Contents

Introduction

This document describes high CPU utilization on various Catalyst platforms due to flooding of IPV6 Multicast Listener Discovery packets and ways to mitigate this problem.

Prerequisites

There are no prerequisites.

Requirements

There are no specific requirements for this document.

Components Used

The information in this document is based on Cisco Catalyst 6500 Series Switches, Catalyst 4500 Series Switches and Catalyst 3850 Series Switches.

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.

Problem

High CPU Utilization may be seen on some Cisco Catalyst platforms due to IPv6 Multicast traffic with MAC address in the range 3333.xxxx.xxxx being punted to CPU.

As per RFC7042, all MAC-48 multicast identifiers prefixed "33-33" (that is, the 2**32 multicast MAC identifiers in the range from 33-33-00-00-00-00 to 33-33-FF-FF-FF-FF) are used as specified

in [RFC2464] for IPv6 multicast. An IPv6 packet with a multicast destination address DST, consisting of the sixteen octets DST[1] through DST[16], is transmitted to the Ethernet multicast address whose first two octets are the value 3333 hexadecimal and whose last four octets are the last four octets of DST as shown in Figure 1.

It has been seen on some occasions that when hosts devices using a certain NIC card go to sleep mode, they flood IPv6 multicast traffic. This issue is not limited to a particular host vendor, though certain chipsets have been seen to exhibit this behavior more often than others.

Troubleshooting & Solution

You can use following procedures to find out if your Catalyst switch seeing high CPU utilization is affected by this problem, and implement respective solutions.

Catalyst 3850 Series Switches

On Catalyst 3850 switches, NGWC L2M Process uses CPU to process IPv6 packets. When Multicast Listener Discovery (MLD) snooping is disabled on the switch, MLD join/leave packet are flooded to all the member ports. And, if there are many incoming MLD join/leave packets, this process will consume more CPU cycles to send out the packets on all the member ports. It has been seen that when certain host machines go to sleep mode, they may send several thousand packets/sec of IGMPv6 MLD traffic.

```
3850#show processes cpu detailed process iosd sorted | exc 0.0
Core 0: CPU utilization for five seconds: 43%; one minute: 35%; five minutes: 33%
Core 1: CPU utilization for five seconds: 54%; one minute: 46%; five minutes: 46%
Core 2: CPU utilization for five seconds: 75%; one minute: 63%; five minutes: 58%
Core 3: CPU utilization for five seconds: 48%; one minute: 49%; five minutes: 57%
PID    T C  TID    Runtime(ms) Invoked uSecs  5Sec      1Min     5Min     TTY   Process
12577  L           2766882     2422952 291    23.52     23.67    23.69    34816 iosd
12577  L 3  12577  1911782     1970561 0      23.34     23.29    23.29    34818 iosd
12577  L 0  14135  694490      3264088 0      0.28      0.34     0.36     0     iosd.fastpath
162    I           2832830     6643    0      93.11     92.55    92.33    0     NGWC L2M
```

Solution

Configure **ipv6 mld snooping** on the affected switches to globally enable **ipv6 mld snooping**. This should lower down the CPU utilization.

```
3850#conf t
Enter configuration commands, one per line. End with CNTL/Z.
3850(config)#ipv6 mld snooping
3850(config)#end
```

When MLD snooping is enabled, a per-VLAN IPv6 multicast address table is constructed in software and hardware. Switch then performs IPv6 multicast-address based bridging in hardware, which prevents these packets to be processed by software.

Click on link for more information on Configuring MLD Snooping

On earlier versions of IOS XE, it was found that CPU queue could get stuck due to this problem which would stop all control packets in that queue from going to the CPU. This was fixed through CSCuo14829 in IOS versions 3.3.3 and 3.6.0 and later. Please refer this bug for details.

Catalyst 4500 Series Switches

Catalyst 4500 series switches support hardware forwarding of IPv6 Multicast traffic using Ternary Content Addressable Memory (TCAM). This is explained in Multicast on Cisco Catalyst 4500E and 4500X Series Switches

When it comes to IPv6 Multicast Listener Discovery traffic, switch needs to perform software forwarding (using CPU resources). As explained in Configuring IPv6 MLD Snooping on Catalyst 4500 Switches MLD snooping can be enabled or disabled globally or per VLAN. When MLD snooping is enabled, a per-VLAN IPv6 multicast MAC address table is constructed in software and a per-VLAN IPv6 multicast address table is constructed in software and hardware. The switch then performs IPv6 multicast-address based bridging in hardware. This is the expected behavior on Catalyst 4500 series switches.

