# Configure ASA Access Control List for Various Scenarios

## Contents


**Introduction**

**Prerequisites**

- Requirements
- Components Used

**Background Information**

**Configure**

- Scenario 1. Configure an Ace to Allow Access to a Web Server Located Behind the DMZ
  - Network Diagram
  - Verify
- Scenario 2. Configure an Ace to Allow Access to a Web Server with an Fully Qualified Domain Name (FQDN)
  - Network Diagram
  - Verify
- Scenario 3. Configure an Ace to Allow Access to a Website Only for a Specific Time Duration in a Day
  - Network Diagram
  - Verify
- Scenario 4. Configure an Ace to Block Bridge Protocol Data Units (BPDU) Through an ASA in Transparent Mode
  - Network Diagram
  - Verify
- Scenario 5. Allow Traffic to Pass between Interfaces with the Same Security Level
  - Network Diagram
  - Verify
- Scenario 6. Configure an Ace to Control To-The-Box Traffic
  - Network Diagram
  - Verify

**Logging**

**Troubleshoot**


## Introduction

This document describes how to configure an Access Control List (ACL) on the Adaptive Security Appliance (ASA) for various scenarios.

## Prerequisites

### Requirements

Cisco recommends that you have knowledge of ASA.

### Components Used

The information in this document is based on an ASA software version 8.3 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, ensure that you understand the potential impact of any command.

# Background Information

ACLs are used by the ASA to determine if traffic is permitted or denied. By default, traffic that passes from a lower security level interface to a higher security level interface is denied whereas traffic from a higher security level interface to a lower security level interface is allowed. This behavior can also be overridden with an ACL.

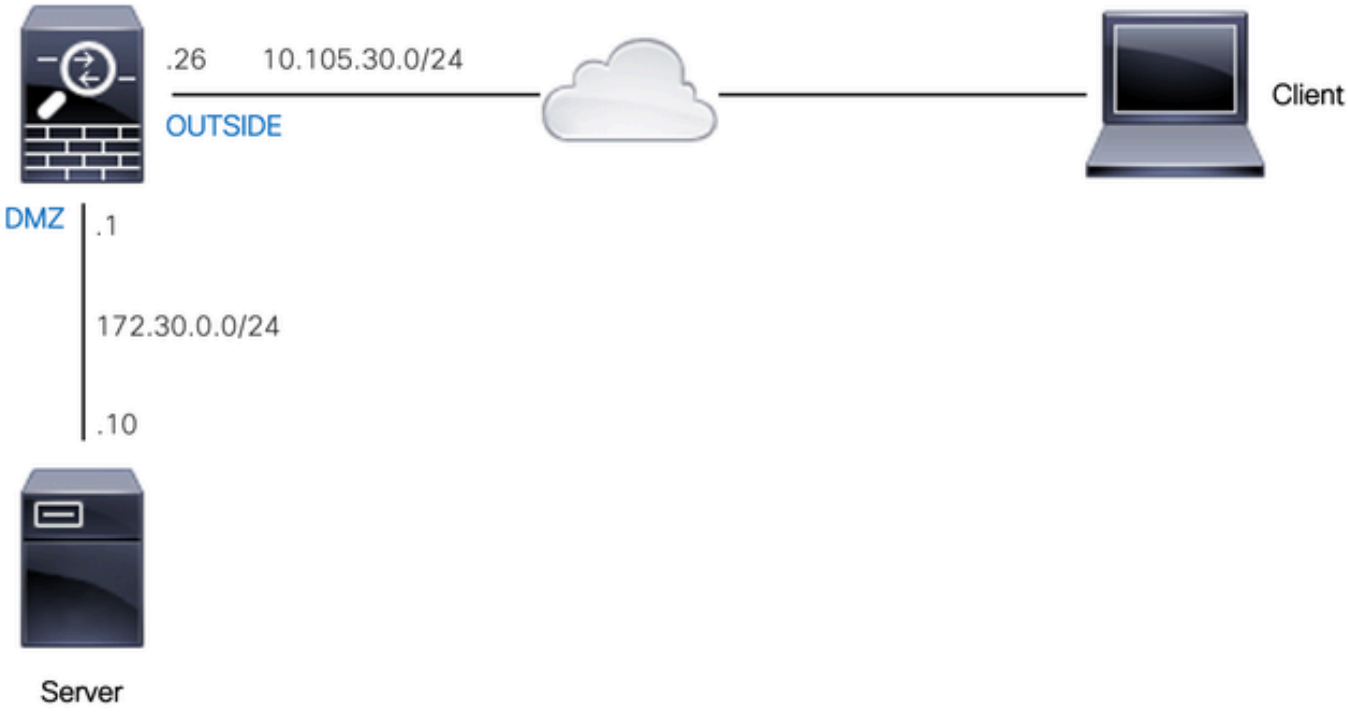
In the presence of NAT rules, in earlier versions of the ASA (8.2 and earlier), the ASA checks the ACL before untranslating the packet based on the NAT rule that was matched. In version 8.3 and later, the ASA untranslates the packet before it checks the ACLs. This means that for an ASA version 8.3 and later, traffic is either permitted or denied based on the real IP address of the host instead of the translated IP address. ACLs are made up of one or more Access Control Entries (ACEs).

