CHAPTER 13

Configuring Dynamic ARP Inspection

This chapter describes how to configure dynamic Address Resolution Protocol (ARP) inspection (DAI).

This chapter includes the following sections:

Information About DAI

This section includes the following topics:

About ARP

ARP provides IP communication within a Layer 2 broadcast domain by mapping an IP address to a MAC address. For example, host B wants to send information to host A but does not have the MAC address of host A in its ARP cache. In ARP terms, host B is the sender and host A is the target.

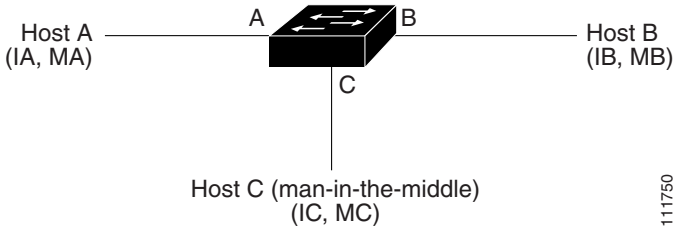
To get the MAC address of host A, host B generates a broadcast message for all hosts within the broadcast domain to obtain the MAC address associated with the IP address of host A. All hosts within the broadcast domain receive the ARP request, and host A responds with its MAC address.

About ARP Spoofing Attacks

In an ARP spoofing attack, a host allows an unsolicited ARP response to update its cache so that traffic is directed through the attacker until it is discovered and the information in the ARP cache is corrected.

An ARP spoofing attack can affect hosts, switches, and routers connected to your Layer 2 network by sending false information to their ARP caches. Figure 13-1 shows an example of ARP cache poisoning.

Figure 13-1 ARP Cache Poisoning



In Figure 13-1, hosts A, B, and C are connected to the device on interfaces A, B, and C, all of which are on the same subnet. Their IP and MAC addresses are shown in parentheses. For example, host A uses IP address IA and MAC address MA.

When host A needs to send IP data to host B, it broadcasts an ARP request for the MAC address associated with IP address IB. When the device and host B receive the ARP request, they add a binding to their ARP caches for a host with the IP address IA and a MAC address MA.

When host B responds, the device and host A update their ARP caches with a binding for a host with the IP address IB and the MAC address MB.

Host C can spoof host A and B by broadcasting the following forged ARP responses:

- one for a host with an IP address of IA and a MAC address of MC
- one for a host with the IP address of IB and a MAC address of MC.

Host B then uses MC as the destination MAC address for traffic that was intended for IA, which means that host C intercepts that traffic. Likewise, host A and the device use MC as the destination MAC address for traffic intended for IB.

Because host C knows the authentic MAC addresses for IA and IB, it can forward the intercepted traffic.

About DAI and ARP Spoofing

DAI is used to validate ARP requests and responses as follows:

- Intercepts all ARP requests and responses on untrusted ports.
- Verifies that a packet has a valid IP-to-MAC address binding before updating the ARP cache or forwarding the packet.
- Drops invalid ARP packets.

DAI can determine the validity of an ARP packet based on valid IP-to-MAC address bindings stored in a Dynamic Host Configuration Protocol (DHCP) snooping binding database. This database is built by DHCP snooping when it is enabled on the VLANs and on the device. It may also contain static entries that you have created.

If an ARP packet is received on a trusted interface, the device forwards the packet without any checks. On untrusted interfaces, the device forwards the packet only if it is valid. For more information about trusted interfaces, see the Interface Trust and Network Security, page 13-3.

You can enable or disable validation of ARP packets for destination MAC address, source MAC address, and IP address. For more information, see the "Validating ARP Packets" section on page 13-13.

Interface Trust and Network Security

DAI identifies interfaces as trusted or untrusted.

In a typical network, interfaces are configured as follows:

- Untrusted—Interfaces that are connected to hosts

  Packets are validated by DAI.

