groups

You use static groups to do the following:

- Define a specific set of attacks to which you know your network is vulnerable
- Group custom attack objects
- Define a specific set of informational attack objects that you use to keep you aware of what is happening on your network

Static groups require more maintenance than dynamic groups because you must manually add or remove attack objects in a static group in order to change the members. However, you can include a dynamic group within a static group to automatically update some attack objects. For example, the predefined attack object group Operating System is a static group that contains four predefined static groups: BSD, Linux, Solaris, and Windows. The BSD group contains the predefined dynamic group BSD-Services-Critical, to which attack objects can be added during an attack database update.

To create a custom static group:

1. In Object Manager, click **Attack Objects > IDP Objects**. The IDP Objects dialog box appears.
2. Click the Custom Attack Groups tab.
3. Click the Add icon, then select **Add Static Group**. The New Static Group dialog box appears.
4. Enter a name and description for the static group. Select a color for the group icon.
5. To add an attack or a group to the static group, select the attack or group from the Attacks/Group list, then click **Add**.
6. Click **OK**.

Configuring Dynamic Groups

A dynamic group contains a dynamic set of attack objects that are automatically added or deleted based on specified criteria for the group. For example, an attack database update can add or remove attack objects from a dynamic group based on the group criteria. This eliminates the need for you to review each new signature to determine if you need to use it in your existing security policy.

A predefined or custom dynamic group can only contain attack objects, not attack groups. Dynamic group members can be either predefined or custom attack objects.

Perform the following steps to create a custom dynamic group:

1. In the Object Manager, click **Attack Objects > IDP Objects**. The IDP Objects dialog box appears.
2. Click the Custom Attack Groups tab.
3. Click the Add icon, then select **Add Dynamic Group**. The New Dynamic Group dialog box appears.
4. Enter a name and description for the static group. Select a color for the group icon.
5. In the Filters tab, click the Add icon, then select one of the following:
   - **Add Products Filter** adds attack objects based on the application that is vulnerable to the attack
   - **Add Severity Filter** adds attack objects based on the attack severity.

NOTE: All predefined attack objects are assigned a severity level by Juniper Networks. However, you can edit this setting to match the needs of your network.

   - **Add Category Filter** adds attack objects based on category

- **Add Last Modified Filter** adds attack objects based on their last modification date

You create filters one at a time; as you add each criteria, IDP compares it to the attributes for each attack object and immediately filters out any attack object that does not match. If you create a filter with attributes that no attack object can match, a message appears warning you that your dynamic group has no members.

From the resulting list of matching attack objects, you can then exclude any attack objects that produce false positives on your network or that detect an attack to which your network is not vulnerable.

NOTE: A dynamic group cannot contain another group (predefined, static, or dynamic). However, you can include a dynamic group as a member of a static group.

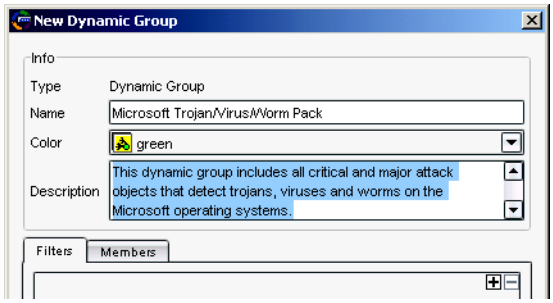
Example: Creating a Dynamic Group

Perform the following steps to create a dynamic group:

1. In the Custom Attack Groups tab, click the Add icon, then select **Add Dynamic Group**. The New Dynamic Group dialog box appears.

2. Enter a name and description for the group. Select a color for the group icon.

Figure 80: New Dynamic Group



3. In the Filters tab, click the Add icon, then add the filters that determine which attack objects should be in the group:

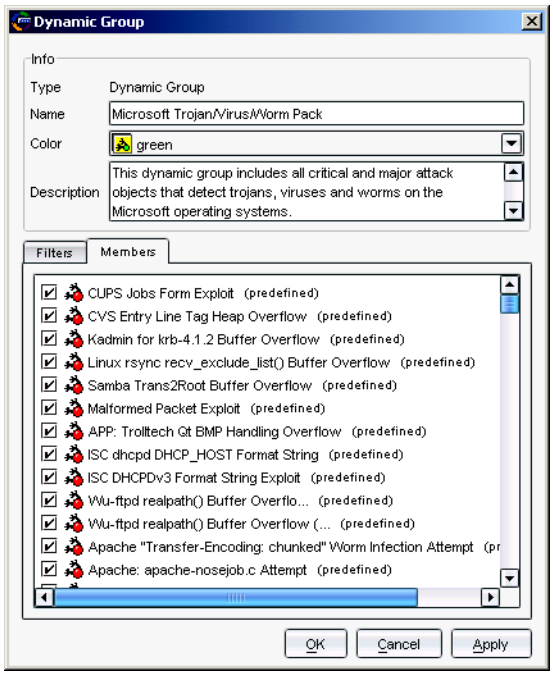
   a. Add a Products filter to add attack objects that detect attacks against all Microsoft Windows operating systems.

   b. Add a Severity filter to add attack objects that have a severity level of Critical or Major.

   IDP automatically applies all filters to the entire attack object database, identifies the attack objects that meet the defined criteria, and adds the matching objects as members of the group.

4. View the members of the group by clicking on the Members tab as shown in Figure 81:

Figure 81: New Dynamic Group Members



5. Click **OK** to save the dynamic group.

Updating Dynamic Groups

When you are satisfied with the group criteria and its members, use the group in a security policy. The next time you update your attack objects, the following tasks are performed automatically:

- For all new attack objects, the update compares the predefined attributes of each attack object to each dynamic group criteria and adds the attack objects that match.
- For all updated attack objects, the update removes attack objects that no longer meet their dynamic group criteria. The update also reviews updated attack objects to determine if they now meet any other dynamic group criteria and adds them to those groups as necessary.
- For all deleted attack objects, the update removes the attack objects from their dynamic groups.