# Configure

### Scenario 1. Configure an Ace to Allow Access to a Web Server Located Behind the DMZ

The client on the internet, located behind the outside interface, wants to access a web server hosted behind the DMZ interface listening on TCP ports 80 and 443.

#### Network Diagram



The real IP address of the web server is 172.30.0.10. A static one-to-one NAT rule is configured to allow internet users to access the web server with a translated IP address 10.105.130.27. The ASA performs proxy-arp for 10.105.130.27 on the outside interface by default when a static NAT rule is configured with a translated IP address that falls in the same subnet as the 'outside' interface IP address 10.105.130.26:

```
object network web-server
 nat (dmz,outside) static 10.105.130.27
```

Configure this ACE to allow any source IP address on the internet to connect to the web server only on TCP ports 80 and 443. Assign the ACL to the outside interface in the inbound direction:

```
access-list OUT-IN extended permit tcp any host 172.30.0.10 eq www
access-list OUT-IN extended permit tcp any host 172.30.0.10 eq https
access-group OUT-IN in interface outside
```

**Verify**

Run a packet-tracer command with these fields. Ingress interface on which to trace packet: outside

Protocol: TCP

Source IP address: Any IP address on the internet

Source IP Port: Any ephemeral port

Destination IP address: Translated IP address of the web-server (10.105.130.27)

Destination Port: 80 or 443

```
ciscoasa# packet-tracer input outside tcp 10.0.50.50 1234 10.105.130.27 443

!--- NAT untranslate from 10.105.130.27/443 to 172.30.0.10/443


Phase: 1
Type: UN-NAT
Subtype: static
Result: ALLOW
Config:
object network web-server
nat (dmz,outside) static 10.105.130.27
Additional Information:
NAT divert to egress interface dmz
Untranslate 10.105.130.27/443 to 172.30.0.10/443

!--- The configured ACL is permitting this packet to 172.30.0.10 on TCP port 443


Phase: 2
Type: ACCESS-LIST
```

```
Subtype: log
Result: ALLOW
Config:
access-group OUT-IN in interface outside
access-list OUT-IN extended permit tcp any host 172.30.0.10 eq https
Additional Information:

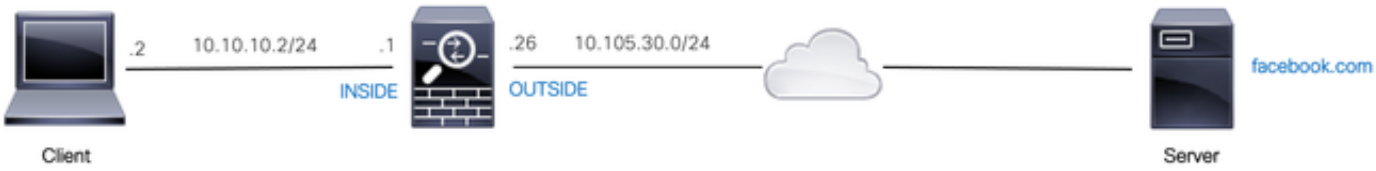
!--- Final result shows allow from the outside interface to the dmz interface

Result:
input-interface: outside
input-status: up
input-line-status: up
output-interface: dmz
output-status: up
output-line-status: up
Action: allow
```

## Scenario 2. Configure an Ace to Allow Access to a Web Server with an Fully Qualified Domain Name (FQDN)

Client with IP address 10.10.10.2 located in the local area network (LAN) is allowed to access facebook.com.

