Troubleshoot DHCP Issues on Catalyst 9000 DHCP Relay Agents

Contents

Introduction

This document describes how to troubleshoot slow or intermittent DHCP address allocation failures on Catalyst 9000 switches as DHCP relay agents.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Dynamic Host Configuration Protocol (DHCP) and DHCP Relay Agents
- Internet Control Message Protocol (ICMP)
- Control Plane Policing (CoPP)

Components Used

The information in this document is based on these software and hardware versions:

- Catalyst 9000 Series Switches
- Cisco IOS® XE Versions 16.x and 17.x

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.

Related Products

This document can also be used with these hardware and software versions:

- Catalyst 3650/3850 series switches with Cisco IOS XE 16.x

Background Information

This document describes how to troubleshoot slow Dynamic Host Configuration Protocol (DHCP) address allocation or intermittent DHCP address allocation failures on Catalyst 9000 series switches as DHCP relay agents.

The Control Plane Policing (CoPP) feature improves security on your device through protection of the CPU from unnecessary traffic and denial of service (DoS) attacks. It can also protect control traffic and management traffic from traffic drops caused by high volumes of other, lower priority traffic.

Your device is typically segmented into three planes of operation, each with its own objective:

- The data plane, to forward data packets.
- The control plane, to route data correctly.
- The management plane, to manage network elements.

You can use CoPP to protect most of the CPU-bound traffic and ensure routing stability, reachability, and packet delivery. Most importantly, you can use CoPP to protect the CPU from a DoS attack.

CoPP uses the modular QoS command-line interface (MQC) and CPU queues to achieve these objectives. Different types of control plane traffic are grouped together based on certain criteria, and assigned to a CPU queue. You can manage these CPU queues by configuration of dedicated policers in hardware. For example, you can modify the policer rate for certain CPU queues (traffic-type), or you can disable the policer for a certain type of traffic.

Although the policers are configured in hardware, CoPP does not affect CPU performance or the performance of the data plane. But since it limits the number of packets headed up to CPU, the CPU load is controlled. This means that services that wait for packets from hardware can see a more controlled rate of ingress packets (the rate is user-configurable).

Problem

A Catalyst 9000 switch is configured as a DHCP relay agent when the **ip helper-address** command is configured on a routed interface or SVI. The interface where the helper address is configured is typically the default gateway for downstream clients. For the switch to provide successful DHCP relay services to its clients, it must be able to process inbound DHCP Discover messages. This requires the switch to receive the DHCP Discover and punt this packet up to its CPU to process. Once the DHCP Discover is received and processed, the relay agent creates a new unicast packet sourced from the interface where the DHCP Discover was received and destined to the IP Address as defined in the **ip helper-address** configuration. After the packet is created it is hardware forwarded and sent to the DHCP Server where it can be processed and finally sent back to the relay agent so the DHCP process can continue for the client.

A common problem that is experienced is when DHCP transaction packets at the relay agent are inadvertently affected by traffic that is sent to the CPU because it is subject to a specific ICMP scenario, such as an ICMP Redirect or an ICMP Destination Unreachable message. This behavior can manifest itself as clients not able to get an IP address from DHCP timely, or even total DHCP assignment failure. In some scenarios the behavior can only be observed at certain times of the day, such as peak business hours when the load on the network is fully maximized.

As mentioned in the Background section, Catalyst 9000 Series Switches come with a default CoPP policy configured and enabled on the device. This CoPP policy acts as a Quality of Service (QoS) policy that sits in the path of traffic that is received on front panel ports and is destined to the device CPU. It rate limits traffic based on the traffic type and the pre-defined thresholds that are configured in the policy. Some examples of traffic that is classified and rate limited by default are Routing Control packets (typically marked with DSCP CS6), Topology Control packets (STP BPDUs), Low Latency packets (BFD). These packets must be prioritized because the ability to do process them reliably results in a stable network environment.

View the CoPP policer statistics with the **show platform hardware fed switch active qos queue stats internal cpu policer** command.

The ICMP Redirect queue (Queue 6) and the BROADCAST queue (Queue 12) both share the same PlcIdx of 0 (Policer Index). This means that any broadcast traffic that needs to be processed by the device CPU, such as a DHCP Discover, is shared with traffic that is also destined to the device CPU in the ICMP Redirect queue. This can result in the problem mentioned previously where DHCP transactions fail because the ICMP Redirect queue traffic starves out traffic that is need to be serviced by the BROADCAST queue, which results in legitimate broadcast packets dropped.