- Trusted—Interfaces that are connected to devices

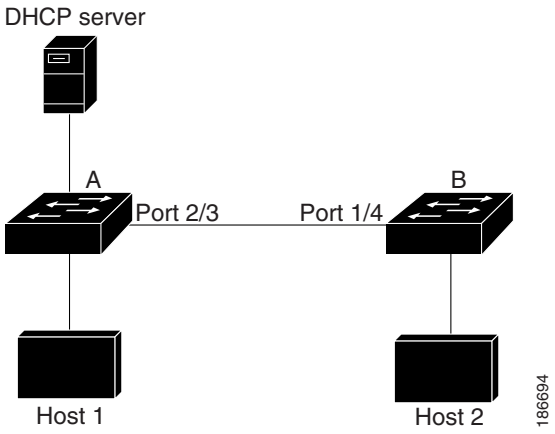
  Packets bypass all DAI validation checks.

With this configuration, all ARP packets that enter the network from a device bypass the security check. No other validation is needed at any other place in the VLAN or in the network. For information about configuring a trusted interface, see the "Configuring a Trusted vEthernet Interface" section on page 13-6.

Caution Use the trust state configuration carefully. Configuring interfaces as untrusted when they should be trusted can result in a loss of connectivity.

In Figure 13-2, assume that both device A and device B are running DAI on the VLAN that includes host 1 and host 2. If host 1 and host 2 acquire their IP addresses from the DHCP server connected to device A, only device A binds the IP-to-MAC address of host 1. If the interface between device A and device B is untrusted, the ARP packets from host 1 are dropped by device B and connectivity between host 1 and host 2 is lost.

Figure 13-2 ARP Packet Validation on a VLAN Enabled for DAI



If you configure interfaces as trusted when they should be untrusted, you may open a security hole in a network. If device A is not running DAI, host 1 can easily poison the ARP cache of device B (and host 2, if you configured the link between the devices as trusted). This condition can occur even though device B is running DAI.

DAI ensures that hosts (on untrusted interfaces) connected to a device that runs DAI do not poison the ARP caches of other hosts in the network; however, DAI does not prevent hosts in other portions of the network from poisoning the caches of the hosts that are connected to a device that runs DAI.

Note Depending on your network setup, you may not be able to validate a given ARP packet on all devices in the VLAN.

Prerequisites for DAI

The following are prerequisite to configuring DAI.

- You are familiar with the following:
  - ARP

    For more information, see IETF Standard RFC-826, *An Ethernet Address Resolution Protocol* (http://tools.ietf.org/html/rfc826).
  - DHCP Snooping

    For more information, see Configuring DHCP Snooping, page 12-1.
- The software running on your Cisco Nexus 1000V supports DAI.
- The VEM feature level is updated to a release that supports DAI.

  For more information about setting the VEM feature level, see the *Cisco Nexus 1000V Software Upgrade Guide, Release 4.2(1)SV1(4b)*.

Guidelines and Limitations

DAI has the following configuration guidelines and limitations:

- DAI is an ingress security feature and does not perform any egress checking.
- DAI is not effective when the host is connected to a device that does not support DAI or that does not have DAI enabled. To prevent attacks that are limited to a single Layer 2 broadcast domain, you should separate a domain with DAI from those without DAI. This separation secures the ARP caches of hosts in the domain with DAI.
- DAI verifies IP-to-MAC address bindings in incoming ARP requests and ARP responses. If you have not configured static entries, then DHCP snooping must be enabled on the same VLANs on which you configure DAI. For more information, see the "Configuring DHCP Snooping" section on page 12-4.

- DAI is supported on vEthernet interfaces and private VLAN ports.
- If you want DAI to use dynamic IP-MAC address bindings to determine if ARP packets are valid, ensure that DHCP snooping is configured. For more information, see the "Configuring DHCP Snooping" section on page 12-4).
- Virtual Service Domain (VSD) service VM ports are trusted ports by default. Even if you configure VSD ports as untrusted, they still appear as trusted ports to DAI.

Default Settings

Table 13-1 lists the DAI defaults.

Table 13-1 Default DAI Settings

Parameters	Default
VLAN	VLANs are not configured for DAI.
Trust state of vEthernet interfaces not in a VSD	Untrusted
Trust state of vEthernet Interfaces in a VSD	Trusted
Trust state of Ethernet port channels	Trusted
Incoming ARP packet rate limit for untrusted interfaces	15 packets per second (pps)
Incoming ARP packet rate limit for trusted interfaces	Unlimited
Rate limit burst interval	1 second
Detecting and Recovering DAI error-disabled interfaces	Error-disabled detection and recovery is not configured.
Validation checks	No checks are performed.
VLAN statistics	ARP request and response statistics.

