Betrieb und Fehlerbehebung bei DHCP-Snooping auf Catalyst Switches der Serie 9000

Inhalt

Einleitung

In diesem Dokument wird beschrieben, wie DHCP Snooping auf Switches der Serie Catalyst 9000 betrieben und auf diese Weise Fehler behoben werden.

Voraussetzungen

Anforderungen

Cisco empfiehlt, dass Sie über Kenntnisse in folgenden Bereichen verfügen:

- Architektur der Catalyst Switches der Serie 9000
- Cisco IOS® XE Software-Architektur

Verwendete Komponenten

Die Informationen in diesem Dokument basierend auf folgenden Software- und Hardware-Versionen:

- C9200
- C9300
- C9400
- C9500
- C9600

Cisco IOS® XE 16.12.X

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die möglichen Auswirkungen aller Befehle kennen.

> **Hinweis**: Die zur Aktivierung dieser Funktionen auf anderen Cisco Plattformen verwendeten Befehle finden Sie im entsprechenden Konfigurationsleitfaden.

Hintergrundinformationen

DHCP-Snooping

Dynamic Host Configuration Protocol (DHCP) Snooping ist eine Sicherheitsfunktion, mit der der DHCP-Datenverkehr überprüft wird, um schädliche DHCP-Pakete zu blockieren. Sie fungiert als Firewall zwischen nicht vertrauenswürdigen Benutzer-Ports und DHCP-Server-Ports im Netzwerk, um böswillige DHCP-Server im Netzwerk zu verhindern, da dies zu einer Diensteverweigerung führen kann.

DHCP-Snooping

DHCP-Snooping arbeitet mit dem Konzept vertrauenswürdiger und nicht vertrauenswürdiger Schnittstellen. Über den Pfad des DHCP-Datenverkehrs überprüft der Switch die an den Schnittstellen empfangenen DHCP-Pakete und verfolgt die erwarteten DHCP-Server-Pakete (OFFER & ACK) über vertrauenswürdige Schnittstellen. Mit anderen Worten blockieren nicht vertrauenswürdige Schnittstellen DHCP-Serverpakete.

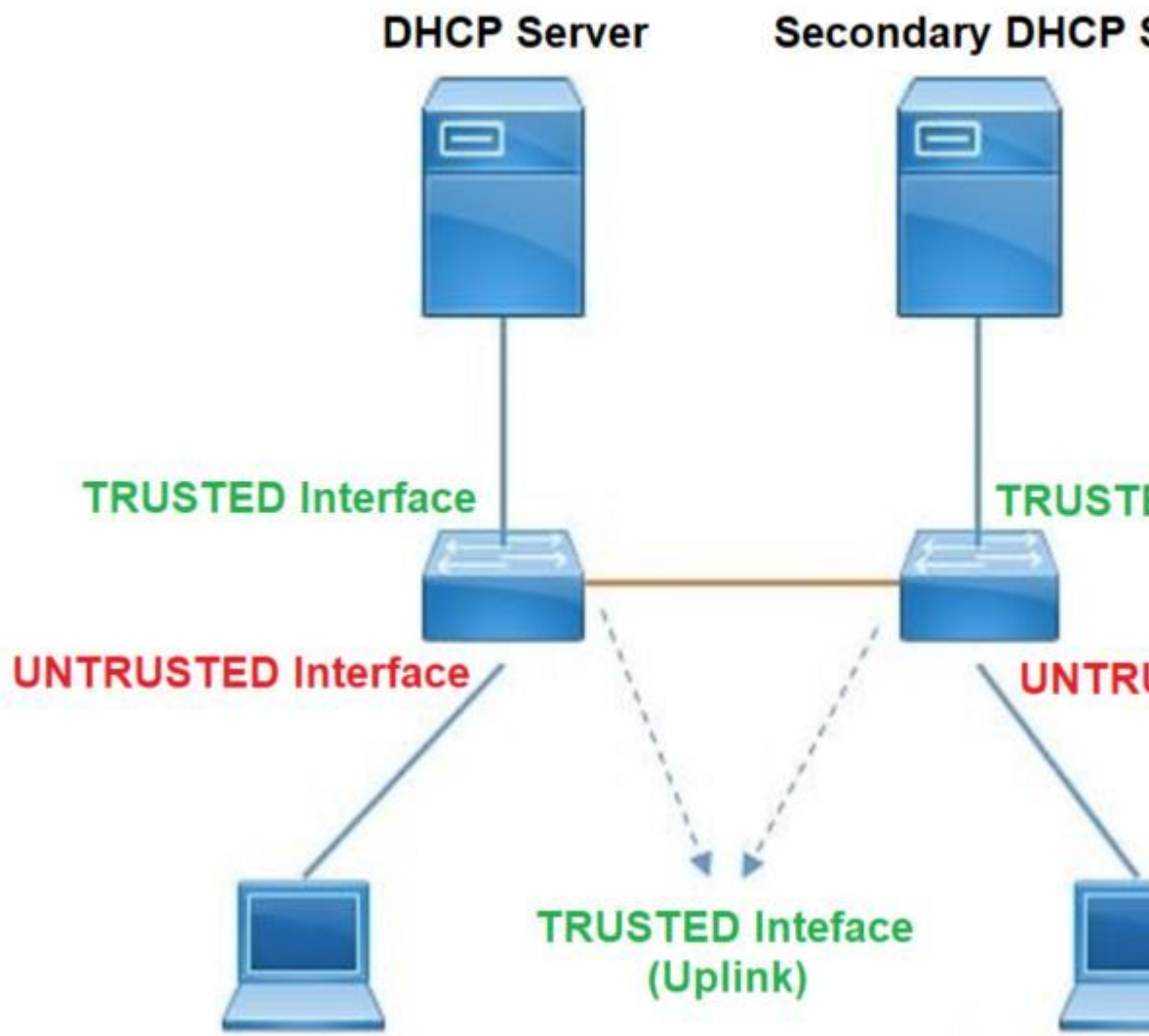
DHCP-Pakete werden an nicht vertrauenswürdigen Schnittstellen blockiert.

- Ein Paket von einem DHCP-Server, z. B. ein DHCPOFFER-, DHCPACK-, DHCPNAK- oder DHCPLEASEQUERY-Paket, wird von außerhalb des Netzwerks oder der Firewall empfangen. Dadurch wird verhindert, dass ein nicht autorisierter DHCP-Server über nicht vertrauenswürdige Ports einen Angriff auf das Netzwerk ausführt.
- Ein an einer nicht vertrauenswürdigen Schnittstelle empfangenes Paket und die Quell-MAC-Adresse sowie die DHCP-Client-Hardwareadresse stimmen nicht überein. Dadurch wird das Spoofing von DHCP-Paketen von einem nicht autorisierten Client verhindert, der einen Denial-of-Service-Angriff auf einen DHCP-Server auslösen könnte.
- Eine DHCPRELEASE- oder DHCPDECLINE-Broadcast-Nachricht mit einer MAC-Adresse in der Datenbank der DHCP-Snooping-Bindung, aber die Schnittstelleninformationen in der Bindungsdatenbank stimmen nicht mit der Schnittstelle überein, an der die Nachricht empfangen wurde. Auf diese Weise werden Denial-of-Service-Angriffe auf Clients verhindert.
- Ein von einem DHCP-Relay-Agent weitergeleitetes DHCP-Paket, das eine Relay-Agent-IP-Adresse enthält, die nicht 0.0.0.0 ist, oder der Relay-Agent leitet ein Paket mit Informationen zur Option 82 an einen nicht vertrauenswürdigen Port weiter. Dadurch wird das Spoofing von Relay-Agent-Informationen im Netzwerk verhindert.

Der Switch, für den Sie DHCP-Snooping konfigurieren, erstellt eine DHCP-Snooping-Tabelle oder eine DHCP-Bindungsdatenbank. Diese Tabelle dient dazu, die von einem legitimen DHCP-Server zugewiesenen IP-Adressen nachzuverfolgen. Die Bindungsdatenbank wird auch von anderen IOS-Sicherheitsfunktionen wie Dynamic ARP Inspection und IP Source Guard verwendet.

Hinweis: Damit DHCP-Snooping richtig funktioniert, müssen Sie allen Uplink-Ports zum Erreichen des DHCP-Servers vertrauen und den Endbenutzer-Ports die Vertrauenswürdigkeit entziehen.