<#root>

```
9300-Switch#

show platform hardware fed switch active qos queue stats internal cpu policer



                         CPU Queue Statistics
============================================================================================
                                              (default) (set)     Queue        Queue
QId PlcIdx  Queue Name                Enabled   Rate     Rate      Drop(Bytes)  Drop(Frames)
--------------------------------------------------------------------------------------------
0    11     DOT1X Auth                  Yes     1000      1000      0            0
1    1      L2 Control                  Yes     2000      2000      0            0
2    14     Forus traffic               Yes     4000      4000      0            0
3    0      ICMP GEN                    Yes     600       600       0            0
4    2      Routing Control             Yes     5400      5400      0            0
5    14     Forus Address resolution    Yes     4000      4000      0            0

6    0      ICMP Redirect               Yes     600       600       0            0  <-- Policer Index 0


7    16     Inter FED Traffic           Yes     2000      2000      0            0
8    4      L2 LVX Cont Pack            Yes     1000      1000      0            0
9    19     EWLC Control                Yes     13000     13000     0            0
10   16     EWLC Data                   Yes     2000      2000      0            0
11   13     L2 LVX Data Pack            Yes     1000      1000      0            0

12   0      BROADCAST                   Yes     600       600       0            0  <-- Policer Index 0


13   10     Openflow                    Yes     200       200       0            0
14   13     Sw forwarding               Yes     1000      1000      0            0
15   8      Topology Control            Yes     13000     16000     0            0
16   12     Proto Snooping              Yes     2000      2000      0            0
17   6      DHCP Snooping               Yes     500       500       0            0
18   13     Transit Traffic             Yes     1000      1000      0            0
19   10     RPF Failed                  Yes     250       250       0            0
20   15     MCAST END STATION           Yes     2000      2000      0            0
```

```
<snip>
```

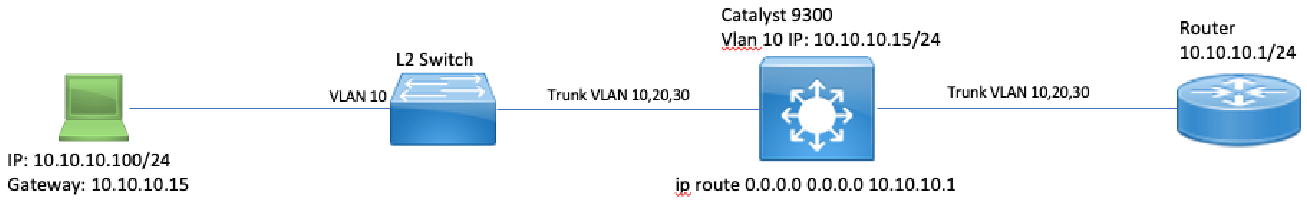
Traffic that exceeds the default 600 packet per second rate in the CoPP policy is dropped before it reaches the CPU.

<#root>

```
9300-Switch#

show platform hardware fed switch active qos queue stats internal cpu policer



                         CPU Queue Statistics
============================================================================================
                                              (default) (set)     Queue        Queue
QId PlcIdx  Queue Name                Enabled   Rate      Rate      Drop(Bytes)  Drop(Frames)
--------------------------------------------------------------------------------------------
0    11     DOT1X Auth                  Yes     1000      1000      0            0
1    1      L2 Control                  Yes     2000      2000      0            0
2    14     Forus traffic               Yes     4000      4000      0            0
3    0      ICMP GEN                    Yes     600       600       0            0
4    2      Routing Control             Yes     5400      5400      0            0
5    14     Forus Address resolution    Yes     4000      4000      0            0

6    0      ICMP Redirect               Yes     600       600       3063106173577 3925209161    <-- Dropp


7    16     Inter FED Traffic           Yes     2000      2000      0            0
8    4      L2 LVX Cont Pack            Yes     1000      1000      0            0
9    19     EWLC Control                Yes     13000     13000     0            0
10   16     EWLC Data                   Yes     2000      2000      0            0
11   13     L2 LVX Data Pack            Yes     1000      1000      0            0

12   0      BROADCAST                   Yes     600       600       1082560387 3133323          <-- Dropp


13   10     Openflow                    Yes     200       200       0            0
14   13     Sw forwarding               Yes     1000      1000      0            0
15   8      Topology Control            Yes     13000     16000     0            0
16   12     Proto Snooping              Yes     2000      2000      0            0
17   6      DHCP Snooping               Yes     500       500       0            0
18   13     Transit Traffic             Yes     1000      1000      0            0
19   10     RPF Failed                  Yes     250       250       0            0
20   15     MCAST END STATION           Yes     2000      2000      0            0
<snip>
```

Scenario 1: ICMP Redirects

Consider this topology for the first scenario:



The sequence of events are as follows:

1. A user at 10.10.10.100 initiates a telnet connection to device 10.100.100.100, a remote network.

2. The destination IP is in a different subnet so the packet is sent to the users default gateway, 10.10.10.15.

3. When the Catalyst 9300 receives this packet to route, it punts the packet to its CPU to generate an ICMP Redirect.

The ICMP Redirect is generated because from the perspective of the 9300 switch, it would be more efficient for the laptop to simply send this packet to the Router at 10.10.10.1 directly, since that is Catalyst 9300's next hop anyways, and it is in the same VLAN that the user is in.