### Network Diagram




Ensure that the DNS server is configured correctly on the ASA:

```
ciscoasa# show run dns
dns domain-lookup outside
dns server-group DefaultDNS
    name-server 10.0.2.2
    name-server 10.0.8.8
```

Configure this network object, FQDN object, and the ACE to allow the client with IP address 10.10.10.2 to access facebook.com.

```
object network obj-10.10.10.2
 host 10.10.10.2

object network obj-facebook.com
 fqdn facebook.com

access-list IN-OUT extended permit ip object obj-10.10.10.2 object obj-facebook.com
```

```
access-group IN-OUT in interface inside
```

### Verify

The output of show dns shows the resolved IP address for the FQDN facebook.com:

```
ciscoasa# show dns

Host                    Flags      Age Type   Address(es)
facebook.com            (temp, OK) 0    IP    10.0.228.35
```

The access list shows the FQDN object as resolved and also shows the resolved IP address:

<#root>

```
ciscoasa# show access-list IN-OUT
access-list IN-OUT; 2 elements; name hash: 0x1b5ff18e
access-list IN-OUT line 1 extended permit ip object obj-10.10.10.2 object obj-facebook.com (hitcnt=1) 0
access-list IN-OUT line 1 extended permit ip host 10.10.10.2 fqdn facebook.com
```

**(resolved)**

```
 0xfea095d7
access-list IN-OUT line 1 extended permit ip host 10.10.10.2 host
```

**10.0.228.35 (facebook.com)**

```
 (hitcnt=1) 0x22075b2a
```

## Scenario 3. Configure an Ace to Allow Access to a Website Only for a Specific Time Duration in a Day

The client located in the LAN is allowed to access a website with IP address 10.0.20.20 daily from 12 PM to 2 PM IST only.

### Network Diagram




Ensure that the time zone is configured correctly on the ASA:

```
ciscoasa# show run clock
```

```
clock timezone IST 5 30
```

Configure a time-range object for the required time duration:

```
time-range BREAK_TIME
 periodic daily 12:00 to 14:00
```

Configure these network objects and ACE to allow any source IP address located in the LAN to access the website only during the time period mentioned in the time-range object named BREAK_TIME:

```
object network obj-website
 host 10.0.20.20

access-list IN-OUT extended permit ip any object obj-website time-range BREAK_TIME
access-group IN-OUT in interface inside
```

**Verify**

The time-range object is active when the clock on the ASA indicates a time that is within the time-range object:

<#root>

```
ciscoasa# show clock
12:03:41.987 IST Mon Oct 4 2021

ciscoasa# show time-range BREAK_TIME

time-range entry: BREAK_TIME (
active
)
   periodic daily 12:00 to 14:00
   used in: IP ACL entry

ciscoasa# show access-list IN-OUT
access-list IN-OUT; 1 elements; name hash: 0x1b5ff18e
access-list IN-OUT line 1 extended permit ip any object obj-website time-range BREAK_TIME (
hitcnt=12
) 0x5a66c8f9
  access-list IN-OUT line 1 extended permit ip any host 10.0.20.20 time-range BREAK_TIME (hitcnt=12) 0x
```

The time-range object as well as the ACE is inactive when the clock on the ASA indicates a time that is

outside of the time-range object:

```
<#root>

ciscoasa# show clock
14:15:44.409 IST Mon Oct 4 2021


ciscoasa# show time-range BREAK_TIME

time-range entry: BREAK_TIME (

inactive

)
   periodic daily 12:00 to 14:00
   used in: IP ACL entry


ciscoasa# show access-list IN-OUT
access-list IN-OUT; 1 elements; name hash: 0x1b5ff18e
access-list IN-OUT line 1 extended permit ip any object obj-website time-range BREAK_TIME (hitcnt=0)

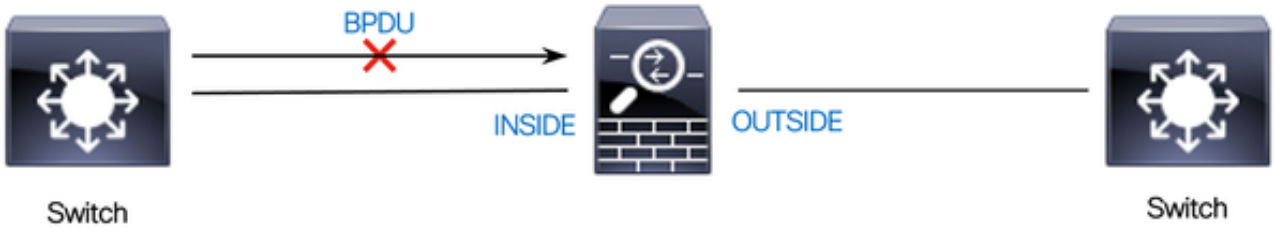
(inactive)

 0x5a66c8f9
access-list IN-OUT line 1 extended permit ip any host 10.0.20.20 time-range BREAK_TIME (hitcnt=0) (inac
```