Topologie



Konfigurieren

Globale Konfiguration

<#root>

```
1. Enable DHCP snooping globally on the switch
    switch(config)#
```

ip dhcp snooping

2. Designate ports that forward traffic toward the DHCP server as trusted
    switch(config-if)#

ip dhcp snooping trust

(Additional verification)

- List uplink ports according to the topology, ensure all the uplink ports toward the DHCP server ar

trusted

- List the port where the Legitimate DHCP Server is connected (include any Secondary DHCP Server)
- Ensure that no other port is configured as trusted

3. Configure DHCP rate limiting on each untrusted port (Optional)
    switch(config-if)#

ip dhcp snooping limit rate 10 << ----- 10 packets per second (pps)

4. Enable DHCP snooping in specific VLAN
    switch(config)#

ip dhcp snooping vlan 10

<< ----- Allow the switch to snoop the traffic for that specific VLAN

5. Enable the insertion and removal of option-82 information DHCP packets
    switch(config)#

ip dhcp snooping information option

<-- Enable insertion of option 82

    switch(config)#

no ip dhcp snooping information option

<-- Disable insertion of option 82

Example

Legitimate DHCP Server Interface and Secondary DHCP Server, if available

Server Interface

interface FortyGigabitEthernet1/0/5
switchport mode access

```
switchport mode access vlan 11
ip dhcp snooping trust

end

Uplink interface

interface FortyGigabitEthernet1/0/10
switchport mode trunk
ip dhcp snooping trust

end

User Interface

 << ----- All interfaces are UNTRUSTED by default

interface FortyGigabitEthernet1/0/2
 switchport access vlan 10
 switchport mode access

ip dhcp snooping limit rate 10

<< ----- Optional

end
```

Hinweis: Um Option-82-Pakete zuzulassen, müssen Sie die **Option ip dhcp snooping information allow-untrusted** aktivieren**.**

Überprüfung

Überprüfen Sie, ob DHCP-Snooping für das gewünschte VLAN aktiviert ist, und vergewissern Sie sich, dass vertrauenswürdige und nicht vertrauenswürdige Schnittstellen gut aufgeführt sind. Wenn eine konfigurierte Rate vorliegt, stellen Sie sicher, dass diese ebenfalls aufgeführt ist.

```
<#root>
switch#show ip dhcp snooping

Switch DHCP snooping is
enabled
```

```
Switch DHCP gleaning is disabled
DHCP snooping is configured on following VLANs:
```

10-11

```
DHCP
```

snooping is operational on following VLANs

```
:
```

<<---- Configured and operational on Vlan 10 & 11

10-11

```
DHCP snooping is configured on the following L3 Interfaces:
```

Insertion of option 82 is disabled

<<---- Option 82 can not be added to DHCP packet

```
   circuit-id default format: vlan-mod-port
   remote-id: 00a3.d144.1a80 (MAC)
Option 82 on untrusted port is not allowed
Verification of hwaddr field is enabled
Verification of giaddr field is enabled
DHCP snooping trust/rate is configured on the following Interfaces:

Interface
```

Trusted

```
     Allow option    Rate limit (pps)
-----------------------        -------     ------------    ----------------
FortyGigabitEthernet1/0/2
```

no

```
          no              10
```

<<--- Trust is NOT set on this interface

```
Custom circuit-ids:
FortyGigabitEthernet1/0/10
```

yes

```
         yes             unlimited
```

<<--- Trust is set on this interface

```
Custom circuit-ids:
```

Sobald Benutzer eine IP über DHCP erhalten, werden sie in dieser Ausgabe aufgeführt.

- DHCP-Snooping entfernt den Eintrag in der Datenbank, wenn die IP-Adressen-Lease abläuft oder der Switch eine DHCPRELEASE-Nachricht vom Host empfängt.
- Stellen Sie sicher, dass die für die Endbenutzer-MAC-Adresse aufgeführten Informationen richtig sind.

<#root>

```
c9500#show ip dhcp snooping binding


MacAddress         IpAddress       Lease(sec) Type          VLAN Interface
------------------ --------------- ---------- ------------- ---- --------------------
00:A3:D1:44:20:46  10.0.0.3

85556

 dhcp-snooping 10   FortyGigabitEthernet1/0/2
Total number of bindings: 1
```

In dieser Tabelle sind die verschiedenen Befehle aufgeführt, die zum Überwachen der DHCP-Snooping-Informationen verwendet werden können.

Command	Zweck
show ip dhcp snooping binding **show ip dhcp snooping binding** [IP-Adresse] [MAC-Adresse] [Schnittstelle Ethernet-Steckplatz/Port] [VLAN-ID]	Zeigt nur die dynamisch konfigurierten Bindungen in der Datenbank für die DHCP-Snooping-Bindung an, die auch als Bindungstabelle bezeichnet wird. - Verbindliche IP-Adresse für den Eintrag - Bindungseintrag MAC-Adresse - Bindende Eingabeschnittstelle - Verbindliches Eingangs-VLAN
show ip dhcp snooping-datenbank	Zeigt den Status und die Statistiken der Datenbank für die DHCP-Snooping-Bindung an.
ip dhcp snooping statistik anzeigen	Zeigt die Statistiken für DHCP-Snooping in zusammengefasster oder detaillierter Form an.
show ip source binding	Anzeigen der dynamisch und statisch konfigurierten Bindungen
show interface vlan xyz **show buffer input-interface Vlan xyz dump**	Das DHCP-Paket wird über die Client-VLAN-SVI an den im Client-VLAN konfigurierten Relay-Agent gesendet. Wenn in der Eingabewarteschlange ein Drop- oder Reach-Maximum angezeigt wird, wurde wahrscheinlich das DHCP-Paket vom Client verworfen und konnte den konfigurierten Relay-Agent nicht

	erreichen. **Hinweis**: Stellen Sie sicher, dass in der Eingabewarteschlange keine Auslassungen angezeigt werden. switch#**show int vlan 670** Last für fünf Sekunden: 13 %/0 %; eine Minute: 10 %; fünf Minuten: 10 % Zeitquelle ist NTP, 18:39:52.476 UTC Do 10. September 2020 Vlan670 ist aktiv, Leitungsprotokoll ist aktiv, Autostate aktiviert Hardware ist Ethernet SVI, Adresse ist 00fd.227a.5920 (bia 00fd.227a.5920) Beschreibung: ion_media_client Internet-Adresse: 10.27.49.254/23 MTU 1.500 Byte, BW 1000000 Kbit/s, DLY 10 usc, Zuverlässigkeit 255/255, txload 1/255, rxload 1/255 Kapselungs-ARPA, Loopback nicht festgelegt Keepalive wird nicht unterstützt ARP-Typ: ARPA, ARP-Zeitüberschreitung 04:00:00 Letzte Eingabe 03:01:29, Ausgabe 00:00:02, Ausgabe hängt nie Letzte Löschung der Zähler für "show interface" niemals **Eingangswarteschlange: 375/375/4020251/0** (Größe/max/**drops/**flushes); Gesamtausgangsverluste: 0 **<â€" 375 Pakete am Eingang der Warteschlange / 4020251 wurden verworfen**

Fehlerbehebung

Software-Fehlerbehebung

Überprüfen Sie, was der Switch empfängt. Diese Pakete werden auf der CPU-Steuerungsebene verarbeitet. Stellen Sie daher sicher, dass Sie alle Pakete in Einführ- und Einführ-Richtung sehen, und bestätigen Sie, dass die Informationen richtig sind.

Vorsicht: Verwenden Sie die Debug-Befehle mit Vorsicht. Beachten Sie, dass viele Debug-Befehle Auswirkungen auf das Live-Netzwerk haben und nur in Laborumgebungen verwendet werden sollten, wenn das Problem reproduziert wird.

Mit der Funktion für bedingtes Debuggen können Sie Debug- und Protokolldateien für bestimmte Features auf der Grundlage einer Reihe von Bedingungen, die Sie definieren, selektiv aktivieren. Dies ist nützlich, um Debugging-Informationen nur für bestimmte Hosts oder Datenverkehr zu enthalten.

Eine Bedingung bezieht sich auf eine Funktion oder Identität, bei der die Identität eine Schnittstelle, eine IP-Adresse oder eine MAC-Adresse usw. sein kann.