The problem is that the entire flow is processed at the CPU since it meets the ICMP Redirect criteria. If other devices are send traffic that meets the ICMP redirect scenario even more traffic begins to get punted to the CPU in this queue which could impact the BROADCAST queue since they share the same CoPP policer.

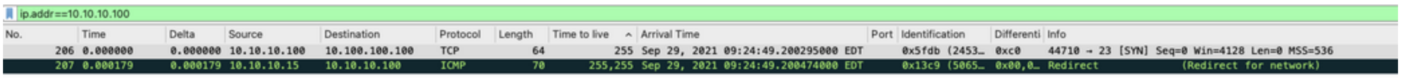
Debug ICMP to view the ICMP Redirect syslog.

<#root>

```
9300-Switch#

debug ip icmp       <-- enables ICMP debugs


ICMP packet debugging is on
9300-Switch#

show logging | inc ICMP


*Sep 29 12:41:33.217: ICMP: echo reply sent, src 10.10.10.15, dst 10.10.10.100, topology BASE, dscp 0 t
*Sep 29 12:41:33.218: ICMP: echo reply sent, src 10.10.10.15, dst 10.10.10.100, topology BASE, dscp 0 t
*Sep 29 12:41:33.219: ICMP: echo reply sent, src 10.10.10.15, dst 10.10.10.100, topology BASE, dscp 0 t
*Sep 29 12:41:33.219: ICMP: echo reply sent, src 10.10.10.15, dst 10.10.10.100, topology BASE, dscp 0 t
*Sep 29 12:43:08.127: ICMP: redirect sent to 10.10.10.100 for dest 10.100.100.100, use gw 10.10.10.1
*Sep 29 12:50:09.517: ICMP: redirect sent to 10.10.10.100 for dest 10.100.100.100, use gw 10.10.10.1

*Sep 29 12:50:10.017: ICMP: redirect sent to 10.10.10.100 for dest 10.100.100.100, use gw 10.10.10.1


*Sep 29 12:50:14.293: ICMP: redirect sent to 10.10.10.100 for dest 10.100.100.100, use gw 10.10.10.1
*Sep 29 12:50:19.053: ICMP: redirect sent to 10.10.10.100 for dest 10.100.100.100, use gw 10.10.10.1
*Sep 29 12:50:23.797: ICMP: redirect sent to 10.10.10.100 for dest 10.100.100.100, use gw 10.10.10.1
*Sep 29 12:50:28.537: ICMP: redirect sent to 10.10.10.100 for dest 10.100.100.100, use gw 10.10.10.1
*Sep 29 12:50:33.284: ICMP: redirect sent to 10.10.10.100 for dest 10.100.100.100, use gw 10.10.10.1
```

⚠ **Caution**: Due to verbosity at scale it is recommended to disable console logging and terminal monitoring before you enable ICMP debugs.

