TRAINING SESSION NAME

Firewall

Table of contents

A.1 Nfacct

1. Technical concept

1.1 Concept

nfacct is the command line tool to create/retrieve/delete accounting objects

Main Features

- listing the objects of the nfacct table in plain text/XML
- automatically get and reset objects of the nfacct table
- adding new objects to the nfacct table
- deleting objects from the nfacct table

1.2 libraries

- libnetfilter_acct is the userspace library providing the interface to extended accounting infrastructure.

- libmnl: is a minimalistic user-space library oriented to Netlink developers. There are a lot of common tasks in parsing, validating, and constructing of both the Netlink header and TLVs that are repetitive and easy to get wrong. This library aims to provide simple helpers that allow you to reuse code and avoid re-inventing the wheel.

1.3 Config

1.3.1 nfacct.conf file

```
{ pkts = 00000000000000000000, bytes = 00000000000000000000 } = 0x01;
{ pkts = 00000000000000000000, bytes = 00000000000000000000 } = 0x02;
{ pkts = 00000000000000000000, bytes = 00000000000000000000 } = 0x03;
{ pkts = 00000000000000000000, bytes = 00000000000000000000 } = 0x04;
{ pkts = 00000000000000000000, bytes = 00000000000000000000 } = 0x05;
```

1.3.2 Restore nfacct config

```
/usr/sbin/nfacct restore < nfacct.conf
```

1.3.3 Save to nfacct config

```
/usr/sbin/nfacct list >  nfacct.conf
```

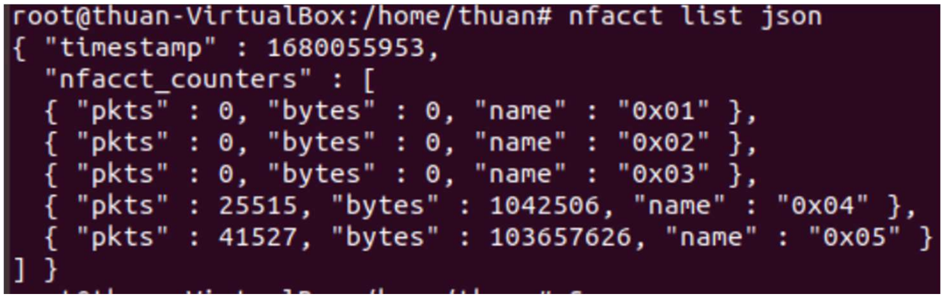
1.3.4 Flush nfacct config

```
/usr/sbin/nfacct flush
```

1.3.5 Show nfacct config

```
/usr/sbin/nfacct list json
```

Output:

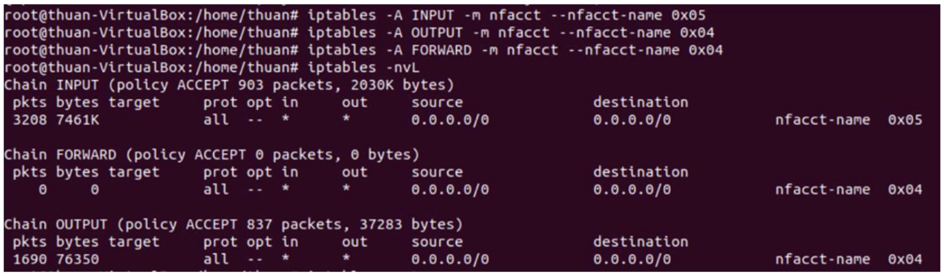


```
root@thuan-VirtualBox:/home/thuan# nfacct list json
{ "timestamp" : 1680055953,
  "nfacct_counters" : [
  { "pkts" : 0, "bytes" : 0, "name" : "0x01" },
  { "pkts" : 0, "bytes" : 0, "name" : "0x02" },
  { "pkts" : 0, "bytes" : 0, "name" : "0x03" },
  { "pkts" : 25515, "bytes" : 1042506, "name" : "0x04" },
  { "pkts" : 41527, "bytes" : 103657626, "name" : "0x05" }
] }
```

1.4 iptables config

```
-A INPUT -m nfacct --nfacct-name 0x05
-A FORWARD -m nfacct --nfacct-name 0x04
-A OUTPUT -m nfacct --nfacct-name 0x04
```

Output:



```
root@thuan-VirtualBox:/home/thuan# iptables -A INPUT -m nfacct --nfacct-name 0x05
root@thuan-VirtualBox:/home/thuan# iptables -A OUTPUT -m nfacct --nfacct-name 0x04
root@thuan-VirtualBox:/home/thuan# iptables -A FORWARD -m nfacct --nfacct-name 0x04
root@thuan-VirtualBox:/home/thuan# iptables -nvL
Chain INPUT (policy ACCEPT 903 packets, 2030K bytes)
 pkts bytes target     prot opt in     out     source               destination
 3208 7461K            all  --  *      *       0.0.0.0/0            0.0.0.0/0            nfacct-name  0x05

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0            all  --  *      *       0.0.0.0/0            0.0.0.0/0            nfacct-name  0x04

Chain OUTPUT (policy ACCEPT 837 packets, 37283 bytes)
 pkts bytes target     prot opt in     out     source               destination
 1690 76350            all  --  *      *       0.0.0.0/0            0.0.0.0/0            nfacct-name  0x04
```

A.2 Nflog

1. Technical concept

1.1 Concept

Package nflog provides an API to interact with the log subsystem of the netfilter family from the linux kernel.

