CHAPTER 11

NetFlow

This chapter describes how to identify and resolve problems that relate to Netflow.

This chapter includes the following sections:

Information About NetFlow

NetFlow is a technology that lets you evaluate IP traffic and understand how and where it flows. NetFlow gathers data that can be used in accounting, network monitoring, and network planning.

A flow is a one-directional stream of packets that arrives on a source interface (or sub-interface), matching a set of criteria. All packets with the same source/destination IP address, source/destination ports, protocol interface and class of service are grouped into a flow and then packets and bytes are tallied. This condenses a large amount of network information into a database called the NetFlow cache.

You create a flow by defining the criteria it gathers. Flow information tells you the following:

- Source address tells you who is originating the traffic.
- Destination address tells who is receiving the traffic.
- Ports characterize the application using the traffic.
- Class of service examines the priority of the traffic.
- The device interface tells how traffic is being used by the network device.
- Tallied packets and bytes show the amount of traffic.

A flow record defines the information that NetFlow gathers, such as packets in the flow and the types of counters gathered per flow. You can define new flow records or use the pre-defined Cisco Nexus 1000V flow record.

For detailed information about configuring NetFlow, see the *Cisco Nexus 1000V System Management Configuration Guide, Release 4.0(4)SV1(1)*.

NetFlow Troubleshooting Commands

Use the commands listed in this section to troubleshoot NetFlow problems.

- **debug logfile** *filename*—Use this command to redirect the output of the following debug commands to a file stored in bootflash.
  - **debug nfm all**
  - **debug sf_nf_srv all**
- **vemdebug netflow dump policy—** Use this command to verify if the correct policy is installed on an interface on a VEM. The output of this command goes to the vemlog internal buffer. Make sure the output shows the cache type as normal, and shows the correct cache size and cache timer values.

```
Apr 14 12:20:51.504410        19   2    2  16   Debug Port 49 has 1
monitors for dir INPUT traffic IPV4
Apr 14 12:20:51.504412        20   2    2  16   Debug Flow Monitor fm1:
Apr 14 12:20:51.504413        21   2    2  16   Debug   Description:
fm1
Apr 14 12:20:51.504413        22   2    2  16   Debug   Monitor ID:
3
Apr 14 12:20:51.504413        23   2    2  16   Debug   Cache:
Apr 14 12:20:51.504414        24   2    2  16   Debug     Type:
normal
Apr 14 12:20:51.504414        25   2    2  16   Debug     Status:
allocated
Apr 14 12:20:51.504415        26   2    2  16   Debug     Size:
256 entries
Apr 14 12:20:51.504415        27   2    2  16   Debug     Inactive
Timeout:  15 secs
Apr 14 12:20:51.504416        28   2    2  16   Debug     Active
Timeout:    1800 secs
```

- **vemdebug netflow dump pakstore—**Use this command to dump pakstore usage for a policy on an interface. The output goes to a vemlog internal buffer. Make sure the output shows the correct monitor name and interface.

```
Apr 14 12:25:30. 29787       260   0    2  16   Debug Pak Store for
Client: fm1
Apr 14 12:25:30. 29793       266   0    2  16   Debug Pak Store for
Client: LTL49
```

- **vemlog debug sfnetflow_cache all**
- **vemlog debug sfnetflow_config all**
- **vemlog debug sfnetflow_flowapi all**

  Use these command to enable NetFlow debugging for policy installation on the VEM. Debug messages are printed for every PDL session open, verify, and commit requests coming from the DPA.
- **vemlog debug sfnetflow all**

  Use this command to enable packet path debugging for Netflow policies on the VEM. Debug messages are printed for every packet that hits a NetFlow policy. Use this command with caution. High traffic could result in lot of debug messages.

Use the following commands to collect information about NFM process run-time configuration errors:

- **show flow internal event-history errors**
- **show flow internal event-history msgs**

- **show flow internal ddb b**
- **show flow internal mem-stats** (to debug memory usage and leaks)

Use the following commands to collect sf_nf_srv process run-time information:

- **show system internal sf_nf_srv event-history errors**
- **show system internal sf_nf_srv event-history msgs**
- **show system internal sf_nf_srv pdl detailed**
- **show system internal sf_nf_srv mem-stats** (to debug memory usage and leaks)

Common NetFlow Problems

Common NetFlow configuration problems on the VSM can occur if you attempt to do the following:

- Use undefined records, exporters, samplers, or monitors
- Use invalid records, exporters, samplers, or monitors
- Modify records, exporters, samplers, or monitors after they are applied to an interface
- Configure a monitor on an interface which causes the VEM to run out of memory and results in a verification error

In addition, a configuration error can occur if there is a mismatch between the UDP port configured on the exporter and the port NetFlow Collector has listening turned on.

Debugging a Policy Verification Error

To debug a policy verification failure due to some processing on the VSM, follow these steps:

Step 1 Issue the **debug nfm all** command.

Step 2 Issue the **debug sf_nf_srv_all** command.

Step 3 Save the Telnet SSH session buffer to a file.

Step 4 Issue the **ip flow mon** *monitor name direction* command.

The command will execute once again and the debug traces will be output to the console.

You can also use the policy verification procedure to collect logs for operations such as defining a flow record or tracing exporter functionality.

Debugging Statistics Export

When debugging a NetFlow statistics export problem, follow these guidelines:

- Ensure the destination IP address is reachable from the VSM.
- Ensure the UDP port configured on the exporter matches that used by the NetFlow Collector.
- Run **tcpdump** on the host running the NetFlow Collector to identify if the data exported from the VSM reaches the host. (Wireshark Ethereal supports NetFlow V5/V9 payload decoding.)

- Issue the **show flow exporter** command to view statistics for the exporter and identify any drops on the VSM.