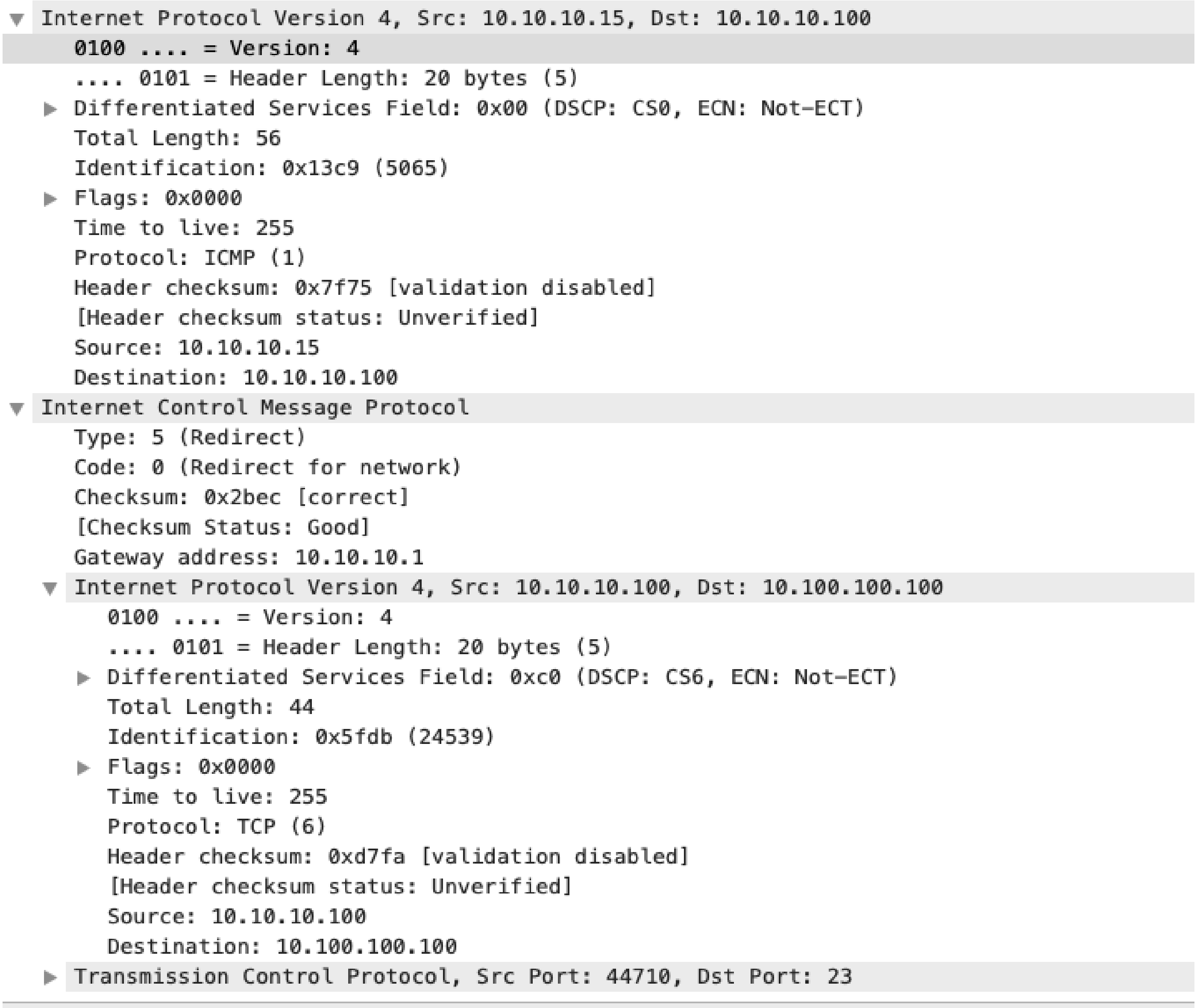
An Embedded Packet Capture at the Catalyst 9300 CPU shows the initial TCP SYN for the Telnet connection at the CPU as well as the ICMP Redirect that is generated.



ip.addr==10.10.10.100

No.	Time	Delta	Source	Destination	Protocol	Length	Time to live	Arrival Time	Port	Identification	Differenti	Info
206	0.000000	0.000000	10.10.10.100	10.100.100.100	TCP	64	255	Sep 29, 2021 09:24:49.200295000 EDT		0x5fdb (2453…	0xc0	44710 → 23 [SYN] Seq=0 Win=4128 Len=0 MSS=536
207	0.000179	0.000179	10.10.10.15	10.10.10.100	ICMP	70	255,255	Sep 29, 2021 09:24:49.200474000 EDT		0x13c9 (5065…	0x00,0…	Redirect (Redirect for network)

The ICMP Redirect packet is sourced from the Catalyst 9300 VLAN 10 interface destined to the client and contains the original packet headers for which the ICMP Redirect packet is sent for.



```
▼ Internet Protocol Version 4, Src: 10.10.10.15, Dst: 10.10.10.100
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▶ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 56
    Identification: 0x13c9 (5065)
  ▶ Flags: 0x0000
    Time to live: 255
    Protocol: ICMP (1)
    Header checksum: 0x7f75 [validation disabled]
    [Header checksum status: Unverified]
    Source: 10.10.10.15
    Destination: 10.10.10.100
▼ Internet Control Message Protocol
    Type: 5 (Redirect)
    Code: 0 (Redirect for network)
    Checksum: 0x2bec [correct]
    [Checksum Status: Good]
    Gateway address: 10.10.10.1
  ▼ Internet Protocol Version 4, Src: 10.10.10.100, Dst: 10.100.100.100
      0100 .... = Version: 4
      .... 0101 = Header Length: 20 bytes (5)
    ▶ Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
      Total Length: 44
      Identification: 0x5fdb (24539)
    ▶ Flags: 0x0000
      Time to live: 255
      Protocol: TCP (6)
      Header checksum: 0xd7fa [validation disabled]
      [Header checksum status: Unverified]
      Source: 10.10.10.100
      Destination: 10.100.100.100
  ▶ Transmission Control Protocol, Src Port: 44710, Dst Port: 23
```

Solution

In this scenario, the packets that are punted up to the CPU can be prevented, which also stops the generation of the ICMP Redirect packet.

