

ASA/PIX 7.x and Later: Mitigating the Network Attacks

Document ID: 100830

Contents

[Introduction](#)

[Prerequisites](#)

- [Requirements](#)
- [Components Used](#)
- [Related Products](#)
- [Conventions](#)

[Protecting Against SYN Attacks](#)

- [TCP SYN Attack](#)
- [Mitigation](#)

[Protecting Against IP Spoofing Attacks](#)

- [IP Spoofing](#)
- [Mitigation](#)

[Spoofing Identification Using Syslog Messages](#)

[Basic Threat Detection Feature in ASA 8.x](#)

- [Syslog Message 733100](#)

[Cisco Support Community - Featured Conversations](#)

[Related Information](#)

Introduction

This document describes how to mitigate the various network attacks, such as Denial-of-Services (DoS), using Cisco Security Appliance (ASA/PIX).

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

The information in this document is based on the Cisco 5500 Series Adaptive Security Appliance (ASA) that runs software version 7.0 and later.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Related Products

This document can also be used with Cisco 500 Series PIX that runs software version 7.0 and later.

Conventions

Refer to [Cisco Technical Tips Conventions](#) for more information on document conventions.

Protecting Against SYN Attacks

How do you mitigate the Transmission Control Protocol (TCP) synchronize/start (SYN) attacks on the ASA/PIX?

TCP SYN Attack

TCP SYN attack is a type of DoS attack in which a sender transmits a volume of connections that cannot be completed. This causes the connection queues to fill up, thereby denying service to legitimate TCP users.

When a normal TCP connection starts, a destination host receives a SYN packet from a source host and sends back a synchronize acknowledge (SYN ACK). The destination host must then hear an ACK of the SYN ACK before the connection is established. This is referred to as the TCP three-way handshake.

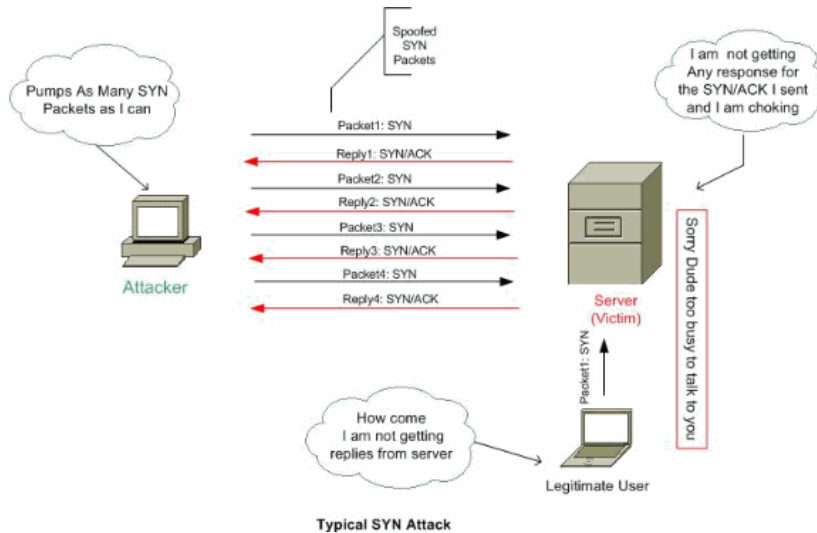
While waiting for the ACK to the SYN ACK, a connection queue of finite size on the destination host keeps track of connections waiting to be completed. This queue typically empties quickly because the ACK is expected to arrive a few milliseconds after the SYN ACK.

The TCP SYN attack exploits this design by having an attacking source host generate TCP SYN packets with random source addresses toward a victim host. The victim destination host sends a SYN ACK back to the random source address and adds an entry to the connection queue. Because the SYN ACK is destined for an incorrect or non-existent host, the last part of the "three-way handshake" is never completed and the entry remains in the connection queue until a timer expires, typically for about one minute. By generating phony TCP SYN packets from random IP addresses at a rapid rate, it is possible to fill up the connection queue and deny TCP services (such as e-mail, file transfer, or WWW) to legitimate users.

