**Discovery 21: Configure and Verify Flexible NetFlow**

**Introduction**

The objective of this discovery lab is to provide you with some experience with the syntax of a Flexible NetFlow configuration to facilitate the capture of a traffic flow. This lab is prepared with the devices that are represented in the topology diagram and the connectivity table. The devices have their basic configurations in place, including hostnames and IP addresses.

**Note**

The PCs in the lab environment are routers simulating hosts, so you will use Cisco IOS commands to configure them or make verifications.

In the discovery lab, you will:

* Configure a single NetFlow record to export the traffic flow to a single collector.
* Use show commands to verify the NetFlow configuration.

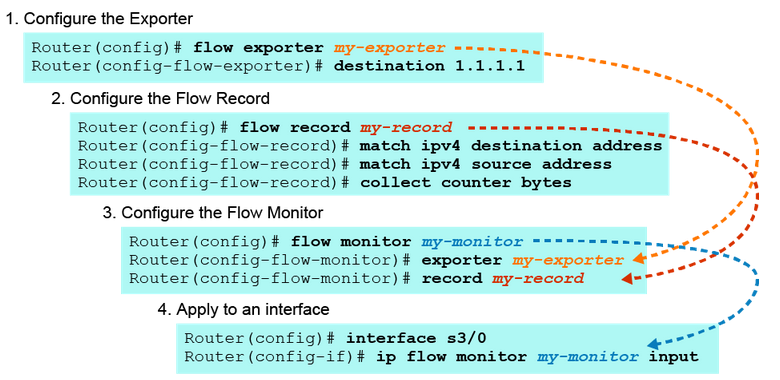
Note

Individual device platforms and Cisco IOS Software revisions may have different configuration steps. You should consult the specific administration guide and release notes for more accurate configuration requirements.

**Task 1: Configure Flexible NetFlow on R1**

**Activity**

A basic Flexible NetFlow configuration consists of four steps shown in this example:



**Step 1:** On R1, configure SRV1 (10.1.1.10) as the NetFlow Collector to which the Netflow traffic will be exported.

The exporter configuration describes where the flows are sent. This terminology is confusing because most NetFlow users (including the Stealthwatch system) refer to an “exporter” as the router itself.

On R1, enter the following commands:

R1(config)# flow exporter EXPORTER1

R1(config-flow-exporter)# description Stealthwatch

R1(config-flow-exporter)# source loopback0

R1(config-flow-exporter)# destination 10.1.1.10

R1(config-flow-exporter)# transport udp 2055

R1(config-flow-exporter)# exit



In this example, the NetFlow information is sent to a Stealthwatch server acting as a NetFlow collector at 10.1.1.10 using UDP port 2055. Each flow exporter supports only one destination. You can create multiple exporters if you have multiple Stealthwatch collectors.

Using the loopback interface as the source ensures that all NetFlow packets source from the same IP address on the router.

**Step 2:** On R1, configure the NetFlow Record.

A NetFlow record is a combination of key and non-key fields used to identify flows. There are both predefined and user-defined records that can be configured. Customized user-defined flow records are used to analyze traffic data for a specific purpose.

A customized flow record must have at least one match criterion for use as the key field and typically has at least one collect criterion for use as a non-key field. You have to specify a series of match and collect commands that tell the router which fields to include in the outgoing NetFlow PDU. The match fields are the key fields: they are used to determine the uniqueness of the flow. The collect fields are just extra info (non-key) that you include to provide more detail to the collector for reporting and analysis.

On R1, enter the following commands:

R1(config)# flow record LANCOPE1

R1(config-flow-record)# match ipv4 tos

R1(config-flow-record)# match ipv4 protocol

R1(config-flow-record)# match ipv4 source address

R1(config-flow-record)# match ipv4 destination address

R1(config-flow-record)# match transport source-port

R1(config-flow-record)# match transport destination-port

R1(config-flow-record)# match interface input

R1(config-flow-record)# collect routing next-hop address ipv4

R1(config-flow-record)# collect ipv4 dscp

R1(config-flow-record)# collect interface output

R1(config-flow-record)# collect counter bytes

R1(config-flow-record)# collect counter packets

R1(config-flow-record)# collect timestamp sys-uptime first

R1(config-flow-record)# collect timestamp sys-uptime last

R1(config-flow-record)# exit



An explanation of the flow record commands used in the example is as follows:

Required Key Fields

match ipv4 tos

match ipv4 protocol

match ipv4 source address

match ipv4 destination address

match transport source-port

match transport destination-port

match interface input

Others

collect routing next-hop address ipv4 --- required

collect ipv4 dscp --- optional; used to generate QoS reports

collect interface output --- required; used for computing bps rates

collect counter bytes --- required; used for bps calculation

collect counter packets --- required; used for pps calculation

collect timestamp sys-uptime first --- required; for calculating duration

collect timestamp sys-uptime last --- required; for calculating duration

You do not want to modify the match fields much. The seven match entries that are shown should always be included in your configuration. The collect fields, however, can vary, depending on how much information you want to send to the collector. The configuration that is listed is recommended for all Stealthwatch installations.

**Step 3:** On R1, configure the NetFlow Monitor.

The monitor represents the memory-resident NetFlow database of the router. Flexible NetFlow allows you to create multiple independent monitors. While it can be useful in some situations, most users create a single main cache for collecting and exporting NetFlow data.

On R1, enter the following commands:

R1(config)# flow monitor MONITOR1

R1(config-flow-monitor)# description Main Cache

R1(config-flow-monitor)# record LANCOPE1

R1(config-flow-monitor)# exporter EXPORTER1

R1(config-flow-monitor)# cache timeout active 60

R1(config-flow-monitor)# cache timeout inactive 60

R1(config-flow-monitor)# exit



Configuring multiple monitors will use a significant amount of memory in the exporter, so be aware. In this example, a single monitor called “MONITOR1” has been configured and is associated with the flow record and flow exporter.

**Step 4:** On R1, apply the flow monitor to each Layer 3-enabled interface.

NetFlow must be enabled at each entry point to the router. In almost all cases, you want to use input monitoring. It ensures that both sides of all communications through the router are captured by Flexible NetFlow and sent to the collector.

On R1, enter the following commands:

R1(config)# interface GigabitEthernet0/0

R1(config-if)# ip flow monitor MONITOR1 input

R1(config-if)# interface GigabitEthernet0/2

R1(config-if)# ip flow monitor MONITOR1 input

R1(config-if)# end

****

**Step 5:** On R1, verify the Flexible NetFlow configuration with the show flow monitor command.

On R1, enter the following commands:

R1# show flow monitor

****

**Step 6:** On PC1, generate traffic for NetFlow to capture:

* Ping PC2, 10.20.20.10.
* Telnet to PC2, 10.20.20.10, using the username of admin and password of cisco123.

On PC1, enter the following commands:

PC1# ping 10.20.20.10



**Step 7:** Return to R1, and display the NetFlow cache.

On R1, enter the following command:

R1# show flow monitor MONITOR1 cache



Although your lab output may slightly differ from the example presented here, notice the major flows captured:

The ping from PC1 to PC2 and its return:

IPV4 SOURCE ADDRESS: 10.10.10.10

IPV4 DESTINATION ADDRESS: 10.20.20.10

TRNS SOURCE PORT: 0

TRNS DESTINATION PORT: 2048

INTERFACE INPUT: G0/0

---------------------------------------

IPV4 SOURCE ADDRESS: 10.20.20.10

IPV4 DESTINATION ADDRESS: 10.10.10.10

TRNS SOURCE PORT: 0

TRNS DESTINATION PORT: 0

INTERFACE INPUT: G0/2

The telnet from PC1 to PC2 and its return:

IPV4 SOURCE ADDRESS: 10.10.10.10

IPV4 DESTINATION ADDRESS: 10.20.20.10

TRNS SOURCE PORT: 45288

TRNS DESTINATION PORT: 23

INTERFACE INPUT: G0/0

----------------------------------------

IPV4 SOURCE ADDRESS: 10.20.20.10

IPV4 DESTINATION ADDRESS: 10.10.10.10

TRNS SOURCE PORT: 23

TRNS DESTINATION PORT: 45288

INTERFACE INPUT: G0/2