Modern operating systems do not employ the use of ICMP Redirect messages so the the resources required to generate and send and process these packets are not an efficient use of CPU resources on network devices.

Alternatively, point the user to use default gateway of 10.10.10.1, but such configuration can be in place for a reason and is outside the scope of this document.

Simply disable ICMP redirects with the **no ip redirects** CLI.

<#root>

```
9300-Switch#

conf t


Enter configuration commands, one per line. End with CNTL/Z.
9300-Switch(config)#

interface vlan 1

0
9300-Switch(config-if)#

no ip redirects       <-- disable IP redirects


9300-Switch(config-if)#end
```

Verify ICMP Redirects are disabled on an interface.

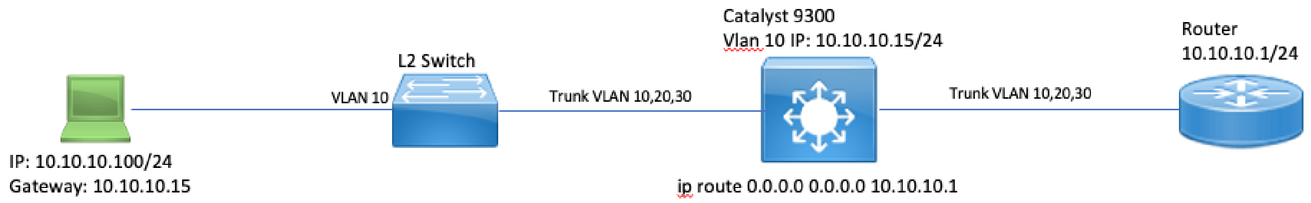
<#root>

```
9300-Switch#

show ip interface vlan 10


 Vlan10 is up, line protocol is up
 Internet address is 10.10.10.15/24
 Broadcast address is 255.255.255.255
 Address determined by setup command
 MTU is 1500 bytes
 Helper address is not set
 Directed broadcast forwarding is disabled
 Multicast reserved groups joined: 224.0.0.102
 Outgoing Common access list is not set
 Outgoing access list is not set
 Inbound Common access list is not set
 Inbound access list is BLOCK-TELNET
 Proxy ARP is disabled
 Local Proxy ARP is disabled
 Security level is default
 Split horizon is enabled


ICMP redirects are never sent       <-- redirects disabled


 ICMP unreachables are never sent
 ICMP mask replies are never sent
 IP fast switching is enabled
 IP Flow switching is disabled
 IP CEF switching is enabled
 IP CEF switching turbo vector
 <snip>
```

More information about ICMP Redirects and when they are sent can be found at this link: https://www.cisco.com/c/en/us/support/docs/ip/routing-information-protocol-rip/13714-43.html

Scenario 2: ICMP Unreachables

Consider the same topology where the user at 10.10.10.100 initiates a Telnet connection to 10.100.100.100. This time an access-list has been configured inbound on the VLAN 10 SVI that blocks telnet connections.



<#root>

```
9300-Switch#

show running-config interface vlan 10

Building Configuration..

Current Configuration : 491 bytes
!
interface Vlan10
 ip address 10.10.10.15 255.255.255.0
 no ip proxy-arp


ip access-group BLOCK-TELNET in             <-- inbound ACL


end



9300-Switch#



9300-Switch#

show ip access-list BLOCK-TELNET


Extended IP access list BLOCK-TELNET


10 deny tcp any any eq telnet            <-- block telnet


    20 permit ip any any
```

```
9300-Switch#
```

The sequence of events are as follows:

1. User at 10.10.10.100 initiates a telnet connection to device 10.100.100.100.

2. The destination IP is in a different subnet so the packet is sent to the users default gateway.

3. When the Catalyst 9300 receives this packet it is evaluated against the inbound ACL and be blocked.

4. Since the packet is blocked and IP unreachables are enabled on the interface, the packet is punted to the CPU so the device can generate an ICMP destination unreachable packet.

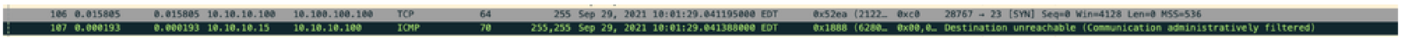
Debug ICMP to view the ICMP destination unreachable syslog.