There is no easy way to trace the originator of the attack because the IP address of the source is forged.

The external manifestations of the problem include inability to get e-mail, inability to accept connections to WWW or FTP services, or a large number of TCP connections on your host in the state SYN_RCVD.

Refer to [Defenses Against TCP SYN Flooding Attacks](#) for more information on TCP SYN attacks.



Mitigation

This section describes how to mitigate the SYN attacks by setting the maximum TCP and User Datagram Protocol (UDP) connections, maximum embryonic connections, connection timeouts, and how to disable TCP sequence randomization.

If the embryonic connection limit is reached, then the security appliance responds to every SYN packet sent to the server with a SYN+ACK, and does not pass the SYN packet to the internal server. If the external device responds with an ACK packet, then the security appliance knows it is a valid request (and not part of a potential SYN attack). The security appliance then establishes a connection with the server and joins the connections together. If the security appliance does not get an ACK back from the server, it aggressively times out that embryonic connection.

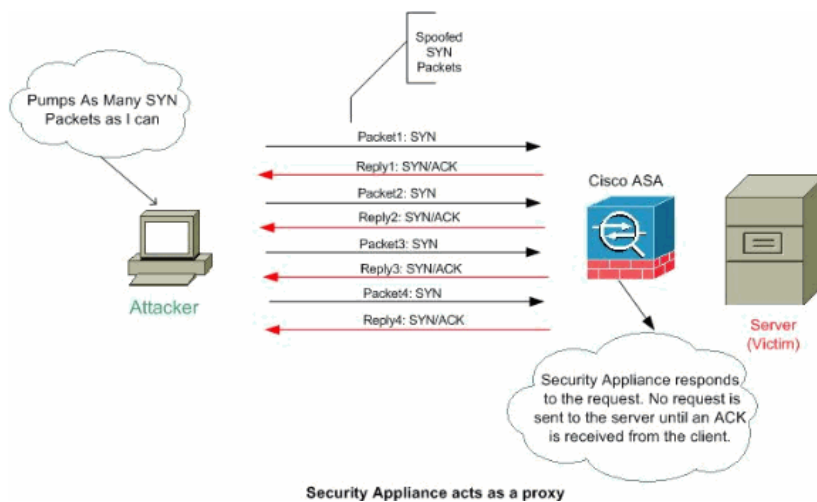
Each TCP connection has two Initial Sequence Number (ISNs): one generated by the client and one generated by the server. The security appliance randomizes the ISN of the TCP SYN passing in both the inbound and outbound directions.

Randomizing the ISN of the protected host prevents an attacker from predicting the next ISN for a new connection and potentially hijacking the new session.

TCP initial sequence number randomization can be disabled if required. For example:

- If another in-line firewall is also randomizing the initial sequence numbers, there is no need for both firewalls to be performing this action, even though this action does not affect the traffic.
- If you use external BGP (eBGP) multi-hop through the security appliance, and the eBGP peers are using MD5, randomization breaks the MD5 checksum.
- You use a Wide Area Application Services (WAAS) device that requires the security appliance not to randomize the sequence numbers of connections.

Note: You can also configure maximum connections, maximum embryonic connections, and TCP sequence randomization in the NAT configuration. If you configure these settings for the same traffic using both methods, then the security appliance uses the lower limit. For TCP sequence randomization, if it is disabled using either method, then the security appliance disables TCP sequence randomization.



Complete these steps in order to set connection limits:

1. In order to identify the traffic, add a class map using the **class-map** command according to [Using Modular Policy Framework](#).
2. In order to add or edit a **policy map** that sets the actions to take with the class map traffic, enter this command:

```
hostname(config)#policy-map name
```

3. In order to identify the class map (from step 1) to which you want to assign an action, enter this command:

```
hostname(config-pmap)#class class_map_name
```

4. In order to set the maximum connections (both TCP and UDP), maximum embryonic connections, per-client-embryonic-max, per-client-max or whether to disable TCP sequence randomization, enter this command:

```
hostname(config-pmap-c)#set connection {[conn-max number]
[embryonic-conn-max number] [per-client-embryonic-max number]
[per-client-max number][random-sequence-number {enable |
disable}]}
```

Where number is an integer between 0 and 65535. The default is 0, which means no limit on connections.