You can also edit a dynamic group manually, adding new filters or adjusting existing filters to get the type of attack objects you want. You can also edit a dynamic group from within a security policy (see “Configuring Security Policies” on page 173).

Editing a Custom Attack Group

To modify a custom attack group, double-click the group in the Custom Attack Groups tab in the IDP Objects dialog box. The Static Group or Dynamic Group dialog box appears with the previously configured information displayed. Enter any changes you want to make, then click **Apply**; to close the dialog box, click **OK**.

Deleting a Custom Attack Group

To delete a custom attack group, right-click the group in the Custom Attack Groups tab in the IDP Objects dialog box, then select **Delete**. A confirmation window asks you to verify that you want to delete the item. Click **OK**.

Configuring the Device as a Standalone IDP Device

You can deploy the IDP-capable device as a standalone IDP security system protecting critical segments of your private network. For example, you might already have a security device actively screening traffic between the Internet and your private network (some devices can optionally use Deep Inspection to inspect this traffic). But you still need to protect internal systems, such as mail servers, from attacks that might originate from user machines in an otherwise trusted network. In this case, you need a security system that provides IDP instead of firewall functions.

This section describes how to configure the security device to provide standalone IDP functions.

NOTE: Juniper Networks offers standalone IDP appliances that provide IDP functionality without integrated firewall/VPN capabilities. You can use the NetScreen-Security Manager system to manage these appliances as well as IDP-capable firewall/VPN devices.

Enabling IDP

To enable IDP, you need to configure a firewall rule in a security policy that directs traffic between the applicable zones to be checked against IDP rulebases. You can make this firewall rule very simple in that it can match all traffic from all sources to all destinations for all services.

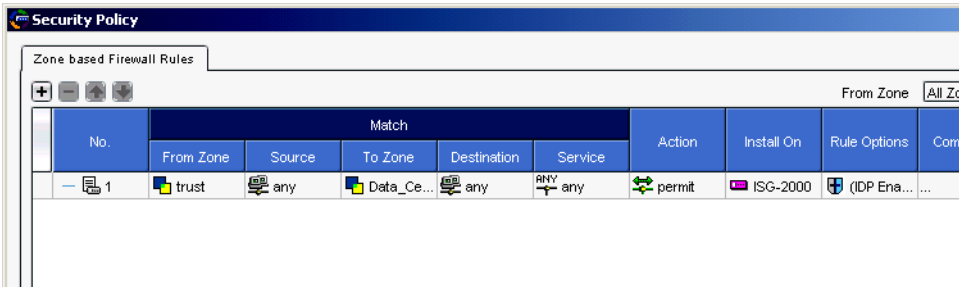
1. Create a firewall rule that permits traffic from any source to any destination for any service.
2. Right-click in the Rule Options column for the firewall rule, then select **DI Profile/Enable IDP**.
3. In the DI Profile/Enable IDP dialog box, click the button to enable IDP, then select **OK**.
4. Configure IDP rules, creating IDP rulebases as needed.

For more information about configuring security policies that include IDP rules, see “Configuring Security Policies” on page 173.

Example: Configuring a Firewall Rule for Standalone IDP

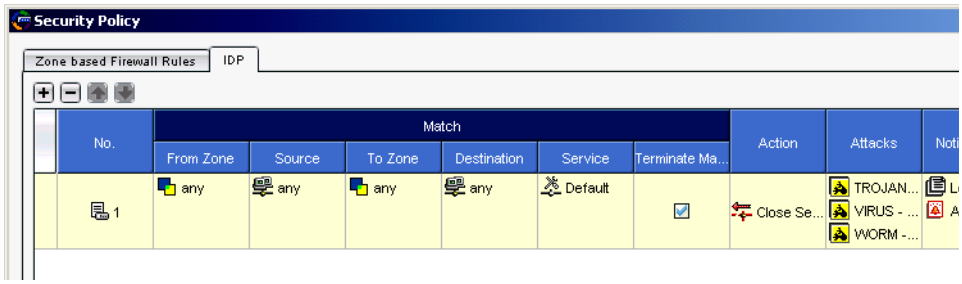
In this example, you are deploying an IDP/firewall/VPN device as a standalone IDP security system between the Trust zone and the custom Data_Center zone in your network. Your company's file, mail, and database servers reside in the Data_Center zone. While you want to allow users in the Trust zone to be able to access the servers in the Data_Center zone, you also need to protect the servers from attacks that inadvertently might have been introduced into a user machine in the Trust zone. You create a firewall rule from the Trust to the Data_Center zone that allows traffic from any source to any destination for any service, then enable IDP in the Rule Options column, as shown in Figure 82.

Figure 82: Firewall Rule for Standalone IDP



You would then add and configure IDP rulebases for the security policy to detect possible attacks against servers in the Data_Center zone, as shown in Figure 83.

Figure 83: IDP Rules for Standalone IDP



Configuring Role-Based Administration

NetScreen-Security Manager's role-based administration (RBA) allows the super administrator (superadmin) to create a custom role and administrator for the standalone IDP device. This gives the IDP administrator permission to perform only those tasks that are specific to configuring and administering IDP functions; the IDP administrator does not need to create, edit, delete, view, or update device configurations. When the IDP administrator logs into the NetScreen-Security Manager UI, he or she only sees the menus and options that are applicable to IDP.

Example: Configuring an IDP-Only Administrator

In this example, you (the superadmin) create a custom role and an IDP administrator who can only perform tasks that are specific to configuring and administering IDP on the standalone IDP device.

1. Log into the global domain as the superadmin. From the Menu bar, select **Tools > Manage Administrators and Domains**.