In order to check type of packet being punted to CPU we can run "**debug platform packet all buffer**" followed by "**show platform cpu packet buffered**" command.

```
4500#debug platform packet all buffer
platform packet debugging is on
Cat4500#sh platform cpu packet buffered
Total Received Packets Buffered: 1024
------------------------------------
Index 0:
33 days 11:42:21:833532 - RxVlan: 214, RxPort: Te1/15
Priority: Normal, Tag: Dot1Q Tag, Event: L2 Router, Flags: 0x40, Size: 90
Eth: Src 44:39:C4:39:5A:4A Dst 33:33:FF:7F:EB:DB Type/Len 0x86DD
Remaining data&colon;
0: 0x60 0x0  0x0  0x0  0x0  0x20 0x0  0x1  0xFE 0x80
10: 0x0  0x0  0x0  0x0  0x0  0x0  0x46 0x39 0xC4 0xFF
20: 0xFE 0x39 0x5A 0x4A 0xFF 0x2  0x0  0x0  0x0  0x0
30: 0x0  0x0  0x0  0x0  0x0  0x1  0xFF 0x7F 0xEB 0xDB
40: 0x3A 0x0  0x5  0x2  0x0  0x0  0x1  0x0  0x83 0x0
```

This packet arrived on interface Tengigabitethernet1/15 on vlan 214 from source mac address 44:39:C4:39:5A:4A. Protocol 0x86DD is IPv6 and Dst MAC 33:33:FF:7F:EB:DB is being used for Multicast IPv6 MLD nodes in this case.

Solution

We have two options to fix high CPU utilization due to this traffic.

1. Disable generation of IPv6 Multicast Listener Discovery traffic on end host. This can done by upgrading NIC drivers or disabling the feature on the BIOS of hosts sending IPv6 packets. You can contact your client machine's vendor who can help to disable feature on BIOS or upgrade NIC drivers.

1. Enable Control Plane Policing (CoPP) in order to drop the excessive amount of IPv6 Multicast Listener Discovery traffic which is being punted to the CPU. And, these packets are hop limit of one link local, thus it is expected behavior that these packets will be punted to CPU.

```
4500#debug platform packet all buffer
platform packet debugging is on
Cat4500#sh platform cpu packet buffered
Total Received Packets Buffered: 1024
------------------------------------
```

```
Index 0:
33 days 11:42:21:833532 - RxVlan: 214, RxPort: Te1/15
Priority: Normal, Tag: Dot1Q Tag, Event: L2 Router, Flags: 0x40, Size: 90
Eth: Src 44:39:C4:39:5A:4A Dst 33:33:FF:7F:EB:DB Type/Len 0x86DD
Remaining data&colon;
0: 0x60 0x0  0x0  0x0  0x0  0x20 0x0  0x1  0xFE 0x80
10: 0x0  0x0  0x0  0x0  0x0  0x0  0x46 0x39 0xC4 0xFF
20: 0xFE 0x39 0x5A 0x4A 0xFF 0x2  0x0  0x0  0x0  0x0
30: 0x0  0x0  0x0  0x0  0x0  0x1  0xFF 0x7F 0xEB 0xDB
40: 0x3A 0x0  0x5  0x2  0x0  0x0  0x1  0x0  0x83 0x0
```

In above example, we are limiting the amount of IPv6 traffic which is handled by the CPU to 32000 packets per second.

Catalyst 6500 Series Switches

Catalyst 6500 Switches make forwarding decisions in hardware using TCAM which does not normally need CPU assistance as long as TCAM has the forwarding entry.

Supervisor Enginet 720 on Catalyst 6500 Switches have two CPUs. One CPU is the Network Management Processor (NMP) or the Switch Processor (SP). The other CPU is the Layer 3 CPU, which is called the Route Processor (RP).