You can enter this command all on one line (in any order), or you can enter each attribute as a separate command. The command is combined on one line in the running configuration.

5. In order to set the timeout for connections, embryonic connections (half-opened) and half-closed connections, enter this command:

```
hostname(config-pmap-c)#set connection {[embryonic hh[:mm[:ss]]]
[half-closed hh[:mm[:ss]]] [tcp hh[:mm[:ss]]]}
```

Where **embryonic** hh:mm:ss is a time between 0:0:5 and 1192:59:59. The default is 0:0:30. You can also set this value to 0, which means the connection never times out.

The **half-closed** hh:mm:ss and **tcp** hh:mm:ss values are a time between 0:5:0 and 1192:59:59. The default for **half-closed** is 0:10:0 and the default for **tcp** is 1:0:0. You can also set these values to 0, which means the connection never times out.

You can enter this command all on one line (in any order), or you can enter each attribute as a separate command. The command is combined on one line in the running configuration.

- o **Embryonic (Half-opened) connection**—An embryonic connection is a TCP connection request that has not finished the necessary handshake between source and destination.
 - o **Half-closed connection**—Half closed connection is when the connection is only closed in one direction by sending FIN. However, TCP session is still maintained by peer.
 - o **Per-client-embryonic-max**—The maximum number of simultaneous embryonic connections allowed per client, between 0 and 65535. The default is 0, which allows unlimited connections.
 - o **Per-client-max**—The maximum number of simultaneous connections allowed per client, between 0 and 65535. The default is 0, which allows unlimited connections.
6. In order to activate the policy map on one or more interfaces, enter this command:

```
hostname(config)#service-policy policymap_name {global | interface
interface_name}
```

Where **global** applies the policy map to all interfaces, and **interface** applies the policy to one interface. Only one global policy is allowed. You can override the global policy on an interface by applying a service policy to that interface. You can only apply one policy map to each interface.

Example:

```
ciscoasa(config)#class-map tcp_syn
ciscoasa(config-cmap)#match port tcp eq 80
ciscoasa(config-cmap)#exit
ciscoasa(config)#policy-map tcpmap
ciscoasa(config-pmap)#class tcp_syn
ciscoasa(config-pmap-c)#set connection conn-max 100
ciscoasa(config-pmap-c)#set connection embryonic-conn-max 200
ciscoasa(config-pmap-c)#set connection per-client-embryonic-max 10
ciscoasa(config-pmap-c)#set connection per-client-max 5
ciscoasa(config-pmap-c)#set connection random-sequence-number enable
ciscoasa(config-pmap-c)#set connection timeout embryonic 0:0:45
ciscoasa(config-pmap-c)#set connection timeout half-closed 0:25:0
ciscoasa(config-pmap-c)#set connection timeout tcp 2:0:0
ciscoasa(config-pmap-c)#exit
ciscoasa(config-pmap)#exit
ciscoasa(config)#service-policy tcpmap global
```

Note: In order to verify the total number of half-open sessions for any particular host, use this command:

```
ASA-5510-8x# show local-host all

Interface dmz: 0 active, 0 maximum active, 0 denied
Interface management: 0 active, 0 maximum active, 0 denied
Interface xx: 0 active, 0 maximum active, 0 denied
Interface inside: 7 active, 18 maximum active, 0 denied

local host: <10.78.167.69>,

TCP flow count/limit = 2/unlimited

TCP embryonic count to host = 0

TCP intercept watermark = unlimited

UDP flow count/limit = 0/unlimited
```

Note: The line, TCP embryonic count to host, displays the number of half-open sessions.

Protecting Against IP Spoofing Attacks

Can the PIX/ASA block IP spoof attacks?