Aktivieren des bedingten Debuggens für Paket- und Ereignisdebugs zur Fehlerbehebung bei DHCP-Snooping

Command	Zweck
debug condition mac <mac-address> Beispiel: switch#**debug condition mac bc16.6509.3314**	Konfiguriert bedingtes Debuggen für die angegebene MAC-Adresse.
debug condition vlan <VLAN-ID> Beispiel: switch#**debug condition vlan 10**	Konfiguriert bedingtes Debugging für das angegebene VLAN.
debug condition interface <Schnittstelle> Beispiel: switch#**debug, Bedingung interface entyFiveGigE 1/0/8**	Konfiguriert bedingtes Debuggen für die angegebene Schnittstelle.

Verwenden Sie zum Debuggen von DHCP-Snooping die Befehle in der Tabelle.

Command	Zweck
debug dhcp [detail | oper | Redundanz]	**Detail** DHCP-Paketinhalt **oper** DHCP intern OPER **Redundanz** DHCP-Client-Redundanz
debug ip dhcp server packet detail	Detaillierte Dekodierung von Nachrichtenempfängen und -übertragung
debug ip dhcp server events	Zuweisungen von Berichtsadressen, Leasingablauf usw.
debug ip dhcp snooping agent	Debug dhcp Snooping-Datenbank lesen und schreiben
debug ip dhcp snooping event	Debug-Ereignis zwischen den einzelnen Komponenten
debug ip dhcp snooping paket	Debug des DHCP-Pakets im DHCP-Snooping-Modul

Dies ist eine teilweise Beispielausgabe des Befehls **debug ip dhcp snooping**.

<#root>

Apr 14 16:16:46.835: DHCP_SNOOPING: process new DHCP packet,

message type: DHCPDISCOVER, input interface: Fo1/0/2

, MAC da: ffff.ffff.ffff, MAC

sa: 00a3.d144.2046,

 IP da: 255.255.255.255, IP sa: 0.0.0.0, DHCP ciaddr: 0.0.0.0, DHCP yiaddr: 0.0.0.0, DHCP siaddr: 0.0.0.
Apr 14 16:16:46.835: DHCP_SNOOPING: bridge packet get invalid mat entry: FFFF.FFFF.FFFF, packet is flooc

Apr 14 16:16:48.837: DHCP_SNOOPING:

received new DHCP packet from input interface (FortyGigabitEthernet1/0/10)

Apr 14 16:16:48.837: DHCP_SNOOPING:

process new DHCP packet, message type: DHCPOFFER, input interface: Fo1/0/10,

MAC da: ffff.ffff.ffff, MAC

sa: 701f.539a.fe46,

 IP da: 255.255.255.255, IP sa: 10.0.0.1, DHCP ciaddr: 0.0.0.0, DHCP yiaddr: 10.0.0.5, DHCP siaddr: 0.0.
Apr 14 16:16:48.837: platform lookup dest vlan for input_if: FortyGigabitEthernet1/0/10, is NOT tunnel,
Apr 14 16:16:48.837: DHCP_SNOOPING: direct forward dhcp replyto output port: FortyGigabitEthernet1/0/2.
Apr 14 16:16:48.838: DHCP_SNOOPING: received new DHCP packet from input interface (FortyGigabitEthernet1
Apr 14 16:16:48.838: Performing rate limit check

Apr 14 16:16:48.838: DHCP_SNOOPING: process new DHCP packet,

message type: DHCPREQUEST, input interface: Fo1/0/2,

 MAC da: ffff.ffff.ffff, MAC

sa: 00a3.d144.2046,

 IP da: 255.255.255.255, IP sa: 0.0.0.0, DHCP ciaddr: 0.0.0.0, DHCP yiaddr: 0.0.0.0, DHCP siaddr: 0.0.0.
Apr 14 16:16:48.838: DHCP_SNOOPING: bridge packet get invalid mat entry: FFFF.FFFF.FFFF, packet is flooc
Apr 14 16:16:48.839: DHCP_SNOOPING: received new DHCP packet from input interface (FortyGigabitEthernet1

Apr 14 16:16:48.840: DHCP_SNOOPING: process new DHCP packet,

message type: DHCPACK, input interface: Fo1/0/10,

 MAC da: ffff.ffff.ffff, MAC

sa: 701f.539a.fe46,

 IP da: 255.255.255.255, IP

sa: 10.0.0.1,

 DHCP ciaddr: 0.0.0.0, DHCP yiaddr: 10.0.0.5, DHCP siaddr: 0.0.0.0, DHCP giaddr: 0.0.0.0, DHCP chaddr: 0
Apr 14 16:16:48.840: DHCP_SNOOPING: add binding on port FortyGigabitEthernet1/0/2 ckt_id 0 FortyGigabitE
Apr 14 16:16:48.840: DHCP_SNOOPING: added entry to table (index 331)

Apr 14 16:16:48.840:

DHCP_SNOOPING: dump binding entry: Mac=00:A3:D1:44:20:46 Ip=10.0.0.5

 Lease=86400 Type=dhcp-snooping

Vlan=10 If=FortyGigabitEthernet1/0/2

```
Apr 14 16:16:48.840: No entry found for mac(00a3.d144.2046) vlan(10) FortyGigabitEthernet1/0/2
Apr 14 16:16:48.840: host tracking not found for update add dynamic (10.0.0.5, 0.0.0.0, 00a3.d144.2046)
Apr 14 16:16:48.840: platform lookup dest vlan for input_if: FortyGigabitEthernet1/0/10, is NOT tunnel,
Apr 14 16:16:48.840: DHCP_SNOOPING: direct forward dhcp replyto output port: FortyGigabitEthernet1/0/2.
```

Gehen Sie wie folgt vor, um DHCP-Snooping-Ereignisse zu debuggen:

> **Vorsicht**: Verwenden Sie die Debug-Befehle mit Vorsicht. Beachten Sie, dass viele **Debug-Befehle** Auswirkungen auf das Live-Netzwerk haben und nur empfohlen werden, sie in einer Laborumgebung zu verwenden, wenn das Problem reproduziert wird.

Zusammenfassende Schritte

1. aktivieren
2. debug plattform condition mac {mac-address}
3. Debug-Plattformbedingungsstart
4. Plattformbedingung anzeigen ODER Debug anzeigen
5. Debug-Plattformbedingung anhalten
6. show platform software trace message ios R0 reverse | DHCP einschließen
7. frei Plattformbedingung alle

Detaillierte Schritte

	Befehl oder Aktion	**Zweck**
Schritt 1	**aktivieren** Beispiel: switch#**enable**	Aktiviert den privilegierten EXEC-Modus. • Geben Sie auf Aufforderung Ihr Kennwort ein.
Schritt 2	**debug, Plattformbedingung mac** {mac-address} Beispiel: switch#**debug platform condition mac 0001.6509.3314**	Konfiguriert bedingtes Debuggen für die angegebene MAC-Adresse.
Schritt 3	**Debug-Plattformbedingungsstart** Beispiel: switch#**debug plattformbedingung start**	Startet das bedingte Debuggen (dies kann die radioaktive Nachverfolgung starten, wenn eine Übereinstimmung in einer der Bedingungen vorliegt).
Schritt 4	**Plattformbedingung anzeigen** ODER **Debugging anzeigen** Beispiel:	Zeigt die aktuell festgelegten Bedingungen an.

	Befehl oder Aktion	Zweck
	switch#**show plattformbedingung** switch#**show debug**	
Schritt 5	**Debug-Plattformbedingung anhalten** Beispiel: switch#**debug plattformbedingung stop**	Stoppt bedingtes Debuggen (dadurch kann die radioaktive Nachverfolgung gestoppt werden).
Schritt 6	**show platform software trace message ios R0 reverse | DHCP einschließen** Beispiel: switch#**show platform software trace message ios R0 reverse | DHCP einschließen**	Zeigt HP-Protokolle an, die aus der neuesten Ablaufverfolgungsdatei zusammengeführt wurden.
Schritt 7	**frei Plattformbedingung alle** Beispiel: switch# **Plattformbedingung löschen all**	Löscht alle Bedingungen.

Dies ist ein Beispiel für eine teilweise Beispielausgabe des **dBuggy-Plattform dhcp-snoop all** Befehl.