2. Click the Roles tab, then click the Add icon to create a role called **IDP_Only**. Select tasks that are specific for IDP configuration and administration, such as:

   - Attack Update
   - Create/View/Edit/Delete Policies
   - Create/View/Edit/Delete Backdoor and IDP Rulebases
   - View Firewall Rulebases
   - Create/Edit/Delete Shared Objects and Groups

   Select any other tasks that might be helpful for the IDP administrator; for example, you can select the options to view Jobs and the System Status Monitor.

3. Click **OK** in the New Role dialog box to return to the Manage Administrators and Domains dialog box.

4. Click the Administrators tab, then click the Add icon to create an administrator called **IDP_Administrator**. The New Admin dialog box appears with the General tab selected.

5. In the Name field, enter **IDP_Administrator**. You can enter contact information for the administrator.

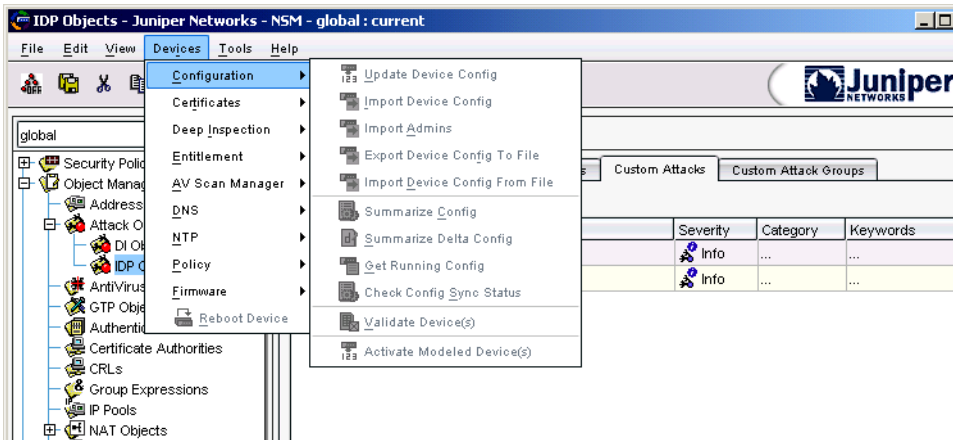
6. Click the Authorization tab. Select the authorization method and the local password for the administrator.

7. Click the Permissions tab, then click the Add icon to select the role **IDP_Only** for this administrator.

8. Click **OK** to close the New Select Role and Domains dialog box. Click **OK** to close the New Admin dialog box. Click **OK** to close the Manage Administrators and Domains dialog box.

The administrator for the standalone IDP device can now log into NetScreen-Security Manager as **IDP_Administrator**. Upon login, the NetScreen-Security Manager UI displays a limited navigation tree and menu options for this user, as shown in Figure 84. Note that the UI displays only the security policy and Object Manager options in the navigation tree; the Devices > Configuration options are not available for this user.

Figure 84: UI Display for IDP_Administrator



Managing IDP

This section describes IDP management on the IDP-capable device.

About Attack Database Updates

Juniper Networks periodically provides attack database updates, in the form of a download file, on the Juniper website. Attack database updates can include the following:

- New or modified predefined IDP attack objects and groups
- New or modified signatures used by the Deep Inspection (DI) feature
- Updates to the IDP engine, which runs in the security device

In a new attack database update, the version number increments by 1. When you download a version of an attack database update from the Juniper Networks website, NetScreen-Security Manager stores the version number of the attack database update. You can check to see if there is a more recent update available than the last one you downloaded.

Downloading Attack Database Updates

The attack database updates are downloaded to the NetScreen-Security Manager GUI server. Perform the following steps to download an attack database update file:

1. From the menu bar, select **Tools > View/Update NSM Attack Database**. The Attack Update Manager wizard appears.
2. Follow the instructions in the Attack Update Manager to download the attack database update file to the NetScreen-Security Manager GUI server.

NOTE: The Juniper Networks website is set by default in the New Preferences dialog box, which you access by selecting **Tools > Preferences**. The GUI Server must have Internet access.

Using Updated Attack Objects

You cannot create, edit, or delete predefined IDP attack objects or groups, but you can update the attack object database installed in the NetScreen-Security Manager GUI server. Updates to predefined IDP attack objects and groups can include the following:

- New descriptions or severities for existing attack objects
- New attack objects or groups
- Deletion of obsolete attack objects

When you download updated IDP attack objects and groups to the GUI server, any new attack objects in the update are available for selection in an IDP rulebase in a security policy. When you install a security policy on your managed device, only the attack objects that are used in IDP rules for the device are pushed from the GUI server to the device.

NOTE: For the DI feature, all updated signatures are pushed to your managed device. For more information about updating the attack object database for DI on your managed device, refer to the *NetScreen-Security Manager Administrator's Guide*.

Updating the IDP Engine

The IDP engine is dynamically changeable firmware that runs on the firewall/VPN device. There are two ways that the IDP engine can be updated on the device:

- When you upgrade the firmware on an IDP/firewall/VPN device, the upgraded firmware will typically include a recent version of the IDP engine as well as a new version of ScreenOS. (For information about upgrading the firmware on a security device, refer to the *NetScreen-Security Manager Administrator's Guide*.)
- You can update the IDP engine on a managed device from an attack database update on the GUI server. Because attack database updates are available more often than firmware releases, an attack database update may include a more recent version of the IDP engine than is available on the latest firmware release. For example, an attack database update might contain updated IDP attack objects that can only be used with an updated version of the IDP engine.