## Scenario 4. Configure an Ace to Block Bridge Protocol Data Units (BPDU) Through an ASA in Transparent Mode

To prevent loops with the Spanning Tree Protocol (STP), BPDUs are passed through the ASA in transparent mode by default. To block BPDUs, you need to configure an EtherType rule to deny them.

### Network Diagram




Configure the EtherType ACL to block BPDUs from passing through the inside interface of the ASA in the inbound direction as shown here:

```
access-list block-bpdu ethertype deny dsap bpdu
access-list block-bpdu ethertype permit any
access-group block-bpdu in interface inside
```

### Verify

Check the hit count in the access-list to verify that BPDUs are blocked by the ASA:

<#root>

```
ciscoasa# show access-list block-bpdu
access-list block-bpdu; 2 elements
access-list block-bpdu ethertype deny dsap bpdu
```

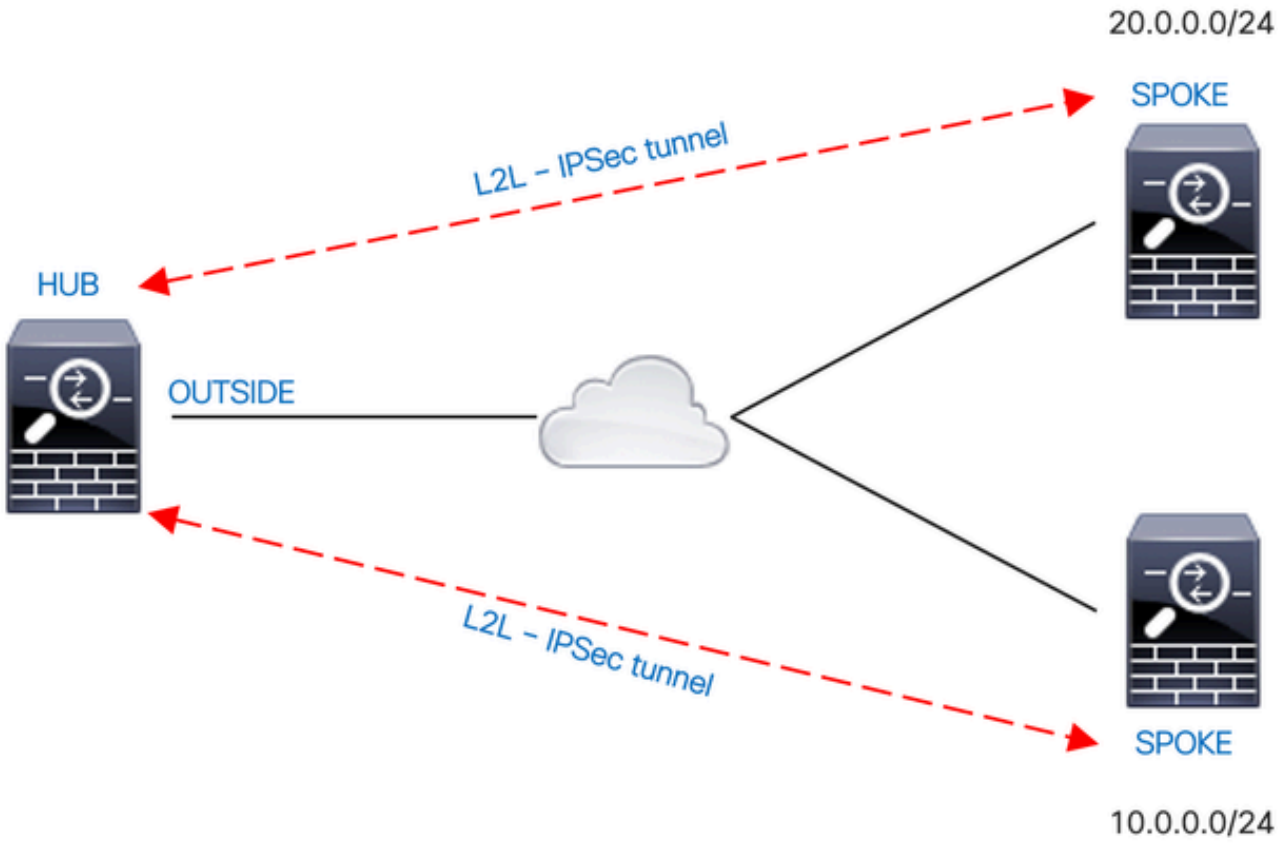
**(hitcount=14)**

```
access-list block-bpdu ethertype permit any (hitcount=48)
```