<#root>

debug platform dhcp-snoop all

DHCP Server UDP port

(67)

DHCP Client UDP port

(68)

RELEASE

```
Apr 14 16:44:18.629: pak->vlan_id = 10
Apr 14 16:44:18.629: dhcp packet src_ip(10.0.0.6) dest_ip(10.0.0.1) src_udp(68) dest_udp(67) src_mac(00a
Apr 14 16:44:18.629: ngwc_dhcpsn_process_pak(305): Packet handedover to SISF on vlan 10
Apr 14 16:44:18.629: dhcp pkt processing routine is called for pak with SMAC = 00a3.d144.2046{mac} and S
```

DISCOVER

```
Apr 14 16:44:24.637: dhcp packet src_ip(0.0.0.0) dest_ip(255.255.255.255) src_udp(68) dest_udp(67) src_m
Apr 14 16:44:24.637: ngwc_dhcpsn_process_pak(305): Packet handedover to SISF on vlan 10
Apr 14 16:44:24.637: dhcp pkt processing routine is called for pak with SMAC = 00a3.d144.2046{mac} and S
Apr 14 16:44:24.637: sending dhcp packet out after processing with SMAC = 00a3.d144.2046{mac} and SRC_AD
Apr 14 16:44:24.638: pak->vlan_id = 10
```

OFFER

```
Apr 14 16:44:24.638: dhcp packet src_ip(10.0.0.1) dest_ip(255.255.255.255) src_udp(67) dest_udp(68) src_
Apr 14 16:44:24.638: ngwc_dhcpsn_process_pak(305): Packet handedover to SISF on vlan 10
Apr 14 16:44:24.638: dhcp pkt processing routine is called for pak with SMAC = 701f.539a.fe46{mac}  and
```

REQUEST

```
Apr 14 16:44:24.638: ngwc_dhcpsn_process_pak(284): Packet handedover to SISF on vlan 10
c9500#dhcp pkt processing routine is called for pak with SMAC = 0a3.d144.2046{mac} and SRC_ADDR = 0.0.0.
```

ACK

```
Apr 14 16:44:24.640:  dhcp paket src_ip(10.10.10.1) dest_ip(255.255.255.255) src_udp(67) dest_udp(68) sr
Apr 14 16:44:24.640: ngwc_dhcpsn_process_pak(284): Packet handedover to SISF on vlan 10dhcp pkt processi
```

In dieser Tabelle sind die verschiedenen Befehle aufgeführt, die zum Debuggen von DHCP-Snooping in der Plattform verwendet werden können.

> **Vorsicht**: Verwenden Sie die Debug-Befehle mit Vorsicht. Bitte beachten Sie, dass viele Debug-Befehle Auswirkungen auf das Live-Netzwerk haben und nur empfohlen werden, sie in einer Laborumgebung zu verwenden, wenn das Problem reproduziert wird.

Command	Zweck
switch#**debug platform dhcp-snoop [all | Paket | pd-shim]**	**Alle** NGWC DHCP-Snooping **Paket**-NGWC DHCP-Snooping-Paket Debug-Info **pd-shim** NGWC DHCP Snooping IOS Shim Debug-Info
switch#**debug plattform software infrastruktur punt dhcp-snoop**	Pakete, die auf dem FP empfangen werden und auf die Kontrollebene gelocht werden)
switch#**debug plattform software infrastruktur inject**	Pakete, die von der Kontrollebene in den FP eingespeist werden

Fehlerbehebung: Punt-/Pfad-Datenverkehr (CPU)

Prüfen Sie aus FED-Sicht, welcher Datenverkehr in jeder CPU-Warteschlange empfangen wird (DHCP Snooping ist eine Art von Datenverkehr, der von der Kontrollebene verarbeitet wird).

- Wenn der Datenverkehr in den Switch eingeht, wird er in PUNT-Richtung an die CPU gesendet und an die **DHCP-Snoop**-Warteschlange.
- Sobald der Datenverkehr vom Switch verarbeitet wird, verlässt er die Leitung über die INJECT-Richtung. DHCP OFFER- und ACK-Pakete fallen in die L2-Steuerungs-/Legacy-Warteschlange.

<#root>

```
c9500#show platform software fed switch active punt cause summary

Statistics for all causes

Cause   Cause Info                  Rcvd         Dropped
-----------------------------------------------------------------------------
21      RP<->QFP keepalive          8533         0
79      dhcp snoop                  71           0      <<---- If drop counter increases, there can be a

96      Layer2 control protocols    45662        0
109     snoop packets               100          0
-----------------------------------------------------------------------------

c9500#show platform software fed sw active inject cause summary

Statistics for all causes

Cause Cause Info                  Rcvd           Dropped
-----------------------------------------------------------------------------
1     L2 control/legacy

         128354         0     <<---- dropped counter must NOT increase

2     QFP destination lookup      18             0
5     QFP <->RP keepalive         8585           0
12    ARP request or response     68             0
25    Layer2 frame to BD          81             0
-----------------------------------------------------------------------------
```

Mit diesem Befehl können Sie den an die CPU gesendeten Datenverkehr bestätigen und überprüfen, ob DHCP-Snooping den Datenverkehr verwirft.

<#root>

```
c9500#

show platform software fed switch active punt cpuq rates
```

```
Punt Rate CPU Q Statistics

Packets per second averaged over 10 seconds, 1 min and 5 mins
================================================================================
Q  |              Queue              |  Rx  |  Rx   |  Rx   |  Drop  |  Drop  |  Drop
no |              Name               | 10s  |  1min |  5min |  10s   |  1min  |  5min
================================================================================
0 CPU_Q_DOT1X_AUTH                     0        0      0        0        0        0
1 CPU_Q_L2_CONTROL                     0        0      0        0        0        0
2 CPU_Q_FORUS_TRAFFIC                  0        0      0        0        0        0
3 CPU_Q_ICMP_GEN                       0        0      0        0        0        0
4 CPU_Q_ROUTING_CONTROL                0        0      0        0        0        0
5 CPU_Q_FORUS_ADDR_RESOLUTION          0        0      0        0        0        0
6 CPU_Q_ICMP_REDIRECT                  0        0      0        0        0        0
7 CPU_Q_INTER_FED_TRAFFIC              0        0      0        0        0        0
8 CPU_Q_L2LVX_CONTROL_PKT              0        0      0        0        0        0
9 CPU_Q_EWLC_CONTROL                   0        0      0        0        0        0
10 CPU_Q_EWLC_DATA                     0        0      0        0        0        0
11 CPU_Q_L2LVX_DATA_PKT                0        0      0        0        0        0
12 CPU_Q_BROADCAST                     0        0      0        0        0        0
13 CPU_Q_LEARNING_CACHE_OVFL           0        0      0        0        0        0
14 CPU_Q_SW_FORWARDING                 0        0      0        0        0        0
15 CPU_Q_TOPOLOGY_CONTROL              2        2      2        0        0        0
16 CPU_Q_PROTO_SNOOPING                0        0      0        0        0        0
17 CPU_Q_DHCP_SNOOPING

0        0      0        0        0

        0    <<---- drop counter must NOT increase

18 CPU_Q_TRANSIT_TRAFFIC               0        0      0        0        0        0
19 CPU_Q_RPF_FAILED                    0        0      0        0        0        0
20 CPU_Q_MCAST_END_STATION_SERVICE     0        0      0        0        0        0
21 CPU_Q_LOGGING                       0        0      0        0        0        0
22 CPU_Q_PUNT_WEBAUTH                  0        0      0        0        0        0
23 CPU_Q_HIGH_RATE_APP                 0        0      0        0        0        0
24 CPU_Q_EXCEPTION                     0        0      0        0        0        0
25 CPU_Q_SYSTEM_CRITICAL               8        8      8        0        0        0
26 CPU_Q_NFL_SAMPLED_DATA              0        0      0        0        0        0
27 CPU_Q_LOW_LATENCY                   0        0      0        0        0        0
28 CPU_Q_EGR_EXCEPTION                 0        0      0        0        0        0
29 CPU_Q_FSS                           0        0      0        0        0        0
30 CPU_Q_MCAST_DATA                    0        0      0        0        0        0
31 CPU_Q_GOLD_PKT                      0        0      0        0        0        0

--------------------------------------------------------------------------------
```

Fehlerbehebung bei Hardware

Forwarding Engine-Treiber (FED)

FED ist der Treiber, der den ASIC programmiert. FED-Befehle werden verwendet, um die Übereinstimmung von Hardware- und Softwarestatus zu überprüfen.

Abrufen des Werts DI_Handle

- Das DI-Handle bezieht sich auf den Zielindex für einen bestimmten Port.

<#root>

c9500#show platform software fed switch active security-fed dhcp-snoop vlan vlan-id 10

Platform Security DHCP Snooping Vlan Information

Value of Snooping DI handle

is::

0x7F7FAC23E438 <<---- If DHCP Snooping is not enabled the hardware handle can not be present

Port Trust Mode

FortyGigabitEthernet1/0/10

trust <<---- Ensure TRUSTED ports are listed

Überprüfen Sie die ifm-Zuordnung, um die ASIC und den Core der Ports zu ermitteln.