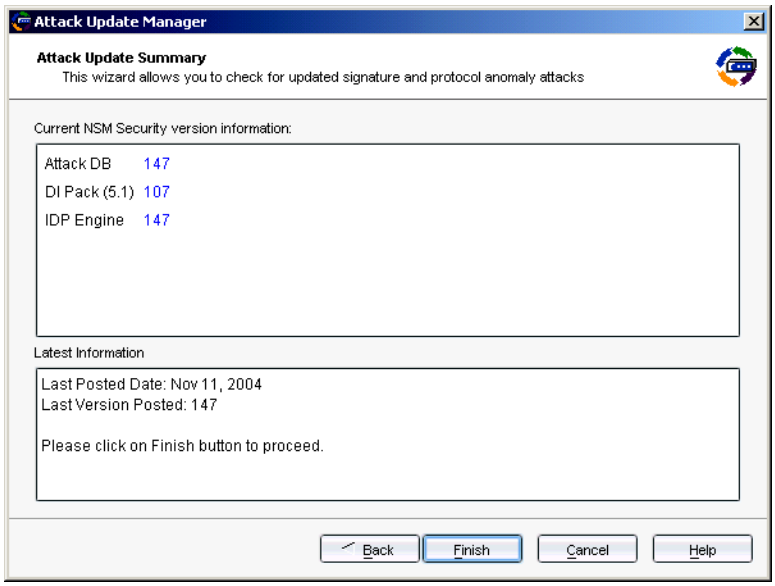
Perform the following steps to see the version of the IDP engine that is currently running on the device:

1. Select **Tools > View/Update NSM Attack Database**. The Attack Update Manager wizard appears.

2. Click **Nex**t.

   The Attack Update Summary, as shown in Figure 85, displays information about the current version downloaded on the GUI server and the latest version available from Juniper Networks.

Figure 85: Attack Update Summary



3. Click **Finish** to continue downloading the latest attack database update, or click **Cancel** to exit the Attack Update Manager.

To update the IDP engine on the device:

1. Select **Devices > IDP Detector Engine > IDP Detector Engine**. The Change Device Sigpack dialog box appears.

NOTE: The IDP engine version you install on the security device must be compatible with the version of the firmware that is running in the device. You cannot downgrade the IDP engine version on the device.

2. Click **Next**, then select the managed devices on which you want to install the IDP engine update.

3. Follow the instructions in the Change Device Manager to update the IDP engine on the selected device.

NOTE: Updating the IDP engine on a device does not require a reboot of the device.

Viewing IDP Logs

When attack objects are matched in an IDP rule, IDP log entries appear in the NetScreen-Security Manager Log Viewer. Perform the following steps to receive IDP log entries in the Log Viewer:

1. Enable the device to send log entries with the desired severity settings to NetScreen-Security Manager:
   a. In Device Manager, open the device configuration for the device.
   b. In the device navigation tree, select **Report Settings > General > NSM**.
   c. Select the severity settings you want logged to NetScreen-Security Manager.
   d. Click **OK**.
2. Enable IDP detection and logging in the security policy installed on the device. For detailed information about configuring IDP logging in the security policy, see "Configuring Security Policies" on page 173.

IDP alarm log entries appear in the Log Viewer and display the following columns of information:

- Source and Destination Address
- Action
- Protocol
- Category (Anomaly, Custom, or Signature)
- Subcategory
- Severity
- Device

Chapter 7

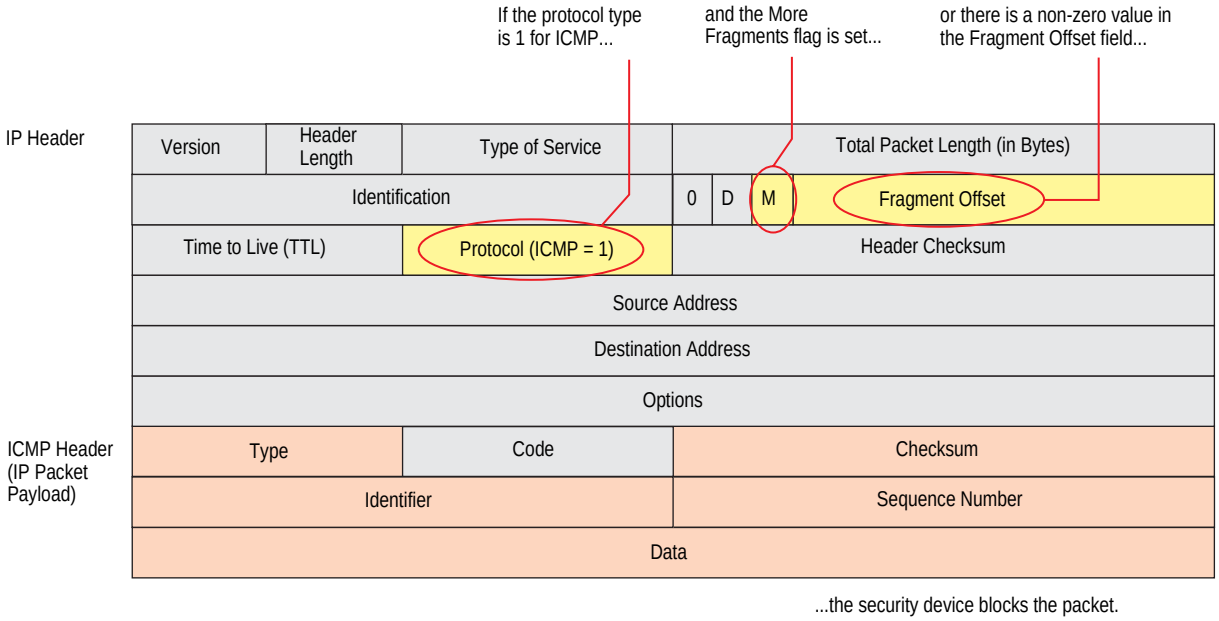
Suspicious Packet Attributes

As shown in the other chapters in this volume, attackers can craft packets to perform reconnaissance or launch denial-of-service (DoS) attacks. Sometimes it is unclear what the intent of a crafted packet is, but the very fact that it is crafted suggests that its being put to some kind of insidious use. All of the SCREEN options presented in this chapter block suspicious packets that might contain hidden threats:

ICMP Fragments

Internet Control Message Protocol (ICMP) provides error reporting and network probe capabilities. Because ICMP packets contain very short messages, there is no legitimate reason for ICMP packets to be fragmented. If an ICMP packet is so large that it must be fragmented, something is amiss. When you enable the ICMP Fragment Protection SCREEN option, the security device blocks any ICMP packet that has the More Fragments flag set or that has an offset value indicated in the offset field.