## Scenario 5. Allow Traffic to Pass between Interfaces with the Same Security Level

### Network Diagram



By default, the traffic that passes between interfaces of the same security level is blocked. To permit communication between interfaces with equal security levels, or to allow traffic to enter and exit the same interface (hairpin/u-turn), use the **same-security-traffic** command in global configuration mode.

This command shows how to permit communication between different interfaces that have the same security level:

```
same-security-traffic permit inter-interface
```

This example shows how to permit communication in and out of the same interface:

```
same-security-traffic permit intra-interface
```

This feature is useful for VPN traffic that enters an interface but is then routed out of that same interface. For example, if you have a hub-and-spoke VPN network where this ASA is the hub and the remote VPN networks are spokes, in order for one spoke to communicate with another spoke, traffic must go to the ASA and then out again to the other spoke.

**Verify**

Without the **same-security-traffic permit inter-interface** command, the output of packet-tracer indicates that the traffic passing between different interfaces of the same security level is blocked due to an implicit rule as shown here:

<#root>

```
!--- The interfaces named 'test' and 'outside' have the same security level of 0


ciscoasa# show nameif
Interface               Name                    Security
GigabitEthernet0/0      inside                  100
GigabitEthernet0/1      dmz                     50
GigabitEthernet0/2      test                    0


GigabitEthernet0/5      outside                 0


Management0/0           mgmt                    0


!--- Traffic between different interfaces of same security level is blocked by an implicit rule


ciscoasa# packet-tracer input test tcp 172.16.20.10 1234 10.0.8.8 443 detailed
```

```
Phase: 2
Type: ACCESS-LIST
Subtype:

Result: DROP


Config:

Implicit Rule


Additional Information:
 Forward Flow based lookup yields rule:
 in id=0x7f9960a2ff90, priority=110, domain=permit, deny=true
    hits=0, user_data=0x0, cs_id=0x0, flags=0x3000, protocol=0
    src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any
    dst ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any, dscp=0x0, nsg_id=none
    input_ifc=test, output_ifc=any

Result:

input-interface: test


input-status: up
input-line-status: up

output-interface: outside


output-status: up
output-line-status: up

Action: drop



Drop-reason: (acl-drop) Flow is denied by configured rule, Drop-location: frame 0x00005638dfd7da57 flow




!--- After running the command 'same-security-traffic permit inter-interface'



ciscoasa# show running-config same-security-traffic
same-security-traffic permit inter-interface


!--- Traffic between different interfaces of same security level is allowed



ciscoasa# packet-tracer input test tcp 172.16.20.10 1234 10.0.8.8 443 detailed

Phase: 3
Type: ACCESS-LIST
Subtype:

Result: ALLOW


Config:
```

```
Implicit Rule


Additional Information:
Forward Flow based lookup yields rule:
in id=0x7f9960a352d0, priority=2, domain=permit, deny=false
hits=2, user_data=0x0, cs_id=0x0, flags=0x3000, protocol=0
src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any
dst ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any, dscp=0x0, nsg_id=none
input_ifc=test, output_ifc=any

Result:

input-interface: test


input-status: up
input-line-status: up

output-interface: outside


output-status: up
output-line-status: up

Action: allow
```

Without the **same-security-traffic permit intra-interface** command, the output of packet-tracer indicates that the traffic passing in and out of the same interface is blocked due to an implicit rule as shown here:

<#root>

```
!--- Traffic in and out of the same interface is blocked by an implicit rule



ciscoasa# packet-tracer input outside tcp 10.0.0.10 1234 10.1.0.10 443 detailed

Phase: 3
Type: ACCESS-LIST
Subtype:

Result: DROP


Config:

Implicit Rule


Additional Information:
Forward Flow based lookup yields rule:
in id=0x7f9960a32f30, priority=111, domain=permit, deny=true
hits=0, user_data=0x0, cs_id=0x0, flags=0x4000, protocol=0
src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any
dst ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any, dscp=0x0, nsg_id=none
input_ifc=outside, output_ifc=outside
```

Result:

**input-interface: outside**

input-status: up
input-line-status: up

**output-interface: outside**

output-status: up
output-line-status: up

**Action: drop**

**Drop-reason: (acl-drop) Flow is denied by configured rule, Drop-location: frame 0x00005638dfd7da57 flow**

**!--- After running the command 'same-security-traffic permit intra-interface'**

ciscoasa# show running-config same-security-traffic
same-security-traffic permit intra-interface

**!--- Traffic in and out of the same interface is allowed**

Phase: 3
Type: ACCESS-LIST
Subtype:

**Result: ALLOW**

Config:

**Implicit Rule**

Additional Information:
 Forward Flow based lookup yields rule:
 in id=0x7f99609291c0, priority=3, domain=permit, deny=false
    hits=1, user_data=0x0, cs_id=0x0, flags=0x4000, protocol=0
    src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any
    dst ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any, dscp=0x0, nsg_id=none
    input_ifc=outside, output_ifc=outside

Result:

**input-interface: outside**

input-status: up
input-line-status: up

```
output-interface: outside


output-status: up
output-line-status: up

Action: allow
```

## Scenario 6. Configure an Ace to Control To-The-Box Traffic

The control-plane keyword specifies if the ACL is used to control to-the-box traffic. Access control rules for to-the-box management traffic (defined by such commands as **http**, **ssh**, or **telnet**) have higher precedence than a management access rule applied with the control-plane option. Therefore, such permitted management traffic must be allowed to come in even if explicitly denied by the to-the-box ACL.

Unlike regular access rules, there is no implicit deny at the end of a set of management rules for an interface. Instead, any connection that does not match a management access rule is then evaluated by regular access control rules. Alternatively, you can use Internet Control Message Protocol (ICMP) rules to control ICMP traffic to the device.

### Network Diagram




An ACL is configured with the control-plane keyword to block to-the-box traffic sourced from the IP address 10.65.63.155 and destined to the outside interface IP address of the ASA.

```
access-list control-plane-test extended deny ip host 10.65.63.155 any
access-group control-plane-test in interface outside control-plane
```

### Verify

Check the hit count in the access list to verify that traffic is blocked by the ACL:

<#root>

```
ciscoasa# show access-list control-plane-test
access-list control-plane-test; 1 elements; name hash: 0x6ff5e700
access-list control-plane-test line 1 extended deny ip host 10.65.63.155 any (

hitcnt=4

) 0xedad4c6f
```

Syslog messages indicate that traffic is dropped on the identity interface:

<#root>

```
Dec 27 2021 13:19:44: %ASA-4-106023: Deny tcp src outside:10.65.63.155/54108 dst
identity
:10.105.130.26/8000 by access-group "
control-plane-test
" [0xedad4c6f, 0x0]
Dec 27 2021 13:19:45: %ASA-4-106023: Deny tcp src outside:10.65.63.155/54108 dst identity:10.105.130.26
Dec 27 2021 13:19:46: %ASA-4-106023: Deny tcp src outside:10.65.63.155/54108 dst identity:10.105.130.26
Dec 27 2021 13:19:47: %ASA-4-106023: Deny tcp src outside:10.65.63.155/54108 dst identity:10.105.130.26
```

# Logging

The log keyword sets logging options when an ACE matches a packet for network access (an ACL applied with the **access-group** command). If you enter the log keyword without any arguments, you enable system log message 106100 at the default level (6) and for the default interval (300 seconds). If you do not enter the log keyword, then the default system log message 106023 is generated for denied packets. Log options are:

- level — A severity level between 0 and 7. The default is 6 (informational). If you change this level for an active ACE, the new level applies to new connections; existing connections continue to be logged at the previous level.
- interval secs — The time interval in seconds between syslog messages, from 1 to 600. The default is 300. This value is also used as the timeout value for deleting an inactive flow from the cache used to collect drop statistics.
- disable — Disables all ACE logging.
- default — Enables logging to message 106023. This setting is the same as not including the log option.