<#root>

```
9300-Switch#

debug ip icmp              <-- enables ICMP debugs


ICMP packet debugging is on
9300-Switch#

show logging | include ICMP


<snip>

*Sep 29 14:01:29.041: ICMP: dst (10.100.100.100) administratively prohibited unreachable sent to 10.10.1
```

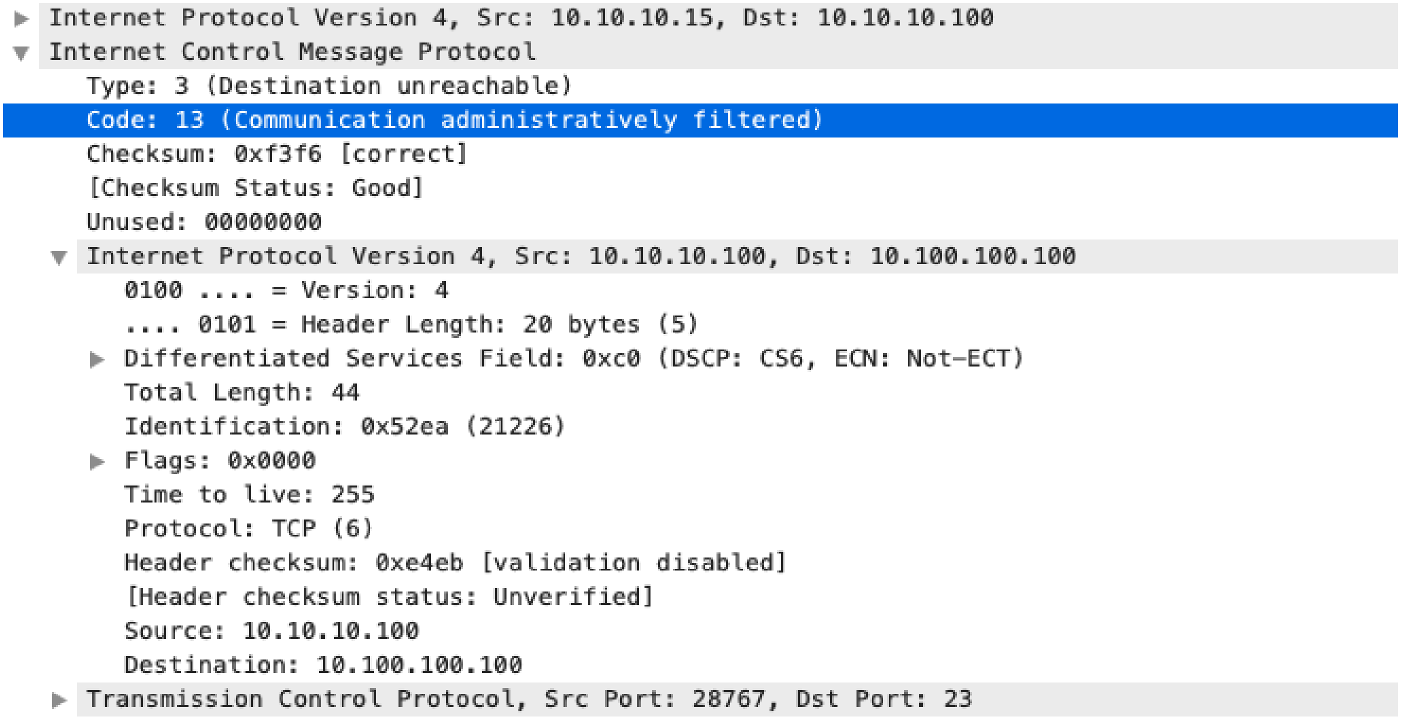
Caution: Due to verbosity at scale it is recommended to disable console logging and terminal monitoring before you enable ICMP debugs.

An Embedded Packet Capture at the Catalyst 9300 CPU shows the initial TCP SYN for the Telnet connection at the CPU as well as the ICMP Destination Unreachable that is sent.



```
106 0.015805     0.015805 10.10.10.100   10.100.100.100   TCP    64      255 Sep 29, 2021 10:01:29.041195000 EDT    0x52ea (2122…  0xc0     28767 → 23 [SYN] Seq=0 Win=4128 Len=0 MSS=536
107 0.000193     0.000193 10.10.10.15    10.10.10.100     ICMP   70  255,255 Sep 29, 2021 10:01:29.041388000 EDT    0x1888 (6280…  0x00,0…  Destination unreachable (Communication administratively filtered)
```

