

Monitoring Netflow with NfSen

Contents

1	Introduction	1
1.1	Goals	1
1.2	Notes	1
2	Export flows from a Cisco router	2
2.1	Group 1, Router 1	2
2.2	Group 2, Router 2	2
3	Configuring the routers	2

1 Introduction

1.1 Goals

- Learn how to export flows from a Cisco router

1.2 Notes

- Commands preceded with “\$” imply that you should execute the command as a general user - not as root.
- Commands preceded with “#” imply that you should be working as root.
- Commands with more specific command lines (e.g. “rtrX>” or “mysql>”) imply that you are executing commands on remote equipment, or within another program.

2 Export flows from a Cisco router

You will configure your router to export the same flow data to all PCs in your group.

2.1 Group 1, Router 1

```
rtr1 ==> pc1 on port 9001
rtr1 ==> pc2 on port 9001
rtr1 ==> pc3 on port 9001
rtr1 ==> pc4 on port 9001
```

2.2 Group 2, Router 2

```
rtr2 ==> pc5 on port 9001
rtr2 ==> pc6 on port 9001
rtr2 ==> pc7 on port 9001
rtr2 ==> pc8 on port 9001
```

etc.

3 Configuring the routers

```
$ ssh cisco@rtrX.ws.nsrc.org
rtrX> enable
```

or, if ssh is not configured yet:

```
$ telnet 10.10.1.254
Username: cisco
Password:
Router1>enable
Password:
```

The following configures the FastEthernet 0/0 interface to export flows. Replace 10.10.X.A to .D with the IP addresses of the PCs in your group.

```
rtrX# configure terminal
rtrX(config)#

flow exporter EXPORTER-1
```

```

description Export to pcA
destination 10.10.X.A
transport udp 9001
template data timeout 60

flow exporter EXPORTER-2
description Export to pcB
destination 10.10.X.B
transport udp 9001
template data timeout 60

flow exporter EXPORTER-3
description Export to pcC
destination 10.10.X.C
transport udp 9001
template data timeout 60

flow exporter EXPORTER-4
description Export to pcD
destination 10.10.X.D
transport udp 9001
template data timeout 60

flow monitor FLOW-MONITOR-V4
exporter EXPORTER-1
exporter EXPORTER-2
exporter EXPORTER-3
exporter EXPORTER-4
record netflow ipv4 original-input
cache timeout active 300

interface FastEthernet 0/0
ip flow monitor FLOW-MONITOR-V4 input
ip flow monitor FLOW-MONITOR-V4 output

snmp-server ifindex persist

```

Since you have not specified a protocol version for the exported flow records, you get the default which is Netflow v9.

Netflow v9 packets cannot be decoded by the receiver until it has received a template packet. The command `template data timeout 60` tells the router to send it every 60 seconds, to make the lab exercises work more quickly. (In production a value of 300 is fine).

The `cache timeout active 300` command breaks up long-lived flows into 5-minute fragments. If you leave it at the default of 30 minutes your traffic graphs

will have spikes.

Aside: if you want to monitor IPv6 flows you would have to create a new flow monitor for IPv6 and attach it to the interface and the existing exporters.

```
flow monitor FLOW-MONITOR-V6
  exporter EXPORTER-1
  exporter EXPORTER-2
  exporter EXPORTER-3
  exporter EXPORTER-4
  record netflow ipv6 original-input
  cache timeout active 300
interface FastEthernet 0/0
  ipv6 flow monitor FLOW-MONITOR-V6 input
  ipv6 flow monitor FLOW-MONITOR-V6 output
```

The command `snmp-server ifindex persist` enables ifIndex persistence globally. This ensures that the ifIndex values are retained during router reboots - also if you add or remove interface modules to your network devices.

Now we'll verify what we've done.

First exit from the configuration session:

```
rtrX(config)# end

rtrX# show flow exporter EXPORTER-1
rtrX# show flow exporter EXPORTER-2
etc...
rtrX# show flow monitor FLOW-MONITOR-V4
```

It's possible to see the individual flows that are active in the router:

```
rtrX# show flow monitor FLOW-MONITOR-V4 cache
```

But on a busy router there will be thousands of individual flows, so that's not useful. Press 'q' to escape from the screen output if necessary.

Instead, group the flows so you can see your "top talkers" (traffic destinations and sources). This is one very long command line:

```
rtrX# show flow monitor FLOW-MONITOR-V4 cache aggregate ipv4 source address
      ipv4 destination address sort counter bytes top 20
```

If it all looks good then write your running-config to non-volatile RAM (i.e. the startup-config):

```
rtrX#wr mem
```

You can exit from the router now:

```
rtrX#exit
```

Make sure we have the tcpdump tool installed:

```
$ sudo apt-get install tcpdump
```

Now verify that flows are arriving from your router to your PC:

```
$ sudo tcpdump -i eth0 -nn -Tcnfp port 9001
```

Wait a few seconds and you should see something that looks like:

```
06:12:00.953450 IP s2.ws.nsrc.org.54538 > noc.ws.nsrc.org.9009: NetFlow v5, 9222.333 uptime,
  started 8867.952, last 8867.952
    10.10.0.241/0:0:53 > 10.10.0.250/0:0:49005 >> 0.0.0.0
      udp tos 0, 1 (136 octets)
    started 8867.952, last 3211591.733
      10.10.0.241/10:0:0 > 0.0.0.0/10:0:4352 >> 0.0.0.0
        ip tos 0, 62 (8867952 octets)
[...]
```

These are the UDP packets containing individual flow records.

(Note that the actual output may not be correct, as tcpdump does not decode Netflow properly)

OPTIONAL: you can use tshark (the text version of wireshark), which is able to fully decode Netflow v9 packets:

```
$ sudo apt-get install tshark
$ sudo tshark -i eth0 -nnV -s0 -d udp.port==9001,cflow udp port 9001
```

You are done for this lab.

Go to exercise2-install-nfdump-nfsen.