Process and Interrupt CPU utilization are listed in **show process cpu** command. As shown below, High

```
4500#debug platform packet all buffer
platform packet debugging is on
Cat4500#sh platform cpu packet buffered
Total Received Packets Buffered: 1024
-------------------------------------
Index 0:
33 days 11:42:21:833532 - RxVlan: 214, RxPort: Te1/15
Priority: Normal, Tag: Dot1Q Tag, Event: L2 Router, Flags: 0x40, Size: 90
Eth: Src 44:39:C4:39:5A:4A Dst 33:33:FF:7F:EB:DB Type/Len 0x86DD
Remaining data&colon;
0: 0x60 0x0  0x0  0x0  0x0  0x20 0x0  0x1  0xFE 0x80
10: 0x0  0x0  0x0  0x0  0x0  0x0  0x46 0x39 0xC4 0xFF
20: 0xFE 0x39 0x5A 0x4A 0xFF 0x2  0x0  0x0  0x0  0x0
30: 0x0  0x0  0x0  0x0  0x0  0x1  0xFF 0x7F 0xEB 0xDB
40: 0x3A 0x0  0x5  0x2  0x0  0x0  0x1  0x0  0x83 0x0
```

Check if any interface or Layer 3 Vlan is dropping high amount of traffic. (Input Queue drops). If so, traffic may be getting punted to RP from that vlan.

```
Vlan19 is up, line protocol is up
 Input queue: 0/75/6303532/0 (size/max/drops/flushes); Total output drops: 0
 Queueing strategy: fifo
 5 minute input rate 19932000 bits/sec, 26424 packets/sec
 5 minute output rate 2662000 bits/sec, 1168 packets/sec
```

Following command can be used to find all packets in input queue buffer for interface vlan 19.

```
Vlan19 is up, line protocol is up
 Input queue: 0/75/6303532/0 (size/max/drops/flushes); Total output drops: 0
 Queueing strategy: fifo
 5 minute input rate 19932000 bits/sec, 26424 packets/sec
 5 minute output rate 2662000 bits/sec, 1168 packets/sec
```

Alternatively, you can use NetDR capture to capture traffic going to CPU on a Catalyst 6500 switch. This document explains how to interpret packets captured using NetDR capture.

```
------- dump of incoming inband packet -------
interface Vl16, routine mistral_process_rx_packet_inlin, timestamp 03:17:56.380
dbus info: src_vlan 0x10(16), src_indx 0x1001(4097), len 0x5A(90)
  bpdu 0, index_dir 0, flood 1, dont_lrn 0, dest_indx 0x4010(16400)
  E8820000 00100000 10010000 5A080000 0C000418 01000008 00000008 4010417E
mistral hdr: req_token 0x0(0), src_index 0x1001(4097), rx_offset 0x76(118)
  requeue 0, obl_pkt 0, vlan 0x10(16)
destmac 33.33.FF.4A.C3.FD, srcmac C8.CB.B8.29.33.62, protocol 86DD
protocol ipv6: version 6, flow 1610612736, payload 32, nexthdr 0, hoplt 1
class 0, src FE80::CACB:B8FF:FE29:3362, dst FF02::1:FF4A:C3FD
```

Solution

Use one or more of below solutions.

1. Drop IPv6 Multicast packets by using following configuration.

```
6500(config)#mac-address-table static 3333.FF4A.C3FD vlan <vlan #> drop
```

1. Redirect IPv6 Multicast traffic to an unused or admin shutdown interface (Gi1/22 in this example).

```
6500(config)#mac-address-table static 3333.FF4A.C3FD vlan <vlan #> drop
```

1. Use Vlan Access Control List (VACL) to drop IPv6 Multicast traffic.

```
6500(config)#mac-address-table static 3333.FF4A.C3FD vlan <vlan #> drop
```

1. Disable IPv6 MLD snooping.

```
6500(config)#mac-address-table static 3333.FF4A.C3FD vlan <vlan #> drop
```

1. Drop IPv6 Multicast traffic using Control Plane Policing (CoPP)

```
6500(config)#mac-address-table static 3333.FF4A.C3FD vlan <vlan #> drop
```

1. Use storm-control on ingress interfaces. storm-control monitors incoming traffic levels over a 1 second interval and during this interval it compares traffic level with configured traffic storm-control level. Traffic storm-control level is a percentage of the total available bandwidth of the port. Each port has a single traffic storm-control level that is used for all types of traffic (broadcast, multicast, and unicast).

```
6500(config)#mac-address-table static 3333.FF4A.C3FD vlan <vlan #> drop
```

7. In case if CPU is High on SP (Switch Processor), apply workaround below.

```
6500(config)#mac-address-table static 3333.FF4A.C3FD vlan <vlan #> drop
```

If you are unable to determine reason based on information provided in this document, please open a TAC service request to investigate further.