Figure 86: Blocking ICMP Fragments



To block fragmented ICMP packets, do either of the following, where the specified security zone is the one from which the fragments originate:

WebUI

Screening > Screen (Zone: select a zone name): Select **ICMP Fragment Protection**, then click **Apply**.

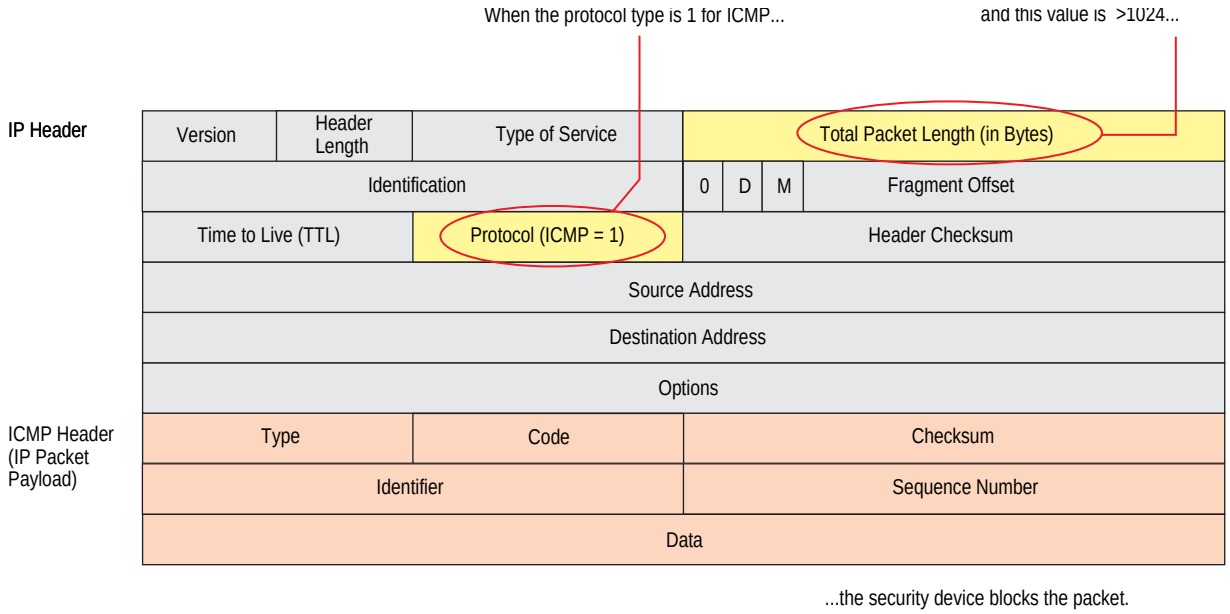
CLI

set zone *zone* screen icmp-fragment

Large ICMP Packets

As stated in “ICMP Fragments” on page 228, Internet Control Message Protocol (ICMP) provides error reporting and network probe capabilities. Because ICMP packets contain very short messages, there is no legitimate reason for large ICMP packets. If an ICMP packet is unusually large, something is wrong. For example, the Loki program uses ICMP as a channel for transmitting covert messages. The presence of large ICMP packets might expose a compromised machine acting as a Loki agent. It also might indicate some other kind of questionable activity.

Figure 87: Blocking Large ICMP Packets



When you enable the Large Size ICMP Packet Protection SCREEN option, the security device checks drops ICMP packets with a length greater than 1024 bytes.

To block large ICMP packets, do either of the following, where the specified security zone is the one from which the ICMP packets originate:

WebUI

Screening > Screen (Zone: select a zone name): Select **Large Size ICMP Packet (Size > 1024) Protection**, then click **Apply**.

CLI

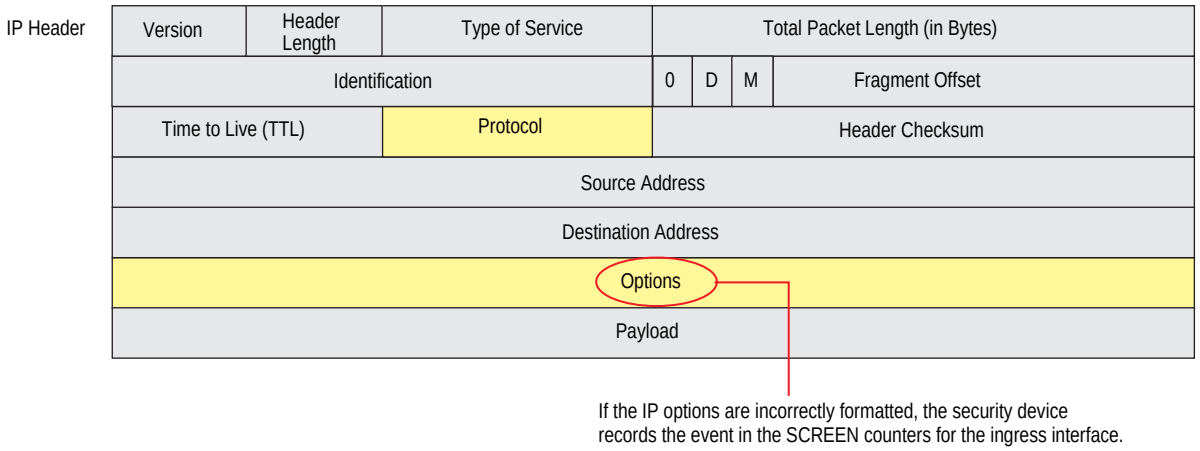
set zone *zone* screen icmp-large

Bad IP Options

The Internet Protocol standard RFC 791, *Internet Protocol*, specifies a set of eight options that provide special routing controls, diagnostic tools, and security. Although the original, intended uses for these options served worthy ends, people have figured out ways to twist these options to accomplish less commendable objectives. (For a summary of the exploits that attackers can initiate from IP options, see "Network Reconnaissance Using IP Options" on page 10.)