IP Spoofing

In order to gain access, intruders create packets with spoofed source IP addresses. This exploits applications that use authentication based on IP addresses and leads to unauthorized user and possibly root access on the targeted system. Examples are the rsh and rlogin services.


It is possible to route packets through filtering-router firewalls if they are not configured to filter incoming packets whose source address is in the local domain. It is important to note that the described attack is possible even if no reply packets can reach the attacker.

Examples of configurations that are potentially vulnerable include:

- Proxy firewalls where the proxy applications use the source IP address for authentication
- Routers to external networks that support multiple internal interfaces
- Routers with two interfaces that support subnetting on the internal network

Mitigation

Unicast Reverse Path Forwarding (uRPF) guards against IP spoofing (a packet uses an incorrect source IP address to obscure its true source) by ensuring that all packets have a source IP address that matches the correct source interface according to the routing table.

Normally, the security appliance only looks at the destination address when determining where to forward the packet. Unicast RPF instructs the security appliance to also look at the source address. This is why it is called **Reverse Path Forwarding**. For any traffic that you want to allow through the security appliance, the security appliance routing table must include a route back to the source address. See [RFC 2267](#)  for more information.

Note: The `-- %PIX-1-106021: Deny protocol reverse path check from src_addr to dest_addr on interface int_name` log message can be seen when the reverse path check is enabled. Disable the reverse path check with the `no ip verify reverse-path interface (interface name)` command in order to resolve this issue:

[no ip verify reverse-path interface \(interface name\)](#)

For outside traffic, for example, the security appliance can use the default route to satisfy the Unicast RPF protection. If traffic enters from an outside interface, and the source address is not known to the routing table, the security appliance uses the default route to correctly identify the outside interface as the source interface.

If traffic enters the outside interface from an address that is known to the routing table, but is associated with the inside interface, then the security appliance drops the packet. Similarly, if traffic enters the inside interface from an unknown source address, the security appliance drops the packet because the matching route (the default route) indicates the outside interface.

Unicast RPF is implemented as shown:

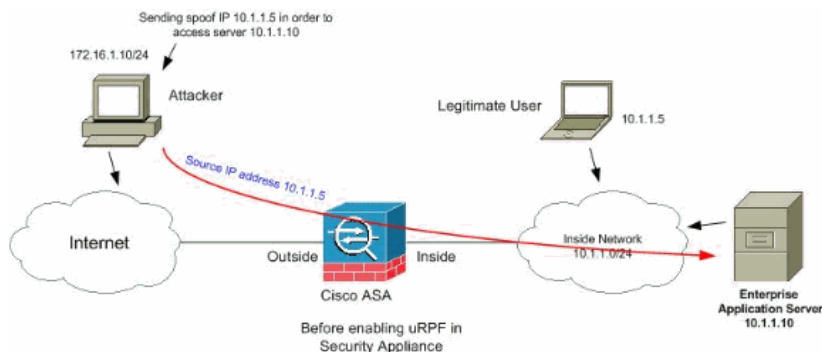
- ICMP packets have no session, so each packet is checked.
- UDP and TCP have sessions, so the initial packet requires a reverse route lookup. Subsequent packets arriving during the session are checked using an existing state maintained as part of the session. Non-initial packets are checked to ensure they arrived on the same interface used by the initial packet.

In order to enable Unicast RPF, enter this command:

```
hostname(config)#ip verify reverse-path interface interface_name
```

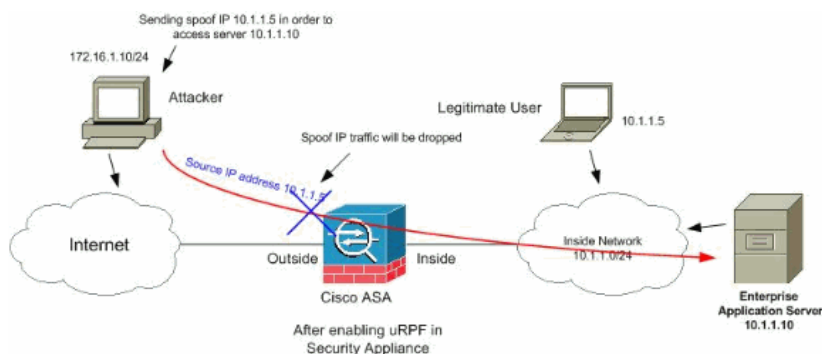
Example:

As shown in this figure, the Attacker PC originates a request to the application server 10.1.1.10 by sending a packet with a forged source IP address 10.1.1.5/24, and the server sends a packet to the real IP address 10.1.1.5/24 in response to the request. This type of illegal packet will attack both the application server and legitimate user in the inside network.



Unicast RPF can prevent attacks based on source address spoofing. You need to configure the uRPF in the outside interface of the ASA as shown here:

```
ciscoasa(config)#ip verify reverse-path interface outside
```



Spoofing Identification Using Syslog Messages

The security appliance keeps receiving syslog error messages as shown. This indicates potential attacks using spoofed packets or that might trigger due to asymmetric routing.

1. `%PIX|ASA-2-106001: Inbound TCP connection denied from IP_address/port to IP_address/port flags tcp_flags on interface interface_name`

Explanation

This is a connection-related message. This message occurs when an attempt to connect to an inside address is denied by the security policy that is defined for the specified traffic type. Possible *tcp_flags* values correspond to the flags in the TCP header that were present when the connection was denied. For example, a TCP packet arrived for which no connection state exists in the security appliance, and it was dropped. The *tcp_flags* in this packet are FIN and ACK.

The *tcp_flags* are as follows:

- o ACK—The acknowledgment number was received.
- o FIN—Data was sent.
- o PSH—The receiver passed data to the application.
- o RST—The connection was reset.
- o SYN—Sequence numbers were synchronized to start a connection.
- o URG—The urgent pointer was declared valid.

There are many reasons for static translation to fail on the PIX/ASA. But, a common reason is if the demilitarized zone (DMZ) interface is configured with the same security level (0) as the outside interface.

In order to resolve this issue, assign a different security level to all interfaces

Refer to [Configuring Interface Parameters](#) for more information.

This error message also appears if an external device sends an IDENT packet to the internal client, which is dropped by the PIX Firewall. Refer to [PIX Performance Issues Caused by IDENT Protocol](#) for more information

2. `%PIX|ASA-2-106007: Deny inbound UDP from outside_address/outside_port to inside_address/inside_port due to DNS {Response|Query}`

Explanation

This is a connection-related message. This message is displayed if the specified connection fails because of an **outbound deny** command. The protocol variable can be ICMP, TCP, or UDP.

Recommended Action: Use the **show outbound** command to check outbound lists.

3. `%PIX|ASA-3-106014: Deny inbound icmp src interface_name: IP_address dst interface_name: IP_address (type dec, code dec)`

Explanation

The security appliance denied any inbound ICMP packet access. By default, all ICMP packets are denied access unless specifically permitted.

4. `%PIX|ASA-2-106016: Deny IP spoof from (IP_address) to IP_address on interface interface_name.`

Explanation

This message is generated when a packet arrives at the security appliance interface that has a destination IP address of 0.0.0.0 and a destination MAC address of the security appliance interface. In addition, this message is generated when the security appliance discarded a packet with an invalid source address, which can include one of the following or some other invalid address:

- o Loopback network (127.0.0.0)
- o Broadcast (limited, net-directed, subnet-directed, and all-subnets-directed)
- o The destination host (land.c)

In order to further enhance spoof packet detection, use the **icmp** command to configure the security appliance to discard packets with source addresses belonging to the internal network. This is because the **access-list** command has been deprecated and is no longer guaranteed to work correctly.

Recommended Action: Determine if an external user is trying to compromise the protected network. Check

for misconfigured clients.