The ICMP Destination Unreachable packet is sourced from the Catalyst 9300 VLAN 10 interface destined to the client and contains the original packet headers for which the ICMP packet is sent for.



```
▶ Internet Protocol Version 4, Src: 10.10.10.15, Dst: 10.10.10.100
▼ Internet Control Message Protocol
    Type: 3 (Destination unreachable)
    Code: 13 (Communication administratively filtered)
    Checksum: 0xf3f6 [correct]
    [Checksum Status: Good]
    Unused: 00000000
  ▼ Internet Protocol Version 4, Src: 10.10.10.100, Dst: 10.100.100.100
      0100 .... = Version: 4
      .... 0101 = Header Length: 20 bytes (5)
    ▶ Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
      Total Length: 44
      Identification: 0x52ea (21226)
    ▶ Flags: 0x0000
      Time to live: 255
      Protocol: TCP (6)
      Header checksum: 0xe4eb [validation disabled]
      [Header checksum status: Unverified]
      Source: 10.10.10.100
      Destination: 10.100.100.100
  ▶ Transmission Control Protocol, Src Port: 28767, Dst Port: 23
```

Solution

In this scenario, disable the behavior where punted packets that are blocked by an ACL in order to generate the ICMP Destination Unreachable message.

IP Unreachable functionality is enabled by default on routed interfaces on Catalyst 9000 series switches.

<#root>

```
9300-Switch#

configure terminal


Enter configuration commands, one per line. End with CNTL/Z.
9300-Switch(config)#

interface vlan 10


9300-Switch(config-if)#

no ip unreachables      <-- disable IP unreachables
```

Verify they are disabled for the interface.

<#root>

```
9300-Switch#

show ip interface vlan 10
```

```
 Vlan10 is up, line protocol is up
 Internet address is 10.10.10.15/24
 Broadcast address is 255.255.255.255
 Address determined by setup command
 MTU is 1500 bytes
 Helper address is not set
 Directed broadcast forwarding is disabled
 Multicast reserved groups joined: 224.0.0.102
 Outgoing Common access list is not set
 Outgoing access list is not set
 Inbound Common access list is not set
 Inbound access list is BLOCK-TELNET
 Proxy ARP is disabled
 Local Proxy ARP is disabled
 Security level is default
 Split horizon is enabled
 ICMP redirects are never sent
```

```
ICMP unreachables are never sent     <-- IP unreachables disabled
```

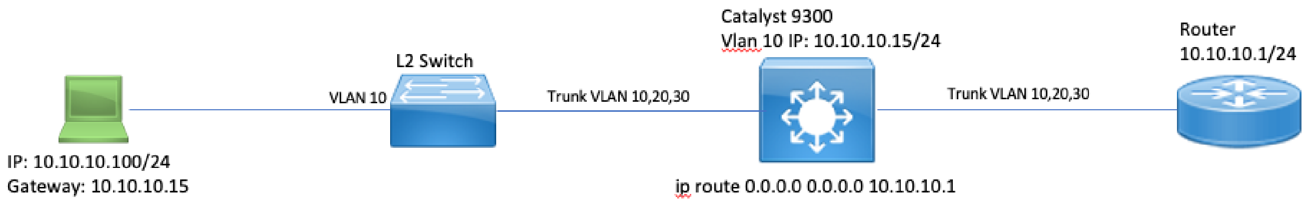
```
 ICMP mask replies are never sent
 IP fast switching is enabled
 IP Flow switching is disabled
 IP CEF switching is enabled
 IP CEF switching turbo vector
 <snip>
```

Scenario 3: ICMP TTL-Exceeded

Consider the prior topology used for the previous 2 scenarios. This time the user at 10.10.10.100 tries to reach a resource in a network that has since decommissioned. Due to this, the SVI and VLAN that used to host this network no longer exists on the Catalyst 9300. However, the Router still has a static route that points to the Catalyst 9300 VLAN 10 interface as the next hop for this network.

Since the Catalyst 9300 no longer has this network configured it does not show as directly connected and the 9300 routes any packets for which it does not have a specific route for to its static default route which points to the Router at 10.10.10.1.

This behavior introduces a routing loop in the network when the user tries to connect to a resource in the 192.168.10.0/24 address space. The packet is looped between the 9300 and the Router until the TTL expires.



1. User tries to connect to a resource in 192.168.10/24 network

2. Packet is received by Catalyst 9300 and is routed to its default route with next hop 10.10.10.1 and decrements the TTL by 1.

3. Router receives this packet and checks the routing table to find there is a route for this network with next hop 10.10.10.15. It decrements the TTL by 1 and routes the packet back to the 9300.

4. Catalyst 9300 receives the packet and once again routes it back to 10.10.10.1 and decrements the TTL by 1.

This process repeats until the IP TTL reaches zero.

When the Catalyst receives the packet with IP TTL = 1 it punts the packet to the CPU and generate an ICMP TTL-Exceeded message.