Either intentionally or accidentally, attackers sometimes configure IP options incorrectly, producing either incomplete or malformed fields. Regardless of the intentions of the person who crafted the packet, the incorrect formatting is anomalous and potentially harmful to the intended recipient.

Figure 88: Incorrectly Formatted IP Options



When you enable the Bad IP Option Protection SCREEN option, the security device blocks packets when any IP option in the IP packet header is incorrectly formatted. The security device records the event in the event log.

To detect and block IP packets with incorrectly formatted IP options, do either of the following, where the specified security zone is the one from which the packets originate:

WebUI

Screening > Screen (Zone: select a zone name): Select **Bad IP Option Protection**, then click **Apply**.

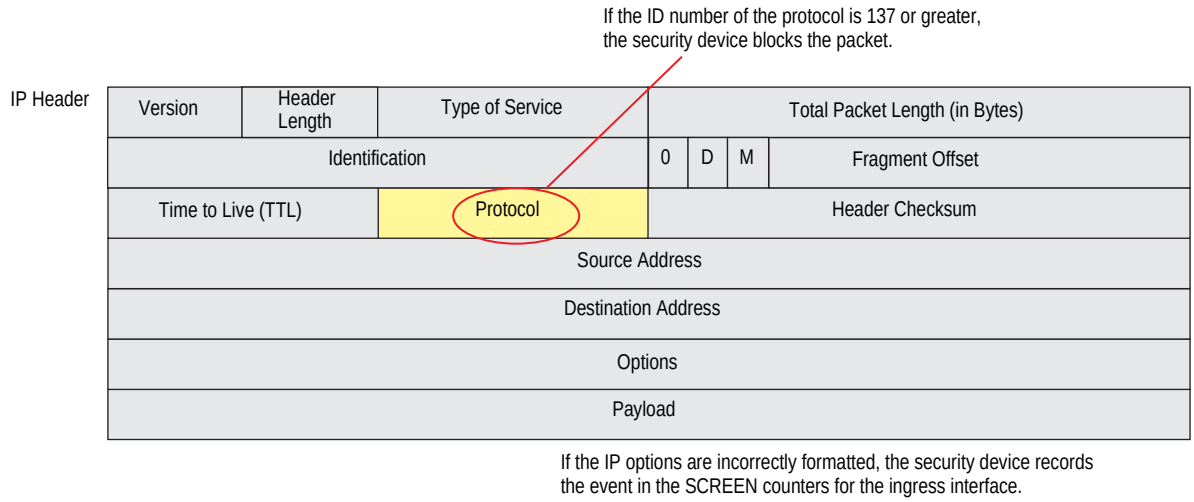
CLI

set zone *zone* screen ip-bad-option

Unknown Protocols

These protocol types with ID numbers of 137 or greater are reserved and undefined at this time. Precisely because these protocols are undefined, there is no way to know in advance if a particular unknown protocol is benign or malicious. Unless your network makes use of a nonstandard protocol with an ID number of 137 or greater, a cautious stance is to block such unknown elements from entering your protected network.

Figure 89: Unknown Protocols



When you enable the Unknown Protocol Protection SCREEN option, the security device drops packets when the protocol field is contains a protocol ID number of 137 or greater.

To drop packets using an unknown protocol, do either of the following, where the specified security zone is the one from which the packets originate:

WebUI

Screening > Screen (Zone: select a zone name): Select **Unknown Protocol Protection**, then click **Apply**.

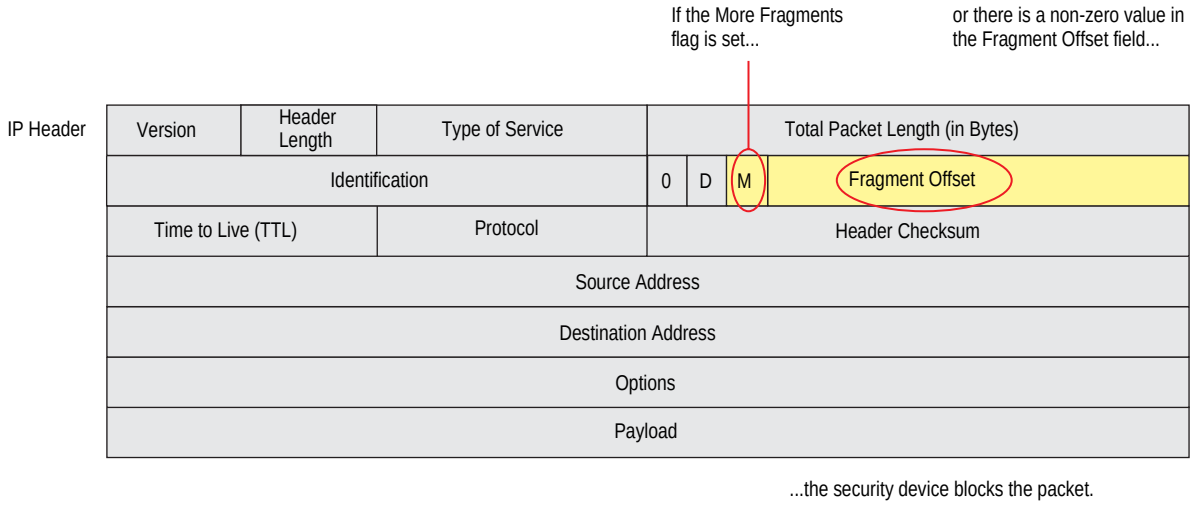
CLI

set zone *zone* screen unknown-protocol

IP Packet Fragments

As packets traverse different networks, it is sometimes necessary to break a packet into smaller pieces (fragments) based upon the maximum transmission unit (MTU) of each network. IP fragments might contain an attacker's attempt to exploit the vulnerabilities in the packet reassembly code of specific IP stack implementations. When the victim receives these packets, the results can range from processing the packets incorrectly to crashing the entire system.