- IFM ist ein interner Schnittstellenindex, der einem bestimmten Port/Core/Basic zugeordnet ist.

<#root>

c9500#show platform software fed switch active ifm mappings

Interface IF_ID Inst Asic Core Port SubPort Mac Cntx LPN GPN Type Active
FortyGigabitEthernet1/0/10

0xa

3

1 1

1 0 4 4 2 2 NIF Y

Verwenden Sie das DI_Handle, um den Hardware-Index abzurufen.

<#root>

c9500#show platform hardware fed switch active fwd-asic abstraction print-resource-handle 0x7F7FAC23E438

0
Handle:0x7f7fac23e438 Res-Type:ASIC_RSC_DI Res-Switch-Num:255 Asic-Num:255 Feature-ID:AL_FID_DHCPSNOOPIN
priv_ri/priv_si Handle: (nil)Hardware Indices/Handles:

index0:0x5f03

mtu_index/l3u_ri_index0:0x0 index1:0x5f03 mtu_index/l3u_ri_index1:0x0 index2:0x5f03 mtu_index/l3u_ri_in
<SNIP>

<-- Index is 0x5f03

Konvertieren Sie den Indexwert 0x5f03 aus dem Hexadezimalformat in das Dezimalformat.

0x5f03 = 24323

Verwenden Sie diesen Indexwert im Dezimalformat und die ASIC- und Core-Werte in diesem Befehl, um die für den Port festgelegten Flags anzuzeigen.

<#root>

c9500#show platform hardware fed switch 1 fwd-asic regi read register-name SifDestinationIndexTable-2432

asic

1

core

1

For asic 1 core 1

Module 0 - SifDestinationIndexTable[0][

24323

]

<-- the decimal hardware index matches 0x5f03 = 24323

copySegment0 :

0x1 <<---- If you find this as 0x0, means that the traffic is not forwarded out of this port. (refer to

CSCvi39202)copySegment1 : 0x1
dpuSegment0 : 0x0
dpuSegment1 : 0x0
ecUnicast : 0x0
etherChannel0 : 0x0
etherChannel1 : 0x0
hashPtr1 : 0x0
stripSegment : 0x0

Stellen Sie sicher, dass DHCP-Snooping für das jeweilige VLAN aktiviert ist.

<#root>

c9500#show platform software fed switch 1 vlan 10

VLAN Fed Information

Vlan Id IF Id LE Handle STP Handle L3 IF Handle SVI IF

```
--------------------------------------------------------------------------------------------------------------
10      0x0000000000420011

0x00007f7fac235fa8

 0x00007f7fac236798  0x0000000000000000  0x0000000000000000  15



c9500#

show platform hardware fed switch active fwd-asic abstraction print-resource-handle


0x00007f7fac235fa8 1  <<---- Last number might be 1 or 0, 1 means detailed, 0 means brief output


Handle:0x7f7fac235fa8 Res-Type:ASIC_RSC_VLAN_LE Res-Switch-Num:255 Asic-Num:255 Feature-ID:AL_FID_L2 Lkp
priv_ri/priv_si Handle: (nil)Hardware Indices/Handles: index0:0xf mtu_index/l3u_ri_index0:0x0 sm handle
Cookie length: 56
00 00 00 00 00 00 00 00 0a 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00


Detailed Resource Information (ASIC_INSTANCE# 0)
----------------------------------------

LEAD_VLAN_IGMP_MLD_SNOOPING_ENABLED_IPV4 value 1 Pass   <<---- Verify the highlighted values, if any are


LEAD_VLAN_IGMP_MLD_SNOOPING_ENABLED_IPV6 value 0 Pass

LEAD_VLAN_ARP_OR_ND_SNOOPING_ENABLED_IPV4 value 1 Pass


LEAD_VLAN_ARP_OR_ND_SNOOPING_ENABLED_IPV6 value 1 Pass
LEAD_VLAN_BLOCK_L2_LEARN value 0 Pass
LEAD_VLAN_CONTENT_MATCHING_ENABLED value 0 Pass
LEAD_VLAN_DEST_MOD_INDEX_TVLAN_LE value 0 Pass

LEAD_VLAN_DHCP_SNOOPING_ENABLED_IPV4 value 1 Pass


LEAD_VLAN_DHCP_SNOOPING_ENABLED_IPV6 value 1 Pass
LEAD_VLAN_ENABLE_SECURE_VLAN_LEARNING_IPV4 value 0 Pass
LEAD_VLAN_ENABLE_SECURE_VLAN_LEARNING_IPV6 value 0 Pass
LEAD_VLAN_EPOCH value 0 Pass
LEAD_VLAN_L2_PROCESSING_STP_TCN value 0 Pass
LEAD_VLAN_L2FORWARD_IPV4_MULTICAST_PKT value 0 Pass
LEAD_VLAN_L2FORWARD_IPV6_MULTICAST_PKT value 0 Pass
LEAD_VLAN_L3_IF_LE_INDEX_PRIO value 0 Pass
LEAD_VLAN_L3IF_LE_INDEX value 0 Pass
LEAD_VLAN_LOOKUP_VLAN value 15 Pass
LEAD_VLAN_MCAST_LOOKUP_VLAN value 15 Pass
LEAD_VLAN_RIET_OFFSET value 4095 Pass
LEAD_VLAN_SNOOPING_FLOODING_ENABLED_IGMP_OR_MLD_IPV4 value 1 Pass
LEAD_VLAN_SNOOPING_FLOODING_ENABLED_IGMP_OR_MLD_IPV6 value 1 Pass
LEAD_VLAN_SNOOPING_PROCESSING_STP_TCN_IGMP_OR_MLD_IPV4 value 0 Pass
LEAD_VLAN_SNOOPING_PROCESSING_STP_TCN_IGMP_OR_MLD_IPV6 value 0 Pass
LEAD_VLAN_VLAN_CLIENT_LABEL value 0 Pass
LEAD_VLAN_VLAN_CONFIG value 0 Pass
LEAD_VLAN_VLAN_FLOOD_ENABLED value 0 Pass
LEAD_VLAN_VLAN_ID_VALID value 1 Pass
LEAD_VLAN_VLAN_LOAD_BALANCE_GROUP value 15 Pass
LEAD_VLAN_VLAN_ROLE value 2 Pass
```

```
LEAD_VLAN_VLAN_FLOOD_MODE_BITS value 3 Pass
LEAD_VLAN_LVX_VLAN value 0 Pass
LEAD_VLAN_EGRESS_DEJAVU_CANON value 0 Pass
LEAD_VLAN_EGRESS_INGRESS_VLAN_MODE value 0 Pass
LEAD_VLAN_EGRESS_LOOKUP_VLAN value 0 Pass
LEAD_VLAN_EGRESS_LVX_VLAN value 0 Pass
LEAD_VLAN_EGRESS_SGACL_DISABLED value 3 Pass
LEAD_VLAN_EGRESS_VLAN_CLIENT_LABEL value 0 Pass
LEAD_VLAN_EGRESS_VLAN_ID_VALID value 1 Pass
LEAD_VLAN_EGRESS_VLAN_LOAD_BALANCE_GROUP value 15 Pass
LEAD_VLAN_EGRESS_INTRA_POD_BCAST value 0 Pass
```

`LEAD_VLAN_EGRESS_DHCP_SNOOPING_ENABLED_IPV4 value 1 Pass`

```
LEAD_VLAN_EGRESS_DHCP_SNOOPING_ENABLED_IPV6 value 1 Pass
LEAD_VLAN_EGRESS_VXLAN_FLOOD_MODE value 0 Pass
LEAD_VLAN_MAX value 0 Pass
<SNIP>
```

In dieser Tabelle sind die verschiedenen gebräuchlichen Show/Debug-Befehle von Punject aufgeführt, mit denen der Pfad des DHCP-Pakets in einem aktiven Netzwerk verfolgt werden kann.