Configuring DAI

This section includes the following topics:

Configuring a VLAN for DAI

Use this procedure to configure a VLAN or a list of VLANs for DAI.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- You are logged in to the CLI in EXEC mode.
- By default, VLANs are not configured for DAI.
- You have already enabled DHCP snooping. For more information, see the "Enabling or Disabling the DHCP Feature" section on page 12-5.
- You know which VLANs you want to configure for DAI and they have already been created.

SUMMARY STEPS

1. **config t**
2. **[no] ip arp inspection vlan** *list*
3. **show ip arp inspection vlan** *list*
4. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Places you into the CLI Global Configuration mode.

Step 2	**ip arp inspection vlan** *list* **Example:** `switch(config)# ip arp inspection vlan 13`	Configures the specified VLAN or list of VLANs for DAI.
Step 3	**show ip arp inspection vlan** *list* **Example:** `switch(config)# show ip arp inspection vlan 13`	(Optional) Shows the DAI status for the specified list of VLANs.
Step 4	**copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Configuring a Trusted vEthernet Interface

Use this procedure to configure a trusted vEthernet interface.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- You are logged in to the CLI in EXEC mode.
- By default, vEthernet interfaces are untrusted, unless they are part of a VSD.
- If an interface is untrusted, all ARP requests and responses are verified for a valid IP-MAC address binding before the local cache is updated and the packet forwarded. If a packet has an invalid IP-MAC address binding, it is dropped.
- ARP packets received on a trusted interface are forwarded but not checked.
- You can configure a trusted interface on either of the following:
  - the interface, itself
  - the existing port profile that the interface is assigned to

    If configuring a trusted interface on the port profile, it has already been created and you know its name.

SUMMARY STEPS

1. **config t**
2. **interface vethernet** *interface-number*

   port-profile *profilename*
3. **[no] ip arp inspection trust**
4. **show ip arp inspection interface** *type slot/number*

   show port-profile *profilename*
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Places you into the CLI Global Configuration mode.

Step 2	**interface vethernet** *interface-number*
	Example:
	`switch(config)# interface vethernet 3`
	`switch(config-if)#`
		Places you into the CLI Interface Configuration mode, for the specified vEthernet interface.

	port-profile *profilename*
	Example:
	`switch(config)# port-profile vm-data`
	`switch(config-port-prof)#`
		Places you into the CLI Port Profile Configuration mode for the specified port profile.

Step 3	**ip arp inspection trust** **Example:** `switch(config-if)# ip arp inspection trust`	Configures the interface as a trusted ARP interface.

	Command	Purpose
	ip arp inspection trust **Example:** `switch(config-port-prof)# ip arp inspection trust`	Configures the interfaces assigned to the port profile as trusted ARP interfaces.
Step 4	**show ip arp inspection interface vethernet** *interface-number* **Example:** `switch(config-if)# show ip arp inspection interface vethernet 2`	(Optional) Displays the trusted state and the ARP packet rate for the specified interface.
	show port-profile *profilename* **Example:** `switch(config)# show port-profile vm-data`	(Optional) Displays the port profile configuration including the ARP trusted state.
Step 5	**copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Resetting a vEthernet Interface to Untrusted

Use this procedure to remove a trusted designation from a vEthernet interface, returning it to the default untrusted designation.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- You are logged in to the CLI in EXEC mode.
- By default, vEthernet interfaces are untrusted, unless they are part of a VSD.
- If an interface is untrusted, all ARP requests and responses are verified for a valid IP-MAC address binding before the local cache is updated and the packet forwarded. If a packet has an invalid IP-MAC address binding, it is dropped.

SUMMARY STEPS

1. **config t**
2. **interface vethernet** *interface-number*
3. **default ip arp inspection trust**
4. **show ip arp inspection interface** *type slot/number*
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Places you into the CLI Global Configuration mode.

Step 2	**interface vethernet** *interface-number*
	Example:
	`switch(config)# interface vethernet 3`
	`switch(config-if)#`
		Places you into the CLI Interface Configuration mode, for the specified vEthernet interface.

