Troubleshooting High CPU Utilization Due to the IP Input Process

Document ID: 41160

Contents

Introduction

This document explains how to troubleshoot high CPU utilization due to the IP input process.

Note: This document does not provide strategies to prevent different types of attacks.

Prerequisites

Requirements

Cisco recommends that you read Troubleshooting High CPU Utilization on Cisco Routers before you proceed with this document.

Components Used

This document is not restricted to specific software and hardware versions.

The information presented in this document was created from devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If you are working in a live network, ensure that you understand the potential impact of any command before using it.

Conventions

Refer to Cisco Technical Tips Conventions for more information on document conventions.

IP Input

The Cisco IOS® software process called *IP input* takes care of process−switching IP packets. If the IP input process uses unusually high CPU resources, the router is process−switching a lot of IP traffic. Check these issues:

- **Interrupt switching is disabled on an interface (or interfaces) that has (have) a lot of traffic**

Interrupt switching refers to the use of switching algorithms other than process switching. Examples include fast switching, optimum switching, Cisco Express Forwarding switching, and so on (refer to Performance Tuning Basics for details). Examine the output of the **show interfaces switching** command to see which interface is burdened with traffic. You can check the **show ip interface** command to see which switching method(s) are used on each interface. Re−enable interrupt switching on that interface. Remember that regular fast switching is configured on output interfaces: if fast switching is configured on an interface, packets that go out of that interface are fast−switched. Cisco Express Forwarding switching is configured on input interfaces. To create Forwarding Information Base (FIB) and adjacency table entries on a particular interface, configure Cisco Express Forwarding switching on all interfaces that route to that interface.

- **Fast switching on the same interface is disabled**

  If an interface has a lot of secondary addresses or subinterfaces and there is a lot of traffic sourced from the interface and destined for an address on that same interface, then all of those packets are process−switched. In this situation, you should enable **ip route−cache same−interface** on the interface. When Cisco Express Forwarding switching is used, you do not need to enable Cisco Express Forwarding switching on the same interface separately.
- **Fast switching on an interface providing policy routing is disabled**

  If a route−map has been configured on an interface, and a lot of traffic is handled by the route−map, then the router process−switches this traffic. In this situation, you should enable **ip route−cache policy** on the interface. Check the restrictions mentioned in the "Enabling Fast−Switched Policy−Based Routing" section of Configuring Policy−Based Routing.
- **Traffic that cannot be interrupt−switched arrives**

  This can be any of the listed types of traffic. Click on linked items for more information.

  ♦ Packets for which there is no entry yet in the switching cache

    Even if fast, optimum, or Cisco Express Forwarding switching (CEF) is configured, a packet for which there is no match in the fast−switching cache or FIB and adjacency tables is processed. An entry is then created in the appropriate cache or table, and all subsequent packets that match the same criteria are fast, optimum, or CEF−switched. In normal circumstances, these processed packets do not cause high CPU utilization. However, if there is a device in the network which 1) generates packets at an extremely high rate for devices reachable through the router, and 2) uses different source or destination IP addresses, there is not a match for these packets in the switching cache or table, so they are processed by the IP Input process (if NetFlow switching is configured, source and destination TCP ports are checked against entries in the NetFlow cache as well). This source device can be a non−functional device or, more likely, a device attempting an attack.

    (*) Only with glean adjacencies. Refer to Cisco Express Forwarding for more information about Cisco Express Forwarding adjacencies.

  ♦ Packets destined for the router

    These are examples of packets destined for the router:

    ◊ Routing updates that arrive at an extremely high rate. If the router receives an enormous amount of routing updates that have to be processed, this task might overload the CPU. Normally, this cannot happen in a stable network. The way you can gather more information depends on the routing protocol you have configured. However, you can start to check the output of the **show ip route summary** command periodically. Values that change rapidly are a sign of an unstable network. Frequent routing table changes mean increased routing protocol processing, which results in increased CPU utilization. For further information on how to troubleshoot this issue, refer to the Troubleshooting TCP/IP section of the Internetwork Troubleshooting Guide.

- Any other kind of traffic destined for the router. Check who is logged on to the router and user actions. If someone is logged on and issues commands that produce long output, the high CPU utilization by the "IP input" process is followed by a much higher CPU utilization by the Virtual Exec process.
- Spoof attack. To identify the problem, issue the **show ip traffic** command to check the amount of IP traffic. If there is a problem, the number of received packets with a local destination is significant. Next, examine the output of the **show interfaces** and **show interfaces switching** commands to check which interface the packets are coming in. Once you have identified the receiving interface, turn on **ip accounting** on the outgoing interface and see if there is a pattern. If there is an attack, the source address is almost always different, but the destination address is the same. An access list can be configured to solve the issue temporarily (preferably on the device closest to the source of the packets), but the real solution is to track down the source device and stop the attack.

- Broadcast traffic

  Check the number of broadcast packets in the **show interfaces** command output. If you compare the amount of broadcasts to the total amount of packets that were received on the interface, you can gain an idea of whether there is an overhead of broadcasts. If there is a LAN with several switches connected to the router, then this can indicate a problem with Spanning Tree.
- IP packets with options
- Packets that require protocol translation
- Multilink Point−to−Point Protocol (supported in Cisco Express Forwarding switching)
- Compressed traffic

  If there is no Compression Service Adapter (CSA) in the router, compressed packets must be process−switched.
- Encrypted traffic

  If there is no Encryption Service Adapter (ESA) in the router, encrypted packets must be process−switched.
- Packets that go through serial interfaces with X.25 encapsulation

  In the X.25 protocol suite, flow control is implemented on the second Open System Interconnection (OSI) layer.