5. `%PIX|ASA-2-106017: Deny IP due to Land Attack from IP_address to IP_address`

Explanation

The security appliance received a packet with the IP source address equal to the IP destination, and the destination port equal to the source port. This message indicates a spoofed packet that is designed to attack systems. This attack is referred to as a Land Attack.

Recommended Action: If this message persists, an attack might be in progress. The packet does not provide enough information to determine where the attack originates.

6. `%PIX|ASA-1-106021: Deny protocol reverse path check from source_address to dest_address on interface interface_name`

Explanation

An attack is in progress. Someone is attempting to spoof an IP address on an inbound connection. Unicast RPF, also known as reverse route lookup, detected a packet that does not have a source address represented by a route and assumes that it is part of an attack on your security appliance.

This message appears when you have enabled Unicast RPF with the `ip verify reverse-path` command. This feature works on packets input to an interface. If it is configured on the outside, then the security appliance checks packets arriving from the outside.

The security appliance looks up a route based on the source address. If an entry is not found and a route is not defined, then this system log message appears and the connection is dropped.

If there is a route, the security appliance checks which interface it corresponds. If the packet arrived on another interface, it is either a spoof or there is an asymmetric routing environment that has more than one path to a destination. The security appliance does not support asymmetric routing.

If the security appliance is configured on an internal interface, it checks static `route` command statements or RIP. If the source address is not found, then an internal user is spoofing their address.

Recommended Action: Even though an attack is in progress, if this feature is enabled, no user action is required. The security appliance repels the attack.

Note: The `show asp drop` command shows the packets or connections dropped by the accelerated security path (asp), which might help you troubleshoot a problem. It also indicates when the last time the asp drop counters were cleared. Use the `show asp drop rpf-violated` command in which the counter is incremented when `ip verify reverse-path` is configured on an interface and the security appliance receives a packet for which the route lookup of the source IP did not yield the same interface as the one on which the packet was received.

```
ciscoasa#show asp drop frame rpf-violated
Reverse-path verify failed 2
```

Note: Recommendation: Trace the source of traffic based on the source IP printed in this next system message, and investigate why it is sending spoofed traffic.

Note: System log messages: 106021

7. `%PIX|ASA-1-106022: Deny protocol connection spoof from source_address to dest_address on interface interface_name`

Explanation

A packet matching a connection arrives on a different interface from the interface where the connection began.

For example, if a user starts a connection on the inside interface, but the security appliance detects the same connection arriving on a perimeter interface, the security appliance has more than one path to a destination. This is known as asymmetric routing and is not supported on the security appliance.

An attacker also might attempt to append packets from one connection to another as a way to break into the security appliance. In either case, the security appliance displays this message and drops the connection.

Recommendation Action: This message appears when the `ip verify reverse-path` command is not configured. Check that the routing is not asymmetric.

8. `%PIX|ASA-4-106023: Deny protocol src [interface_name:source_address/source_port] dst interface_name:dest_address/dest_port [type {string}, code {code}] by access_group acl_ID`

Explanation

An IP packet was denied by the ACL. This message displays even if you do not have the **log** option enabled for an ACL.

Recommendation Action: If messages persist from the same source address, messages might indicate a foot-printing or port-scanning attempt. Contact the remote host administrators.

9. %PIX|ASA-3-210011: Connection limit exceeded cnt/limit for dir packet from sip/sport to dip/dport on interface if_name.
10. %ASA-4-419002: Received duplicate TCP SYN from in_interface:src_address/src_port to out_interface:dest_address/dest_port with different initial sequence number.

Explanation

This system log message indicates that establishing a new connection through the firewall device will result in exceeding at least one of the configured maximum connection limits. The system log message applies both for connection limits configured using a static command, or to those configured using Cisco Modular Policy Framework. The new connection will not be allowed through the firewall device until one of the existing connections are torn down, thereby bringing the current connection count below the configured maximum.

- o *cnt*—Current connection count
- o *limit*—Configured connection limit
- o *dir*—Direction of traffic, inbound or outbound
- o *sip*—Source IP address
- o *sport*—Source Port
- o *dip*—Destination IP address
- o *dport*—Destination Port
- o *if_name*—Name of the interface on which the traffic unit is received, either Primary or Secondary.