Step 3	**default ip arp inspection trust** **Example:** `switch(config-if)# default ip arp inspection trust`	Removes the trusted designation from the interface and returns it to the default untrusted state.
Step 4	**show ip arp inspection interface vethernet** *interface-number* **Example:** `switch(config-if)# show ip arp inspection interface vethernet 3`	(Optional) Displays the trusted state and the ARP packet rate for the specified interface.
Step 5	**copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Configuring DAI Rate Limits

Use this procedure to set the rate limit of ARP requests and responses.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- You are logged in to the CLI in EXEC mode.
- Because of their aggregation, trunk ports should be configured with higher rate limit.
- Once the rate of incoming packets exceeds the configured rate, the interface is automatically put into an errdisable state.
- The default DAI rate limits are as follows:
  - Untrusted interfaces = 15 packets per second
  - Trusted interfaces = unlimited
  - Burst interval = 1 second
- You can configure the rate limits for an interface on either of the following:
  - the interface, itself
  - the existing port profile that the interface is assigned to

    If configuring the port profile, it has already been created and you know its name.

SUMMARY STEPS

1. **config t**
2. **interface vethernet** *interface-number*

   port-profile *profilename*
3. **ip arp inspection limit** {**rate** *pps* [**burst interval** *bint*] | **none**}
4. **show running-config dhcp**
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Places you into the CLI Global Configuration mode.

Step 2	**interface vethernet** *interface-number*
	Example:
	`switch(config)# interface vethernet 3`
	`switch(config-if)#`
		Places you into the CLI Interface Configuration mode, for the specified vEthernet interface.

	port-profile *profilename*
	Example:
	`switch(config)# port-profile vm-data`
	`switch(config-port-prof)#`
		Places you into the CLI Port Profile configuration mode for the specified port profile.

Step 3	**ip arp inspection limit** {**rate** *pps* [**burst interval** *bint*]	**none**}
		Example:
		`switch(config-if)# ip arp inspection limit rate 30`
		Example:
		`switch(config-port-prof)# ip arp inspection limit rate 30`
			Configures the specified ARP inspection limit on the interface or the port profile as follows:
			• **rate**: allowable values are between 1 and 2048 packets per second (pps)
			– Untrusted interface default = 15 packets per second
			– Trusted interface default = unlimited
			• **burst interval**: allowable values are between 1 and 15 seconds (default = 1 second).
			• **none**: unlimited number of packets per second

Step 4	**show running-config dhcp** **Example:** `switch(config)# show running-config dhcp`	(Optional) Displays the DHCP snooping configuration, including the DAI configuration.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Resetting DAI Rate Limits to Default Values

Use this procedure to set the rate limit of ARP requests and responses to the defaults, removing any configured values.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- You are logged in to the CLI in EXEC mode.
- The default DAI rate limits are as follows:
  - Untrusted interfaces = 15 packets per second
  - Trusted interfaces = unlimited
  - Burst interval = 1 second
- You can configure the rate limits for an interface on either of the following:
  - the interface, itself
  - the existing port profile that the interface is assigned to

    If configuring the port profile, it has already been created and you know its name.

SUMMARY STEPS

1. **config t**
2. **interface vethernet** *interface-number*
3. **default ip arp inspection limit** {**rate** *pps* [**burst interval** *bint*] | **none**}
4. **show running-config dhcp**
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Places you into the CLI Global Configuration mode.

Step 2	**interface vethernet** *interface-number*
	Example:
	`switch(config)# interface vethernet 3`
	`switch(config-if)#`
		Places you into the CLI Interface Configuration mode, for the specified vEthernet interface.

	Command	Purpose
Step 3	**default ip arp inspection limit** {**rate** *pps* [**burst interval** *bint*] | **none**} **Example:** `switch(config-if)# default ip arp inspection limit rate`	Removes the configured DAI rate limits from the interface and returns them to the default values. • **rate**: – Untrusted interface default = 15 packets per second – Trusted interface default = unlimited • **burst interval**: default = 1 second • **none**: unlimited number of packets per second
Step 4	**show running-config dhcp** **Example:** `switch(config)# show running-config dhcp`	(Optional) Displays the DHCP snooping configuration, including the DAI rate limits.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Detecting and Recovering Error-Disabled Interfaces

Use this procedure to configure the detection and recovery of error-disabled interfaces.