Syslog message 106023:

```
Message:
%ASA-4-106023: Deny protocol src [interface_name :source_address /source_port ] [([idfw_user |FQDN_stri
```

Explanation:

A real IP packet was denied by the ACL. This message appears even if you do not have the log option enabled for an ACL. The IP address is the real IP address instead of the values that display through NAT. Both user identity information and FQDN information is provided for the IP addresses if a matched one is found. The Secure Firewall ASA logs either identity information (domain\user) or FQDN (if the username is not available). If the identity information or FQDN is available, the Secure Firewall ASA logs this information for both the source and destination.

Example:

```
Dec 27 2021 14:58:25: %ASA-4-106023: Deny tcp src outside:10.65.63.155/56166 dst inside:10.5.0.30/8000
Dec 27 2021 14:58:26: %ASA-4-106023: Deny tcp src outside:10.65.63.155/56166 dst inside:10.5.0.30/8000
Dec 27 2021 14:58:27: %ASA-4-106023: Deny tcp src outside:10.65.63.155/56166 dst inside:10.5.0.30/8000
```

Syslog message 106100:

```
Message:

%ASA-6-106100: access-list acl_ID {permitted | denied | est-allowed} protocol interface_name /source_ad
```

Explanation:

The initial occurrence or the total number of occurrences during an interval are listed. This message provides more information than message 106023, which only logs denied packets, and does not include the hit count or a configurable level.

When an access-list line has the log argument, it is expected that this message ID can be triggered because of a non-synchronized packet arrives at the Secure Firewall ASA and evaluated by the access list. For example, if an ACK packet is received on the Secure Firewall ASA (for which no TCP connection exists in the connection table), the Secure Firewall ASA can generate message 106100, indicating that the packet was permitted; however, the packet is later correctly dropped because of no matching connection.

The list describes the message values:

- permitted | denied | est-allowed— These values specify if the packet was permitted or denied by the ACL. If the value is est-allowed, the packet was denied by the ACL but was allowed for an already established session (for example, an internal user is allowed to access the Internet, and responding packets that would normally be denied by the ACL are accepted).
- protocol — TCP, UDP, ICMP, or an IP protocol number.
- interface_name — The interface name for the source or destination of the logged flow. The VLAN interfaces are supported.
- source_address — The source IP address of the logged flow. The IP address is the real IP address instead of the values that display through NAT.
- dest_address — The destination IP address of the logged flow. The IP address is the real IP address instead of the values that display through NAT.
- source_port — The source port of the logged flow (TCP or UDP). For ICMP, the number after the source port is the message type.
- idfw_user — The user identity username, with the domain name that is added to the existing syslog when the Secure Firewall ASA can find the username for the IP address.
- sg_info — The security group tag that is added to the syslog when the Secure Firewall ASA can find a security group tag for the IP address. The security group name is displayed with the security group tag, if available.
- dest_port — The destination port of the logged flow (TCP or UDP). For ICMP, the number after the destination port is the ICMP message code, which is available for some message types. For type 8, it is always 0. For a list of ICMP message types, see the URL: Internet Control Message Protocol (ICMP) Parameters.
- hit-cnt number — The number of times this flow was permitted or denied by this ACL entry in the configured time interval. The value is 1 when the Secure Firewall ASA generates the first message for this flow.
- first hit — The first message generated for this flow.
- number - second interval—The interval in which the hit count is accumulated. Set this interval with

the **access-list** command with the **interval** option.

- hash codes — Two are always printed for the object group ACE and the constituent regular ACE. Values are determined on which ACE the packet hit. To display these hash codes, enter the **show-access list** command.

Example:

```
Dec 27 2021 15:09:58: %ASA-6-106100: access-list OUT-IN permitted tcp outside/10.65.63.155(56261) -> in
Dec 27 2021 15:10:15: %ASA-6-106100: access-list OUT-IN permitted tcp outside/10.65.63.155(56266) -> in
Dec 27 2021 15:10:55: %ASA-6-106100: access-list OUT-IN permitted tcp outside/10.65.63.155(56270) -> in
```

# Troubleshoot

There is currently no specific troubleshooting information available for this configuration.