Recommendation Action: Because connection limits are configured for a good reason, this system log message could indicate a possible DoS attack, in which case the source of the traffic could likely be a spoofed IP address. If the source IP address is not totally random, identifying the source and blocking it using an access-list might help. In other cases, getting sniffer traces and analyzing the source of the traffic would help in isolating unwanted traffic from legitimate traffic.

Basic Threat Detection Feature in ASA 8.x

Cisco Security Appliance ASA/PIX supports the feature called threat detection from software version 8.0 and later. Using basic threat detection, the security appliance monitors the rate of dropped packets and security events due to these reasons:

- Denial by access lists
- Bad packet format (such as invalid-ip-header or invalid-tcp-hdr-length)
- Connection limits exceeded (both system-wide resource limits, and limits set in the configuration)
- DoS attack detected (such as an invalid SPI, Stateful Firewall check failure)
- Basic firewall checks failed (This option is a combined rate that includes all firewall-related packet drops in this bulleted list. It does not include non-firewall-related drops such as interface overload, packets failed at application inspection, and scanning attack detected.)
- Suspicious ICMP packets detected
- Packets failed application inspection
- Interface overload
- Scanning attack detected (This option monitors scanning attacks; for example, the first TCP packet is not a SYN packet, or the TCP connection failed the 3-way handshake. Full scanning threat detection (refer to [Configuring Scanning Threat Detection](#) for more information) takes this scanning attack rate information and acts on it by classifying hosts as attackers and automatically shunning them, for example.)
- Incomplete session detection such as TCP SYN attack detected or no data UDP session attack detected.

When the security appliance detects a threat, it immediately sends a system log message ([730100](#)).

Basic threat detection affects performance only when there are drops or potential threats. Even in this scenario, the performance impact is insignificant.

The **show threat-detection rate** command is used in order to identify potential attacks when you are logged into the security appliance.

```
ciscoasa#show threat-detection rate
                Average(eps)   Current(eps) Trigger   Total events
10-min ACL drop:                0             0         0           16
1-hour ACL drop:                 0             0         0          112
1-hour SYN attck:                5             0         2         21438
10-min Scanning:                 0             0        29           193
1-hour Scanning:               106            0        10        384776
1-hour Bad pkts:                76             0         2        274690
10-min Firewall:                 0             0         3           22
1-hour Firewall:                76             0         2        274844
10-min DoS attck:                0             0         0            6
1-hour DoS attck:                0             0         0            42
10-min Interface:                0             0         0           204
1-hour Interface:               88             0         0        318225
```

Refer to [Configuring Basic Threat Detection](#) section of ASA 8.0 configuration guide for more information on the configuration part.

Syslog Message 733100

Error Message:

```
%ASA-4-733100: Object drop rate rate_ID exceeded. Current burst rate
is rate_val per second, max configured rate is rate_val;
Current average rate is rate_val per second, max configured rate is
rate_val; Cumulative total count is total_cnt
```

The specified object in the system log message has exceeded the specified burst threshold rate or average threshold rate. The object can be drop activity of a host, TCP/UDP port, IP protocol, or various drops due to potential attacks. It indicates the system is under potential attack.

Note: These error messages with resolution are applicable only to ASA 8.0 and later.

1. Object—The general or particular source of a drop rate count, which might include these:
 - o Firewall
 - o Bad pkts
 - o Rate limit
 - o DoS attck
 - o ACL drop
 - o Conn limit
 - o ICMP attk
 - o Scanning
 - o SYN attck
 - o Inspect
 - o Interface
2. *rate_ID*—The configured rate that is being exceeded. Most objects can be configured with up to three different rates for different intervals.
3. *rate_val*—A particular rate value.
4. *total_cnt*—The total count since the object was created or cleared.