The ICMP packet type is 11 with Code 0 (TTL expired in transit). This packet type is unable to be disabled via CLI commands

The problem with DHCP traffic comes into play in this scenario because the packets that are looped are subejct to ICMP redirection since they leave out the same interface that they were received on.

The packets sent from the User are also subject to ICMP redirection. DHCP traffic can easily be starved from the BROADCAST queue in this scenario. At scale, this scenario would be even worse due to the number of packets punted in the redirect queue.

Here CoPP drops are demonstrated via 1000 pings to the 192.168.10.0/24 network with timeout of 0 seconds between each ping. The CoPP statistics on the 9300 are cleared and at zero bytes dropped before the pings are sent.

<#root>

```
9300-Switch#

clear platform hardware fed switch active qos statistics internal cpu policer                    <-- cl

9300-Switch#

show platform hardware fed switch active qos queue stats internal cpu policer | i Redirect|Drop   <-- ve

QId PlcIdx  Queue Name                Enabled   Rate     Rate        Drop(Bytes)     Drop(Frames)
6    0      ICMP Redirect               Yes     600       600

0               0                       <-- bytes dropped 0

<snip>
```

User sends traffic to remote network.

<#root>

```
User#

ping 192.168.10.10 timeout 0 rep 1000   <-- User sends 1000 pings

Type escape sequence to abort.
Sending 1000, 100-byte ICMP Echos to 192.168.10.10, timeout is 0 seconds:
......................................................................
......................................................................
```

```
......................................................................
......................................................................
......................................................................
......................................................................
......................................................................
......................................................................
......................................................................
......................................................................
......................................................................
......................................................................
......................................................................
......................................................................
....................
Success rate is 0 percent (0/1000)
```

ICMP Debugs show the Redirect and TTL-Exceeded syslogs due to routing loop.

<#root>

```
9300-Switch#

debug ip icmp


ICMP packet deubgging is on
*Sep 29 16:33:22.676: ICMP:

redirect sent to 10.10.10.100 for dest 192.168.10.10, use gw 10.10.10.1    <-- redirect sent


*Sep 29 16:33:22.678: ICMP:

time exceeded (time to live) sent to 10.10.10.100 (dest was 192.168.10.10), topology BASE, dscp 0 topoi


*Sep 29 16:33:22.678: ICMP: time exceeded (time to live) sent to 10.10.10.100 (dest was 192.168.10.10),
*Sep 29 16:33:22.678: ICMP: time exceeded (time to live) sent to 10.10.10.100 (dest was 192.168.10.10),
<snip>
```

Caution: Due to verbosity at scale it is recommended to disable console logging and terminal monitoring before you enable ICMP debugs.

CoPP drops are seen due to amount of traffic punted to CPU for redirection. Note that this is only for a single client.

<#root>

```
9300-Switch#

show platform hardware fed switch active qos queue stats internal cpu policer


                         CPU Queue Statistics
============================================================================================
                                              (default) (set)     Queue        Queue
QId PlcIdx  Queue Name                Enabled   Rate     Rate      Drop(Bytes)  Drop(Frames)
```

```
--------------------------------------------------------------------------------------------
0    11     DOT1X Auth                Yes      1000       1000      0             0
1    1      L2 Control                Yes      2000       2000      0             0
2    14     Forus traffic             Yes      4000       4000      0             0
3    0      ICMP GEN                  Yes      600        600       0             0
4    2      Routing Control           Yes      5400       5400      0             0
5    14     Forus Address resolution  Yes      4000       4000      0             0

6    0      ICMP Redirect             Yes      600        600       15407990      126295    <-- drops in r


7    16     Inter FED Traffic         Yes      2000       2000      0             0
8    4      L2 LVX Cont Pack          Yes      1000       1000      0             0
<snip>
```

Solution

The solution in this scenario is to disable ICMP Redirects, the same as in Scenario 1. The routing loop is also an issue but the intensity is compounded because the packets are punted for redirection as well.

ICMP TTL-Exceeded packets are also punted when TTL is 1 but these packets use a different CoPP Policer index and do not share a queue with BROADCAST so DHCP traffic is not affected.

Simply disable ICMP redirects with the no ip redirects CLI.

<#root>

```
9300-Switch#

configure terminal


Enter configuration commands, one per line. End with CNTL/Z.
9300-Switch(config)#

interface vlan 10


9300-Switch(config-if)#

no ip redirects        <-- disable IP redirects


9300-Switch(config-if)#

end
```

Related Information

- Configuring Embedded Packet Capture
- Understanding ICMP Redirects
- Technical Support & Documentation - Cisco Systems