Allgemeine Punt-/Inject-Befehle zum Anzeigen und Debuggen
debug plat soft fed swit acti inject add-filter Cause 255 sub_Cause 0 src_mac 0 0 0 dst_mac 0 0 0 src_ipv4 192.168.12.1 dst_ipv4 0.0.0.0 if_id 0xf
set platform software trace fed [switch<num|active|standby>] inject verbose **â€" > use filter cmand shows to scope the traces to this specific host**
set platform software trace fed [switch<num|active|standby>] inject debug boot **â€" > for reload**
set platform software trace fed [switch<num|active|standby>] punt noise
show platform software fed [switch<num|active|standby>] injection Cause summary
show platform software fed [switch<num|active|standby>] punt-ursache zusammenfassung
show platform software fed [switch<num|active|standby>] inject cpuq 0
show platform software fed [switch<num|active|standby>] punt cpuq 17 (dhcp queue)
show platform software fed [switch<num|active|standby>] active inject packet capture det
show plattform software infrastruktur injektion
show plattform software infrastruktur punt
show plattform software infrastruktur lsmpi treiber
debug plattform software infra punt dhcp
debug plattform software infra inject

Diese Befehle sind nützlich, um zu überprüfen, ob ein DHCP-Paket für einen bestimmten Client empfangen wurde.

- Mit dieser Funktion können Sie die gesamte DHCP-Snooping-Kommunikation erfassen, die mit einer bestimmten Client-MAC-Adresse verknüpft ist und von der CPU über die IOS-DHCP-Software verarbeitet wird.
- Diese Funktion wird sowohl für IPv4- als auch für IPv6-Datenverkehr unterstützt.
- Diese Funktion wird automatisch aktiviert.

Wichtig: Diese Befehle sind in Cisco IOS XE Gibraltar 16.12.X verfügbar.

---

switch#**show platform dhcpsnooping client stats** {mac-address}
switch#**show platform dhcpv6snooping ipv6 client stats** {mac-address}

<#root>

```
C9300#
```

```
show platform dhcpsnooping client stats 0000.1AC2.C148
```

```
DHCPSN: DHCP snooping server
DHCPD:  DHCP protocol daemen
L2FWD:  Transmit Packet to driver in L2 format
FWD:    Transmit Packet to driver
Packet Trace for client MAC 0000.1AC2.C148:
Timestamp            Destination MAC  Destination Ip  VLAN  Message      Handler:Action
-------------------- ---------------- --------------- ----  ------------ --------------
06-27-2019 20:48:28  FFFF.FFFF.FFFF   255.255.255.255 88   DHCPDISCOVER PUNT:RECEIVED
06-27-2019 20:48:28  FFFF.FFFF.FFFF   255.255.255.255 88   DHCPDISCOVER PUNT:TO_DHCPSN
06-27-2019 20:48:28  FFFF.FFFF.FFFF   255.255.255.255 88   DHCPDISCOVER BRIDGE:RECEIVED
06-27-2019 20:48:28  FFFF.FFFF.FFFF   255.255.255.255 88   DHCPDISCOVER BRIDGE:TO_DHCPD
06-27-2019 20:48:28  FFFF.FFFF.FFFF   255.255.255.255 88   DHCPDISCOVER BRIDGE:TO_INJECT
06-27-2019 20:48:28  FFFF.FFFF.FFFF   255.255.255.255 88   DHCPDISCOVER L2INJECT:TO_FWD
06-27-2019 20:48:28  0000.0000.0000   192.168.1.1     0    DHCPDISCOVER INJECT:RECEIVED
06-27-2019 20:48:28  0000.0000.0000   192.168.1.1     0    DHCPDISCOVER INJECT:TO_L2FWD
06-27-2019 20:48:30  0000.0000.0000   10.1.1.3        0    DHCPOFFER    INJECT:RECEIVED
06-27-2019 20:48:30  0000.1AC2.C148   10.1.1.3        0    DHCPOFFER    INTERCEPT:RECEIVED
06-27-2019 20:48:30  0000.1AC2.C148   10.1.1.3        88   DHCPOFFER    INTERCEPT:TO_DHCPSN
06-27-2019 20:48:30  0000.1AC2.C148   10.1.1.3        88   DHCPOFFER    INJECT:CONSUMED
06-27-2019 20:48:30  FFFF.FFFF.FFFF   255.255.255.255 88   DHCPREQUEST  PUNT:RECEIVED
06-27-2019 20:48:30  FFFF.FFFF.FFFF   255.255.255.255 88   DHCPREQUEST  PUNT:TO_DHCPSN
06-27-2019 20:48:30  FFFF.FFFF.FFFF   255.255.255.255 88   DHCPREQUEST  BRIDGE:RECEIVED
06-27-2019 20:48:30  FFFF.FFFF.FFFF   255.255.255.255 88   DHCPREQUEST  BRIDGE:TO_DHCPD
06-27-2019 20:48:30  FFFF.FFFF.FFFF   255.255.255.255 88   DHCPREQUEST  BRIDGE:TO_INJECT
06-27-2019 20:48:30  FFFF.FFFF.FFFF   255.255.255.255 88   DHCPREQUEST  L2INJECT:TO_FWD
06-27-2019 20:48:30  0000.0000.0000   192.168.1.1     0    DHCPREQUEST  INJECT:RECEIVED
06-27-2019 20:48:30  0000.0000.0000   192.168.1.1     0    DHCPREQUEST  INJECT:TO_L2FWD
06-27-2019 20:48:30  0000.0000.0000   10.1.1.3        0    DHCPACK      INJECT:RECEIVED
06-27-2019 20:48:30  0000.1AC2.C148   10.1.1.3        0    DHCPACK      INTERCEPT:RECEIVED
06-27-2019 20:48:30  0000.1AC2.C148   10.1.1.3        88   DHCPACK      INTERCEPT:TO_DHCPSN
```

Verwenden Sie diese Befehle, um die Ablaufverfolgung zu löschen.

switch#**clear platform dhcpsnooping pkt-trace ipv4**
switch#**clear platform dhcpsnooping pkt-trace ipv6**

CPU-Pfad-Paketerfassung

Überprüfen Sie, ob DHCP-Snooping-Pakete eintreffen und die Kontrollebene ordnungsgemäß verlassen.

> **Hinweis**: Weitere Hinweise zur Verwendung des Forwarding Engine Driver CPU Capture Tools finden Sie im Abschnitt "Weiterlesen".

<#root>

debug platform software fed

 [switch<num|active|standby>]

punt/inject

 packet-capture start

debug platform software fed

 [switch<num|active|standby>]

punt/inject

 packet-capture stop

show platform software fed

  [switch<num|active|standby>]

punt/inject

 packet-capture brief

PUNT

DISCOVER

------ Punt Packet Number: 16, Timestamp: 2021/04/14 19:10:09.924 ------
interface :

physical: FortyGigabitEthernet1/0/2

[if-id: 0x0000000a], pal: FortyGigabitEthernet1/0/2 [if-id: 0x0000000a]
metadata : cause: 79

[dhcp snoop],

 sub-cause: 11, q-no: 17, linktype: MCP_LINK_TYPE_IP [1]
ether hdr : dest mac: ffff.ffff.ffff,

src mac: 00a3.d144.2046

ether hdr : ethertype: 0x0800 (IPv4)
ipv4 hdr : dest ip: 255.255.255.255, src ip: 0.0.0.0
ipv4 hdr : packet len: 347, ttl: 255, protocol: 17 (UDP)
udp hdr : dest port:

67

, src port:

68

OFFER

------ Punt Packet Number: 23, Timestamp: 2021/04/14 19:10:11.926 ------
interface :

physical: FortyGigabitEthernet1/0/10

[if-id: 0x00000012], pal: FortyGigabitEthernet1/0/10 [if-id: 0x00000012]
metadata : cause: 79

[dhcp snoop]

, sub-cause: 11, q-no: 17, linktype: MCP_LINK_TYPE_IP [1]
ether hdr : dest mac: ffff.ffff.ffff,

src mac: 701f.539a.fe46

ether hdr : vlan: 10, ethertype: 0x8100
ipv4 hdr : dest ip: 255.255.255.255,

src ip: 10.0.0.1

ipv4 hdr : packet len: 330, ttl: 255, protocol: 17 (UDP)
udp hdr : dest port:

68

, src port:

67

REQUEST

------ Punt Packet Number: 24, Timestamp: 2021/04/14 19:10:11.927 ------
interface :

physical: FortyGigabitEthernet1/0/2

[if-id: 0x0000000a], pal: FortyGigabitEthernet1/0/2 [if-id: 0x0000000a]
metadata : cause: 79

[dhcp snoop]