Figure 90: IP Packet Fragments



When you enable the security device to deny IP fragments on a security zone, the device blocks all IP packet fragments that it receives at interfaces bound to that zone.

To drop fragmented IP packets, do either of the following, where the specified security zone is the one from which the fragments originate:

WebUI

Screening > Screen (Zone: select a zone name): Select **Block Fragment Traffic**, then click **Apply**.

CLI

set zone *zone* screen block-frag

SYN Fragments

The Internet Protocol (IP) encapsulates a Transmission Control Protocol (TCP) SYN segment in the IP packet that initiates a TCP connection. Because the purpose of this packet is to initiate a connection and invoke a SYN/ACK segment in response, the SYN segment typically does not contain any data. Because the IP packet is small, there is no legitimate reason for it to be fragmented. A fragmented SYN packet is anomalous, and as such suspect. To be cautious, block such unknown elements from entering your protected network.

When you enable the SYN Fragment Detection SCREEN option, the security device detects packets when the IP header indicates that the packet has been fragmented and the SYN flag is set in the TCP header. The security device records the event in the SCREEN counters list for the ingress interface.

To drop IP packets containing SYN fragments, do either of the following, where the specified security zone is the one from which the packets originate:

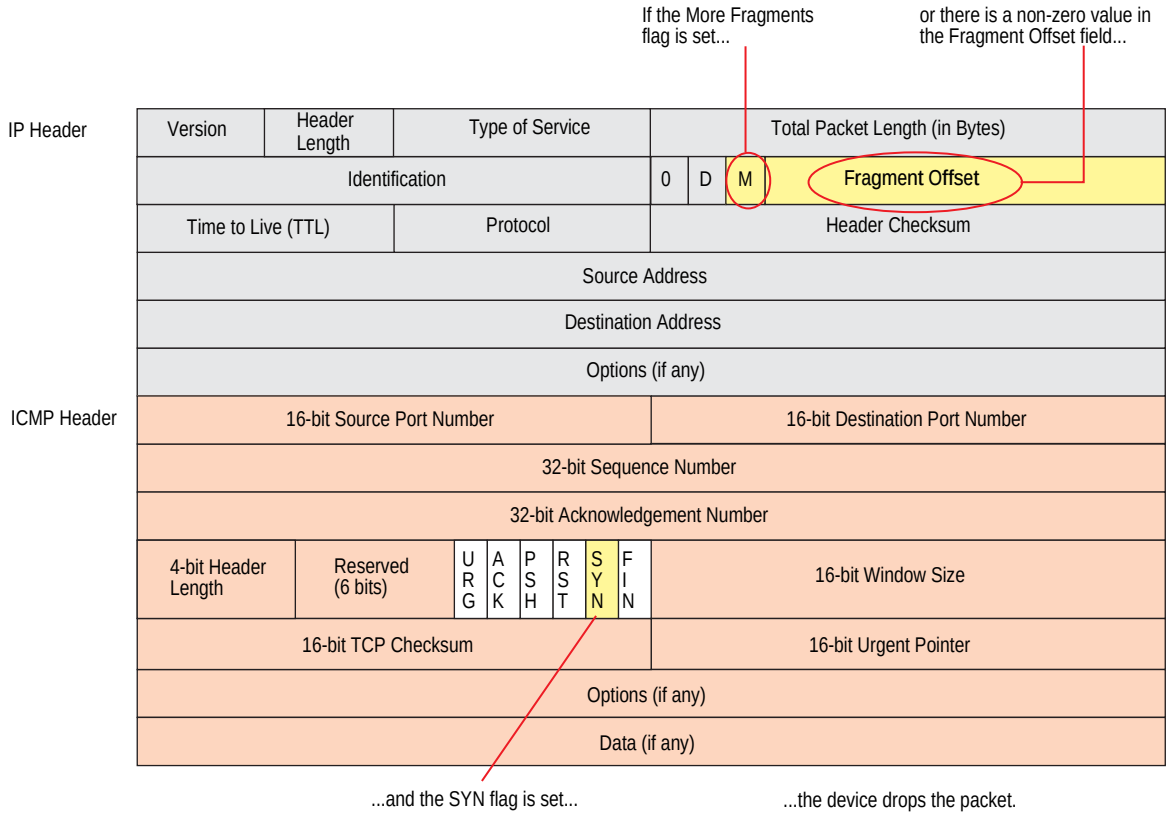
WebUI

Screening > Screen (Zone: select a zone name): Select **SYN Fragment Protection**, then click **Apply**.

CLI

set zone *zone* screen syn-frag

Figure 91: SYN Fragments



Appendix A

Contexts for User-Defined Signatures

The context defines the location in the packet where the Deep Inspection (DI) module searches for a signature matching the attack object pattern. When defining a stateful signature attack object, you can specify any of the contexts in the following lists. After you define an attack object, you must then put it in a user-defined attack object group for use in policies.

NOTE: A user-defined attack object group can only contain user-defined attack objects. You cannot mix predefined and user-defined attack objects in the same attack object group.

When the DI module examines traffic for TCP stream signatures, it does so without regard for contexts. TCP stream signatures look for patterns anywhere in any kind of TCP traffic regardless of the application protocol in use. Stream signatures are defined on NetScreen-5000 and 2000 series systems only. Stream256, however, looks for patterns in the first 256 bytes of data.