BEFORE YOU BEGIN

Before beginning this procedures, you must know or do the following:

- You are logged in to the CLI in EXEC mode.
- By default, interfaces are not configured for DAI error-disabled recovery.
- To manually recover an interface from the error-disabled state, use the following command sequence.

  1. **shutdown**

  2. **no shutdown**

SUMMARY STEPS

1. **config t**
2. **[no] errdisable detect cause arp-inspection**
3. **[no] errdisable recovery cause arp-inspection**
4. **errdisable recovery interval** *timer-interval*
5. **show running-config | include errdisable**
6. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Places you into the CLI Global Configuration mode.

Step 2	**errdisable detect cause arp-inspection** **Example:** `switch(config)# errdisable detect cause arp-inspection`	Configures the detection of interfaces that have been error-disabled by ARP inspection. The **no** option disables the detection.
Step 3	**errdisable recovery cause arp-inspection** **Example:** `switch(config)# errdisable recovery cause arp-inspection`	Configures the recovery of interfaces that have been error-disabled by ARP inspection.
Step 4	**errdisable recovery interval** *timer-interval* **Example:** `switch(config)# errdisable recovery interval 30`	Configures the recovery interval for interfaces that have been error-disabled by ARP inspection. **timer-interval**: allowable values are between 30 and 65535 seconds.
Step 5	**show running-config | include errdisable** **Example:** `switch(config)# show running-config | include errdisable`	(Optional) Displays the errdisable configuration.
Step 6	**copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Validating ARP Packets

Use this procedure to configure the validation of ARP packets.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- You are logged in to the CLI in EXEC mode.
- You can enable validation of the following, which are disabled by default:
  - Destination MAC address

    Checks the destination MAC address in the Ethernet header against the target MAC address in the ARP body, and drops packets with an invalid MAC address.
  - IP address

    Checks the ARP body for invalid and unexpected IP addresses, including 0.0.0.0, 255.255.255.255, and any IP multicast address. Sender IP addresses are checked in both ARP requests and responses. Target IP addresses are checked only in ARP responses.
  - Source MAC address

Checks the source MAC address in the Ethernet header against the sender MAC address in the ARP body for ARP requests and responses, and drops packets with invalid MAC addresses.

- Whenever you configure a validation, any previous validation configuration is overwritten.

SUMMARY STEPS

1. **config t**
2. [**no**] **ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]}
3. **show running-config dhcp**
4. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Places you into the CLI Global Configuration mode.

Step 2	**ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]} **Example:** `switch(config)# ip arp inspection validate src-mac dst-mac ip`	Enables the specified validation and overwrites any existing validation that was previously saved: • Source MAC • Destination MAC • IP You can specify all three of these validations but you must specify at least one. Use the no option to disable a validation.
Step 3	**show running-config dhcp** **Example:** `switch(config)# show running-config dhcp`	(Optional) Displays the DHCP snooping configuration, including the DAI configuration.
Step 4	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Verifying the DAI Configuration

To display and verify the DAI configuration, use the following commands:

Command	Purpose
show running-config dhcp	Displays the DAI configuration.
show ip arp inspection	Displays the status of DAI.

Command	Purpose
show ip arp inspection interface vethernet *interface-number*	Displays the trust state and ARP packet rate for a specific interface.
show ip arp inspection vlan *vlan-ID*	Displays the DAI configuration for a specific VLAN.

For detailed information about command output, see the *Cisco Nexus 1000V Command Reference, Release 4.2(1)SV1(4a)*.

Monitoring DAI

To monitor DAI, use the following commands:

Command	Purpose
show ip arp inspection statistics	Displays DAI statistics.
show ip arp inspection statistics vlan	Displays DAI statistics for a specified VLAN.
clear ip arp inspection statistics	Clears DAI statistics.

For detailed information about command output, see the *Cisco Nexus 1000V Command Reference, Release 4.2(1)SV1(4a)*.

Example DAI Configuration

This example shows how to configure DAI in a network with two VEMs:

- One VEM is hosting an authentic web server and a DHCP server.
- The other VEM is hosting a client virtual machine (VM 1) and a virtual machine (VM 2) with a rogue web server. VM 1 is connected to vEthernet interface 3, which is untrusted by default, and belongs to VLAN 1. VM 2 is connected to vEthernet 10 and VLAN 1.