- A lot of packets, that arrive at an extremely high rate, for a destination in a directly attached subnet, for which there is no entry in the Address Resolution Protocol (ARP) table. This should not happen with TCP traffic because of the windowing mechanism, but can happen with User Datagram Protocol (UDP) traffic. To identify the problem, repeat the actions suggested in order to track down a spoof attack.
- A lot of multicast traffic goes through the router. Unfortunately, there is no easy way to examine the amount of multicast traffic. The **show ip traffic** command only shows summary information. However, if you have configured multicast routing on the router, you can enable fast−switching of multicast packets with the **ip mroute−cache** interface configuration command (fast−switching of multicast packets is off by default).
- Router is oversubscribed. If the router is over−used and cannot handle this amount of traffic, try to distribute the load among other routers or purchase a high−end router.
- IP Network Address Translation (NAT) is configured on the router, and lots of Domain Name System (DNS) packets go through the router. UDP or TCP packets with source or destination port 53 (DNS) are always punted to process level by NAT.

- There are other packet types that are punted to processing.
- There is fragmentation of IP Datagram. There is a small increase in CPU and memory overhead due to fragment of an IP datagram. Refer to Resolve IP Fragmentation, MTU, MSS, and PMTUD Issues with GRE and IPSEC for more information on how to troubleshoot this issue.

Whatever the reason for high CPU utilization in the IP Input process, the source of the problem can be tracked down if you debug the IP packets. Since the CPU utilization is already high, the debug process has to be performed with extreme caution. The debug process produces lots of messages, so only **logging buffered** should be configured.

Logging to a console raises unnecessary interrupts to the CPU and increases the CPU utilization. Logging to a host (or monitor logging) generates additional traffic on interfaces.

The debug process can be started with the **debug ip packet detail** exec command. This session should not last longer than three to five seconds. Debugging messages are written in the logging buffer. A capture of a sample IP debugging session is provided in the Sample IP Packet Debugging Session section of this document. Once the source device of unwanted IP packets is found, this device can be disconnected from the network, or an access list can be created on the router to drop packets from that destination.

Sample IP Packet Debugging Session

Configured logging destinations should be checked first with the **show logging** command:

```
router#show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Console logging: level debugging, 52 messages logged
    Monitor logging: level debugging, 0 messages logged
    Buffer logging: level debugging, 148 messages logged
    Trap logging: level informational, 64 message lines logged
        Logging to 192.168.100.100, 3 message lines logged
        Logging to 192.168.200.200, 3 message lines logged
 --More--
```

Disable all logging destinations except logging buffer, and clear logging buffer:

```
router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router(config)#no logging console
router(config)#no logging monitor
router(config)#no logging 192.168.100.100
router(config)#no logging 192.168.200.200
router(config)#^Z
router#clear logging
Clear logging buffer [confirm]
router#
```

For better readability of debugging output, datetime and millisecond timestamps should be enabled:

```
router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router(config)#service timestamps log datetime msec
router(config)#service timestamps debug datetime msec
router(config)#end
router#
```

A debugging session can now be started:

```
router#debug ip packet detail
IP packet debugging is on (detailed)
```

Debugging should not last more than three to five seconds. The session can be stopped with the **undebug all** exec command:

```
router#undebug all
All possible debugging has been turned off
```

Results can be checked with the **show logging** exec command:

```
router#show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Console logging: disabled
    Monitor logging: disabled
    Buffer logging: level debugging, 145 messages logged
    Trap logging: level informational, 61 message lines logged
Log Buffer (64000 bytes):

*Mar  3 03:43:27.320: IP: s=192.168.40.53 (Ethernet0/1), d=144.254.2.204
   (Ethernet0/0), g=10.200.40.1, len 100, forward
*Mar  3 03:43:27.324: ICMP type=8, code=0
*Mar  3 03:43:27.324: IP: s=192.168.40.53 (Ethernet0/1), d=144.254.2.205
   (Ethernet0/0), g=10.200.40.1, len 100, forward
*Mar  3 03:43:27.324: ICMP type=8, code=0
*Mar  3 03:43:27.328: IP: s=192.168.40.53 (Ethernet0/1), d=144.254.2.206
   (Ethernet0/0), g=10.200.40.1, len 100, forward
*Mar  3 03:43:27.328: ICMP type=8, code=0
...
```

The log shows that:

- A packet has been received every four milliseconds
- The source IP address is 192.168.40.53
- The packets have come in on interface Ethernet0/1
- The packets have different destination IP addresses
- The packets have been sent out on interface Ethernet0/0
- The next−hop IP address is 10.200.40.1
- The packets were ICMP requests (type=8)

In this example, you can see that the high CPU utilization in the IP Input process has been caused by a ping flood from IP address 192.168.40.53.

SYN floods can easily be detected this way because SYN flag presence is indicated in the debugging output:

```
*Mar 3 03:54:40.436: IP: s=192.168.40.53 (Ethernet0/1), d=144.254.2.204
   (Ethernet0/0), g=10.200.40.1, len 44, forward
*Mar 3 03:54:40.440: TCP src=11004, dst=53,
   seq=280872555, ack=0, win=4128 SYN
```

Related Information

- **Troubleshooting High CPU Utilization on Cisco Routers**
- **The show processes Command**
- **High CPU Utilization on Catalyst 2900XL/3500XL Switches**
- **Performance Tuning Basics**
- **Technical Support & Documentation − Cisco Systems**

Updated: Aug 02, 2006 Document ID: 41160