, sub-cause: 11, q-no: 17, linktype: MCP_LINK_TYPE_IP [1]

ether hdr : dest mac: ffff.ffff.ffff,

src mac: 00a3.d144.2046

ether hdr : ethertype: 0x0800 (IPv4)
ipv4 hdr : dest ip: 255.255.255.255, src ip: 0.0.0.0
ipv4 hdr : packet len: 365, ttl: 255, protocol: 17 (UDP)
udp hdr : dest port:

67

, src port:

68

ACK

------ Punt Packet Number: 25, Timestamp: 2021/04/14 19:10:11.929 ------
interface :

physical: FortyGigabitEthernet1/0/10

[if-id: 0x00000012], pal: FortyGigabitEthernet1/0/10 [if-id: 0x00000012]
metadata : cause: 79

[dhcp snoop]

, sub-cause: 11, q-no: 17, linktype: MCP_LINK_TYPE_IP [1]
ether hdr : dest mac: ffff.ffff.ffff,

src mac: 701f.539a.fe46

ether hdr : vlan: 10, ethertype: 0x8100
ipv4 hdr : dest ip: 255.255.255.255,

src ip: 10.0.0.1

ipv4 hdr : packet len: 330, ttl: 255, protocol: 17 (UDP)
udp hdr : dest port:

68

, src port:

67

INJECT

DISCOVER

------ Inject Packet Number: 33, Timestamp: 2021/04/14 19:53:01.273 ------
interface : pal:

FortyGigabitEthernet1/0/2

[if-id: 0x0000000a]
metadata : cause: 25 [Layer2 frame to BD], sub-cause: 1, q-no: 0, linktype: MCP_LINK_TYPE_IP [1]
ether hdr : dest mac: ffff.ffff.ffff,

src mac: 00a3.d144.2046

ether hdr : ethertype: 0x0800 (IPv4)
ipv4 hdr : dest ip: 255.255.255.255, src ip: 0.0.0.0
ipv4 hdr : packet len: 347, ttl: 255, protocol: 17 (UDP)
udp hdr : dest port:

67

, src port:

68

OFFER

------ Inject Packet Number: 51, Timestamp: 2021/04/14 19:53:03.275 ------
interface : pal:

FortyGigabitEthernet1/0/2

[if-id: 0x0000000a]
metadata : cause: 1 [L2 control/legacy], sub-cause: 0, q-no: 0, linktype: MCP_LINK_TYPE_LAYER2 [10]
ether hdr : dest mac: ffff.ffff.ffff,

src mac: 701f.539a.fe46

ether hdr : ethertype: 0x0800 (IPv4)
ipv4 hdr : dest ip: 255.255.255.255,

src ip: 10.0.0.1

ipv4 hdr : packet len: 330, ttl: 255, protocol: 17 (UDP)
udp hdr : dest port:

68,

src port:

67

REQUEST

------ Inject Packet Number: 52, Timestamp: 2021/04/14 19:53:03.276 ------
interface : pal:

FortyGigabitEthernet1/0/2

[if-id: 0x0000000a]
metadata : cause: 25 [Layer2 frame to BD], sub-cause: 1, q-no: 0, linktype: MCP_LINK_TYPE_IP [1]
ether hdr : dest mac: ffff.ffff.ffff,

src mac: 00a3.d144.2046

```
ether hdr : ethertype: 0x0800 (IPv4)
ipv4 hdr : dest ip: 255.255.255.255, src ip: 0.0.0.0
ipv4 hdr : packet len: 365, ttl: 255, protocol: 17 (UDP)
udp hdr : dest port:
```

67

```
, src port:
```

68

ACK

```
------ Inject Packet Number: 53, Timestamp: 2021/04/14 19:53:03.278 ------
interface : pal:
```

FortyGigabitEthernet1/0/2

```
 [if-id: 0x0000000a]
metadata : cause: 1 [L2 control/legacy], sub-cause: 0, q-no: 0, linktype: MCP_LINK_TYPE_LAYER2 [10]
ether hdr : dest mac: ffff.ffff.ffff,
```

src mac: 701f.539a.fe46

```
ether hdr : ethertype: 0x0800 (IPv4)
ipv4 hdr : dest ip: 255.255.255.255,
```

src ip: 10.0.0.1

```
ipv4 hdr : packet len: 330, ttl: 255, protocol: 17 (UDP)
udp hdr : dest port:
```

68

```
, src port:
```

67

Nützliche Spuren

Hierbei handelt es sich um binäre Ablaufverfolgungen, die Ereignisse pro Prozess oder Komponente anzeigen. In diesem Beispiel zeigen die Leiterbahnen Informationen über die dhcpsn-Komponente an.

- Die Ablaufverfolgungen können manuell gedreht werden, d. h. Sie können eine neue Datei erstellen, bevor Sie mit der Fehlerbehebung beginnen, sodass diese sauberere Informationen enthält.

<#root>

9500#

request platform software trace rotate all

9500#

set platform software trace fed [switch

] dhcpsn verbose

c9500#show logging proc fed internal | inc dhcp

<<---- DI_Handle must match with the output which retrieves the DI handle

2021/04/14 19:24:19.159536 {fed_F0-0}{1}: [dhcpsn] [17035]: (info):

VLAN event on vlan 10, enabled 1

2021/04/14 19:24:19.159975 {fed_F0-0}{1}: [dhcpsn] [17035]: (debug): Program trust ports for this vlan
2021/04/14 19:24:19.159978 {fed_F0-0}{1}: [dhcpsn] [17035]: (debug):

GPN (10) if_id (0x0000000000000012) <<---- if_id must match with the TRUSTED port

2021/04/14 19:24:19.160029 {fed_F0-0}{1}: [dhcpsn] [17035]: (debug): trusted_if_q size=1 for vlan=10
2021/04/14 19:24:19.160041 {fed_F0-0}{1}: [dhcpsn] [17035]: (ERR): update ri has failed vlanid[10]
2021/04/14 19:24:19.160042 {fed_F0-0}{1}: [dhcpsn] [17035]: (debug): vlan mode changed to enable
2021/04/14 19:24:27.507358 {fed_F0-0}{1}: [dhcpsn] [23451]: (debug): get di for vlan_id 10
2021/04/14 19:24:27.507365 {fed_F0-0}{1}: [dhcpsn] [23451]: (debug): Allocated rep_ri for vlan_id 10
2021/04/14 19:24:27.507366 {fed_F0-0}{1}: [inject] [23451]: (verbose): Changing di_handle from 0x7f7fac3

0x7f7fac23e438

by dhcp snooping
2021/04/14 19:24:27.507394 {fed_F0-0}{1}: [inject] [23451]: (debug): TX: getting REP RI from dhcpsn fail
2021/04/14 19:24:29.511774 {fed_F0-0}{1}: [dhcpsn] [23451]: (debug): get di for vlan_id 10
2021/04/14 19:24:29.511780 {fed_F0-0}{1}: [dhcpsn] [23451]: (debug): Allocated rep_ri for vlan_id 10
2021/04/14 19:24:29.511780 {fed_F0-0}{1}: [inject] [23451]: (verbose): Changing di_handle from 0x7f7fac3

0x7f7fac23e438

by dhcp snooping
2021/04/14 19:24:29.511802 {fed_F0-0}{1}: [inject] [23451]: (debug): TX: getting REP RI from dhcpsn fail

c9500#set platform software trace fed [switch

] asic_app verbose

c9500#show logging proc fed internal | inc dhcp

2021/04/14 20:13:56.742637 {fed_F0-0}{1}: [dhcpsn] [17035]: (info):

VLAN event on vlan 10

, enabled 0
2021/04/14 20:13:56.742783 {fed_F0-0}{1}: [dhcpsn] [17035]: (debug): vlan mode changed to disable
2021/04/14 20:14:13.948214 {fed_F0-0}{1}: [dhcpsn] [17035]: (info): VLAN event on vlan 10, enabled 1
2021/04/14 20:14:13.948686 {fed_F0-0}{1}: [dhcpsn] [17035]: (debug):

Program trust ports for this vlan

2021/04/14 20:14:13.948688 {fed_F0-0}{1}: [dhcpsn] [17035]: (debug):

GPN (10) if_id (0x0000000000000012) <<---- if_id must match with the TRUSTED port

2021/04/14 20:14:13.948740 {fed_F0-0}{1}: [dhcpsn] [17035]: (debug): trusted_if_q size=1 for vlan=10
2021/04/14 20:14:13.948753 {fed_F0-0}{1}: [dhcpsn] [17035]: (ERR): update ri has failed vlanid[10]
2021/04/14 20:14:13.948754 {fed_F0-0}{1}: [dhcpsn] [17035]: (debug): vlan mode changed to enable

Suggested Traces

set platform software trace fed [switch<num|active|standby>] pm_tdl verbose
set platform software trace fed [switch<num|active|standby>] pm_vec verbose
set platform software trace fed [switch<num|active|standby>] pm_vlan verbose

INJECT

set platform software trace fed [switch<num|active|standby>] dhcpsn verbose
set platform software trace fed [switch<num|active|standby>] asic_app verbose
set platform software trace fed [switch<num|active|standby>] inject verbose

PUNT

set platform software trace fed [switch<num|active|standby>] dhcpsn verbose
set platform software trace fed [switch<num|active|standby>] asic_app verbse
set platform software trace fed [switch<num|active|standby>] punt ver

Syslogs und Erklärungen

Verstöße gegen die DHCP-Ratenbeschränkungen.