Without DAI enabled, VM 2 can spoof the ARP cache in VM 1 by sending a packet even though an ARP request was not generated. In this case, the packet directs VM 1 to send its traffic to the VM 2 web server instead of the authentic web server.

If DAI is enabled when VM2 attempts to spoof the ARP cache in VM1, the unsolicited ARP packet sent by VM 2 is dropped because DAI detects the invalid IP-to-MAC address binding. The attempt to spoof the ARP cache fails, and VM 1 connects to the authentic web server.

Note DAI depends on the DHCP snooping database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically-assigned IP addresses. For configuration information, see Chapter 12, “Configuring DHCP Snooping.”

The following steps are used to configure DAI for this example:

Step 1 Enable DAI on VLAN 1 and verify the configuration.

```
n1000v# config t
n1000v(config)# ip arp inspection vlan 1
n1000v(config)# show ip arp inspection vlan 1

Source Mac Validation      : Disabled
Destination Mac Validation : Disabled
IP Address Validation      : Disabled

Vlan : 1
-----------
Configuration   : Enabled
Operation State : Active
n1000v(config)#
```

Step 2 Check the statistics before and after DAI processes any packets.

```
n1000v# show ip arp inspection statistics vlan 1

Vlan : 1
-----------
ARP Req Forwarded  = 0
ARP Res Forwarded  = 0
ARP Req Dropped    = 0
ARP Res Dropped    = 0
DHCP Drops         = 0
DHCP Permits       = 0
SMAC Fails-ARP Req = 0
SMAC Fails-ARP Res = 0
DMAC Fails-ARP Res = 0
IP Fails-ARP Req   = 0
IP Fails-ARP Res   = 0
n1000v#
```

If VM 1 sends out two ARP requests with an IP address of 10.0.0.1 and a MAC address of 0002.0002.0002, both requests are permitted, as shown in the following command output:

```
n1000v# show ip arp inspection statistics vlan 1

Vlan : 1
-----------
ARP Req Forwarded  = 2
ARP Res Forwarded  = 0
ARP Req Dropped    = 0
ARP Res Dropped    = 0
DHCP Drops         = 0
DHCP Permits       = 2
SMAC Fails-ARP Req = 0
SMAC Fails-ARP Res = 0
DMAC Fails-ARP Res = 0
IP Fails-ARP Req   = 0
IP Fails-ARP Res   = 0
```

If VM 2 tries to send an ARP request with an IP address of 10.0.0.3, the packet is dropped and an error message is logged.

```
00:12:08: %SW_DAI-4-DHCP_SNOOPING_DENY: 2 Invalid ARPs (Req) on vEthernet3, vlan
1.([0002.0002.0002/10.0.0.3/0000.0000.0000/0.0.0.0/02:42:35 UTC Fri Jul 13 2008])
```

The statistics display as follows:

```
n1000v# show ip arp inspection statistics vlan 1
n1000v#

Vlan : 1
-----------
ARP Req Forwarded  = 2
ARP Res Forwarded  = 0
ARP Req Dropped    = 2
ARP Res Dropped    = 0
DHCP Drops         = 2
DHCP Permits       = 2
SMAC Fails-ARP Req = 0
SMAC Fails-ARP Res = 0
DMAC Fails-ARP Res = 0
IP Fails-ARP Req   = 0
IP Fails-ARP Res   = 0
n1000v#
```

Additional References

For additional information related to implementing DAI, see the following sections:

- Related Documents, page 13-17
- Standards, page 13-17

Related Documents

Related Topic	Document Title
DHCP snooping	Configuring DHCP Snooping, page 12-1
DAI and DHCP commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco Nexus 1000V Command Reference, Release 4.2(1)SV1(4a)*

Standards

Standards	Title
RFC-826	*An Ethernet Address Resolution Protocol* (http://tools.ietf.org/html/rfc826)

Feature History for DAI

Table 13-2 lists the release history for the DAI feature.

Table 13-2 Feature History for DAI

Feature Name	Releases	Feature Information
DAI	4.0(4)SV1(2)	This feature was introduced.