Table 21: Contexts for User-Defined Signatures

Protocol	Context	Description (Sets the Context As...)
AIM	aim-chat-room-desc	The description of a chat room in an America Online Instant Messenger (AIM) or ICQ (I Seek You) session.
	aim-chat-room-name	The name of a chat room in an AIM or ICQ session.
	aim-get-file	The name of a file that a user is transferring from a peer.
	aim-nick-name	The nickname of an AIM or ICQ user.
	aim-put-file	The name of a file that a user is transferring to a peer.
	aim-screen-name	The screen name of an AIM or ICQ user.
DNS	dns-cname	The CNAME (canonical name) in a Domain Name System (DNS) request or response, as defined in RFC 1035, *Domain Names—Implementation and Specification*).
FTP	ftp-command	One of the FTP commands specified in RFC 959, *File Transfer Protocol (FTP).*
	ftp-password	An FTP login password.
	ftp-pathname	A directory or file name in any FTP command.
	ftp-username	The name that a user enters when logging in to an FTP server.
Gnutella	gnutella-http-get-filename	The name of a file that a Gnutella client intends to retrieve.
HTTP	http-authorization	The username and password decoded from an Authorization: Basic header in an HyperText Transfer Protocol (HTTP) request, as specified in RFC 1945, *HyperText Transfer Protocol—HTTP/1.0*.

Protocol	Context	Description (Sets the Context As...)
	http-header-user-agent	The user-agent field in the header of an HTTP request. (When users visit a website, they provide information about their browsers in this field.)
	http-request	An HTTP request line.
	http-status	The status line in an HTTP reply. (The status line is a three-digit code that a webserver sends a client to communicate the state of a connection. For example, 401 means "Unauthorized" and 404 means "Not found".)
	http-text-html	The text, or HyperText Markup Language (HTML) data, in an HTTP transaction.
	http-url	The uniform resource locator (URL) in an HTTP request as it appears in the data stream.
	http-url-parsed	A "normalized" text string decoded from a unicode string that comprises a URL used in HTTP.
	http-url-variable-parsed	A decoded common gateway interface (CGI) variable in the URL of an HTTP-GET request.
IMAP	imap-authenticate	an argument in an Internet Mail Access Protocol (IMAP) AUTHENTICATE command. The argument indicates the type of authentication mechanism that the IMAP client proposes to the server. Examples are KERBEROS_V4, GSSAPI (see RFC 1508, *Generic Security Service Application Program Interface*), and SKEY. For information about IMAP, see RFC 1730, *Internet Message Access Protocol - Version 4*, and RFC 1731, *IMAP4 Authentication Mechanisms*.
	imap-login	Either the username or plaintext password in an IMAP LOGIN command.
	imap-mailbox	The mailbox text string in an IMAP SELECT command.
	imap-user	The username in an IMAP LOGIN command.
MSN Messenger	msn-display-name	The display name of a user in a Microsoft Network (MSN) Instant Messaging session.
	msn-get-file	The name of a file that a client is downloading from a peer.
	msn-put-file	The name of a file that a client is sending to a peer.
	msn-sign-in-name	The screen name (login name) of an MSN Instant Messaging user.
POP3	pop3-auth	The AUTH command in a Post Office Protocol, version 3 (POP3) session. For information about POP3, see RFC 1939, *Post Office Protocol—Version 3*.
	pop3-header-from	The text string in the "From:" header of an email in a POP3 transaction.
	pop3-header-line	The text string in any header line of an email in a POP3 transaction.
	pop3-header-subject	The text string in the "Subject:" header of an email in a POP3 transaction.
	pop3-header-to	The text string in the "To:" header of an email in a POP3 transaction.
	pop3-mime-content-filename	The content file name of a Multipurpose Internet Mail Extensions (MIME) attachment in a POP3 session.
	pop3-user	The username in a POP3 session.
SMB	smb-account-name	The name of a Server Message Blocks (SMB) account in a SESSION_SETUP_ANDX request in an SMB session.
	smb-connect-path	The connect path in the TREE_CONNECT_ANDX request in an SMB session.
	smb-connect-service	The name of the connect service in the TREE_CONNECT_ANDX request in an SMB session.
	smb-copy-filename	The name of a file in a COPY request in an SMB session.
	smb-delete-filename	The name of a file in a DELETE request in an SMB session.
	smb-open-filename	The name of a file in the NT_CREATE_ANDX and OPEN_ANDX requests in an SMB session.

Protocol	Context	Description (Sets the Context As...)
SMTP	smtp-from	The text string in a "MAIL FROM" command line in a Simple Mail Transfer Protocol (SMTP) session, as described in RFC 2821, *Simple Mail Transfer Protocol*.
	smtp-header-from	The text string in the "From:" header in an SMTP session.
	smtp-header-line	The text string in any header line in an SMTP session.
	smtp-header-subject	The text string in the "Subject:" header in an SMTP session.
	smtp-header-to	The text string in the "To:" header in an SMTP session.
	smtp-mime-content-filename	The content file name of a Multipurpose Internet Mail Extensions (MIME) attachment in an SMTP session.
	smtp-rcpt	The text string in a "RCPT TO" command line in an SMTP session.
–	stream256	The first 256 bytes of a reassembled, normalized TCP data stream.
Yahoo! Messenger	ymsg-alias	The alternate identifying name associated with the main username of a Yahoo! Instant Messaging user.
	ymsg-chatroom-message	The text in messages exchanged in a Yahoo! Instant Messaging chatroom.
	ymsg-chatroom-name	The name of a Yahoo! Instant Messaging chatroom.
	ymsg-nickname	The nickname of a Yahoo! Instant Messaging user.
	ymsg-p2p-get-filename-url	The location of a file on a Yahoo! Instant Messaging peer's machine from which it can be downloaded.
	ymsg-p2p-put-filename-url	The location of a file on a Yahoo! Instant Messaging peer's machine to which it can be downloaded.

Index

A

I

J

L

M

N

O

P

R

S

T

U

V

W

Y

Z