Erklärung: DHCP-Snooping hat eine Verletzung der DHCP-Paketratengrenze an der angegebenen Schnittstelle erkannt.

```
%DHCP_SNOOPING-4-DHCP_SNOOPING_ERRDISABLE_WARNING: DHCP Snooping received 300 DHCP packets on interface
%DHCP_SNOOPING-4-DHCP_SNOOPING_RATE_LIMIT_EXCEEDED: The interface Fa0/2 is receiving more than the thres
```

DHCP-Server-Spoofing an einem nicht vertrauenswürdigen Port.

Erklärung: Die DHCP-Snooping-Funktion hat bestimmte Arten von DHCP-Nachrichten erkannt, die auf der nicht vertrauenswürdigen Schnittstelle nicht zulässig sind. Dies weist darauf hin, dass ein Host versucht, als DHCP-Server zu fungieren.

```
%DHCP_SNOOPING-5-DHCP_SNOOPING_UNTRUSTED_PORT: DHCP_SNOOPING drop message on untrusted port, message typ
```

Die MAC-Adresse von Layer 2 stimmt nicht mit der MAC-Adresse in der DHCP-Anfrage überein.

Erklärung: Die DHCP-Snooping-Funktion hat eine Überprüfung der MAC-Adresse versucht, und die Prüfung ist fehlgeschlagen. Die Quell-MAC-Adresse im Ethernet-Header stimmt nicht mit der Adresse im Chadr-Feld der DHCP-Anforderungsnachricht überein. Es kann einen schädlichen Host geben, der versucht, einen Denial-of-Service-Angriff auf den DHCP-Server durchzuführen.

```
%DHCP_SNOOPING-5-DHCP_SNOOPING_MATCH_MAC_FAIL: DHCP_SNOOPING drop message because the chaddr doesn't mat
```

Problem mit der Angabe von Option 82.

Erklärung: Die DHCP-Snooping-Funktion hat ein DHCP-Paket entdeckt, dessen Optionswerte auf dem nicht vertrauenswürdigen Port nicht zulässig sind. Dies weist darauf hin, dass ein Host versucht, als DHCP-Relay oder -Server zu fungieren.

```
%DHCP_SNOOPING-5-DHCP_SNOOPING_NONZERO_GIADDR: DHCP_SNOOPING drop message with non-zero giaddr or option
```

Layer-2-MAC-Adresse wurde an falschem Port empfangen.

Erklärung: Die DHCP-Snooping-Funktion hat einen Host erkannt, der versucht, einen Denial-of-Service-Angriff auf einen anderen Host im Netzwerk auszuführen.

```
%DHCP_SNOOPING-5-DHCP_SNOOPING_FAKE_INTERFACE: DHCP_SNNOPING drop message with mismatched source interfa
```

An der nicht vertrauenswürdigen Schnittstelle empfangene DHCP-Nachrichten

Erklärung: Die DHCP-Snooping-Funktion hat bestimmte Arten von DHCP-Nachrichten erkannt, die auf der nicht vertrauenswürdigen Schnittstelle nicht zulässig sind. Dies weist darauf hin, dass ein Host versucht, als DHCP-Server zu fungieren.

```
%DHCP_SNOOPING-5-DHCP_SNOOPING_UNTRUSTED_PORT: DHCP_SNOOPING drop message on untrusted port: GigabitEthe
```

DHCP-Snooping-Übertragung fehlgeschlagen. Kein Zugriff auf URL möglich.

Erklärung: Die Übertragung der DHCP-Snooping-Bindung ist fehlgeschlagen.

```
%DHCP_SNOOPING-4-AGENT_OPERATION_FAILED: DHCP snooping binding transfer failed. Unable to access URL
```

DHCP-Snooping-Hinweise

Cisco Bug-ID	Beschreibung
CSCvi39202	DHCP schlägt fehl, wenn DHCP Snooping Trust auf dem Uplink-EtherChannel aktiviert ist.
CSCvp49518	Die DHCP-Snooping-Datenbank wird nach dem erneuten Laden nicht aktualisiert.
CSCvk16813	DHCP-Client-Datenverkehr wurde durch DHCP-Snooping und Port-Channel oder Stack-übergreifende Uplinks unterbrochen.
CSCvd51480	Aufhebung der Bindung von ip dhcp snooping und Device-Tracking.
CSCvm55401	DHCP Snooping kann die DHCP-Option 82 Pakete mit der IP DHCP Snooping-Informationsoption allow-untrusted verwerfen.
CSCvx25841	Der Vertrauensstatus von DHCP-Snooping wird bei Änderungen im REP-Segment unterbrochen.
CSCvs15759	Der DHCP-Server sendet während des DHCP-Verlängerungsprozesses ein NAK-Paket.
CSCvk34927	DHCP-Snooping-Tabelle wurde beim Neuladen nicht aus der DHCP-Snooping-DB-Datei aktualisiert.

SDA Border DHCP Snooping

CLI für DHCP-Snooping-Statistiken.

Eine neue CLI für SDA zum Überprüfen der DHCP-Snooping-Statistiken.

> **Hinweis**: Weitere Referenzen zu DHCP-Prozess/-Paketfluss und -Decodierung in Cisco SD-Access Fabric Edge finden Sie im Handbuch im Abschnitt "Zugehörige Informationen".

---

switch#**show platform fabric border dhcp snooping ipv4 statistik**
switch#**show platform fabric border dhcp snooping ipv6 statistics**

<#root>

```
SDA-9300-BORDER#
```

```
show platform fabric border dhcp snooping ipv4 statistics
```

```
Timestamp           Source IP    Destination IP  Source Remote Locator  Lisp Instance ID  VLAN  PROCESSE
------------------- ----------   --------------  ---------------------  ----------------  ----- --------
08-05-2019 00:24:16 10.30.30.1   10.40.40.1      192.168.0.1             8189              88    10
08-05-2019 00:24:16 10.30.30.1   10.40.40.1      192.168.0.1             8189              88    11
```

```
SDA-9300-BORDER#
```

```
show platform fabric border dhcp snooping ipv6 statistics
```

```
Timestamp            Source IP              Destination IP          Source Remote Locator  Lisp Instance
-------------------  ---------------------- ----------------------  ---------------------  -------------
08-05-2019 00:41:46  11:11:11:11:11:11:11:1 22:22:22:22:22:22:22:1  192.168.0.3             8089
08-05-2019 00:41:47  11:11:11:11:11:11:11:1 22:22:22:22:22:22:22:1  192.168.0.3             8089
```

Zugehörige Informationen

Konfigurationsanleitung für IP-Adressierungsservices, Cisco IOS XE Amsterdam 17.3.x (Catalyst Switches der Serie 9200)

Konfigurationsanleitung für IP-Adressierungsservices, Cisco IOS XE Amsterdam 17.3.x (Catalyst Switches der Serie 9300)

Konfigurationsanleitung für IP-Adressierungsservices, Cisco IOS XE Amsterdam 17.3.x (Catalyst Switches der Serie 9400)

Konfigurationsanleitung für IP-Adressierungsservices, Cisco IOS XE Amsterdam 17.3.x (Catalyst Switches der Serie 9500)

Konfigurationsanleitung für IP-Adressierungsservices, Cisco IOS XE Amsterdam 17.3.x (Catalyst Switches der Serie 9600)

Cisco SD-Access Fabric Edge DHCP-Prozess/-Paketfluss und -Dekodierung

Konfigurieren der FED-CPU-Paketerfassung auf Catalyst 9000-Switches

Technischer Support und Dokumentation für Cisco Systeme

Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.