These three examples show how these variables occur:

- For an interface drop due to a CPU or bus limitation:

```
%ASA-4-733100: [Interface] drop rate 1 exceeded. Current burst rate is 1 per
second,
max configured rate is 8000; Current average rate is 2030 per second,
max configured rate is 2000; Cumulative total count is 3930654
```

- For a scanning drop due to potential attacks:

```
ASA-4-733100: [Scanning] drop rate-1 exceeded. Current burst rate is 10 per
second_
max configured rate is 10; Current average rate is 245 per second_
max configured rate is 5; Cumulative total count is 147409 (35 instances received)
```

- For bad packets due to potential attacks:

```
%ASA-4-733100: [Bad pkts] drop rate 1 exceeded. Current burst rate is 0 per
second,
max configured rate is 400; Current average rate is 760 per second,
max configured rate is 100; Cumulative total count is 1938933
```

Recommended Action:

Perform these steps according to the specified object type that appears in the message:

1. If the object in the syslog message is one of these:

- Firewall
- Bad pkts
- Rate limit
- DoS attack
- ACL drop
- Conn limit
- ICMP attk
- Scanning
- SYN attck
- Inspect
- Interface

Check whether the drop rate is acceptable for the running environment.

2. Adjust the threshold rate of the particular drop to an appropriate value by running the **threat-detection rate xxx** command, where xxx is one of these:

- acl-drop
- bad-packet-drop
- conn-limit-drop
- dos-drop
- fw-drop
- icmp-drop
- inspect-drop
- interface-drop
- scanning-threat
- syn-attack

3. If the object in the syslog message is a TCP or UDP port, an IP protocol, or a host drop, check whether the drop rate is acceptable for the running environment.
4. Adjust the threshold rate of the particular drop to an appropriate value by running the **threat-detection rate bad-packet-drop** command. Refer to the [Configuring Basic Threat Detection](#) section of the ASA 8.0 Configuration Guide for more information.

Note: If you do not want the drop rate exceed warning to appear, you can disable it by running the **no threat-detection basic-threat** command.

Cisco Support Community - Featured Conversations

Cisco Support Community is a forum for you to ask and answer questions, share suggestions, and collaborate with your peers. Below are just some of the most recent and relevant conversations happening right now.

Want to see more? Join us by clicking here
help firewall capacity expansion for... castleju1 1 Reply 5 months, 3 weeks ago
ASA NAT exempt, TCP sequence Moxv82KAW 5 Replies 1 year, 3 months ago
How to establish a baseline for... mhcraig 0 Replies 2 years, 9 months ago
ASA/PIX 7.X Failover addressing jeffasher 1 Reply 5 years, 2 months ago
Pings thru ASA/PIX 7.x vikram_anumukonda 4 Replies 3 years, 6 months ago
IP Pools assigned from AAA server joseph.kukis_at_stevens.edu 3 Replies 5 months, 3 weeks ago
ASA and OSPF svanquilder 2 Replies 2 years, 7 months ago
Restrict Network Bandwidth ray_stone 4 Replies 3 years, 6 months ago
vpn authentication ofir_at_oscar 7 Replies 2 years, 8 months ago
VPN Backup Architecture gdelpanta 10 Replies 5 years, 6 months ago
Enable Raduis Attributes for Group Lock marcohernandez 4 Replies 8 months, 2 days ago
Start A New Discussion Subscribe

Related Information

- [Cisco 5500 Series Adaptive Security Appliances Support Page](#)
- [Cisco 500 Series PIX Support Page](#)
- [Defenses Against TCP SYN Flooding Attacks](#)
- [Cisco Applied Mitigation Bulletin: Identifying and Mitigating Exploitation of the Denial of Service Vulnerabilities in Content Switching Module](#)
- [Cisco Applied Mitigation Bulletin: Identifying and Mitigating Exploitation of the Multiple Vulnerabilities in Cisco PIX and ASA Appliances and Firewall Services Module](#)
- [IP Spoofing](#)
- [Technical Support & Documentation - Cisco Systems](#)

Updated: Sep 30, 2008

Document ID: 100830