1.2 libraries

`libnetfilter_log` is a userspace library providing interface to packets that have been logged by the kernel packet filter. It is is part of a system that deprecates the old syslog/dmesg based packet logging. This library has been previously known as `libnfnetlink_log`.

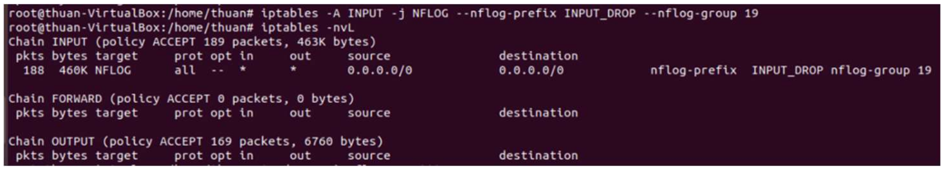
`libnetfilter_log` is used by `ulogd2`.

1.3 Config

1.3.1 iptables config file

```
-A INPUT -j NFLOG --nflog-prefix INPUT_DROP --nflog-group 19
```

Output:

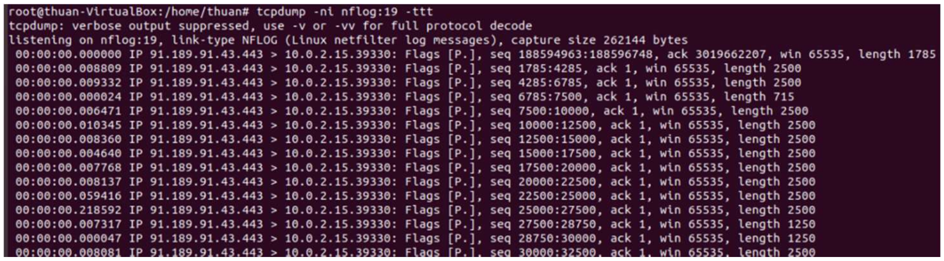


```
root@thuan-VirtualBox:/home/thuan# iptables -A INPUT -j NFLOG --nflog-prefix INPUT_DROP --nflog-group 19
root@thuan-VirtualBox:/home/thuan# iptables -nvL
Chain INPUT (policy ACCEPT 189 packets, 463K bytes)
 pkts bytes target     prot opt in     out     source               destination
  188  460K NFLOG      all  --  *      *       0.0.0.0/0            0.0.0.0/0            nflog-prefix  INPUT_DROP nflog-group 19

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 169 packets, 6760 bytes)
 pkts bytes target     prot opt in     out     source               destination
```

1.3.2 Show log with tcpdump

```
tcpdump -ni nflog:19 -ttt
```

Output:



```
root@thuan-VirtualBox:/home/thuan# tcpdump -ni nflog:19 -ttt
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on nflog:19, link-type NFLOG (Linux netfilter log messages), capture size 262144 bytes
 00:00:00.000000 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 188594963:188596748, ack 3019662207, win 65535, length 1785
 00:00:00.008809 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 1785:4285, ack 1, win 65535, length 2500
 00:00:00.009332 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 4285:6785, ack 1, win 65535, length 2500
 00:00:00.000024 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 6785:7500, ack 1, win 65535, length 715
 00:00:00.006471 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 7500:10000, ack 1, win 65535, length 2500
 00:00:00.010345 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 10000:12500, ack 1, win 65535, length 2500
 00:00:00.008360 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 12500:15000, ack 1, win 65535, length 2500
 00:00:00.004640 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 15000:17500, ack 1, win 65535, length 2500
 00:00:00.007768 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 17500:20000, ack 1, win 65535, length 2500
 00:00:00.008137 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 20000:22500, ack 1, win 65535, length 2500
 00:00:00.059416 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 22500:25000, ack 1, win 65535, length 2500
 00:00:00.218592 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 25000:27500, ack 1, win 65535, length 2500
 00:00:00.007317 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 27500:28750, ack 1, win 65535, length 1250
 00:00:00.000047 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 28750:30000, ack 1, win 65535, length 1250
 00:00:00.008081 IP 91.189.91.43.443 > 10.0.2.15.39330: Flags [P.], seq 30000:32500, ack 1, win 65535, length 2500
```

B. Questions, Exercises

NA

C. References

No.	Info	Link/ file/ name of ebook
1	Nfacct introduction	https://www.netfilter.org/projects/nfacct/index.html
2	Nfacct code review	http://charette.no-ip.com:81/programming/doxygen/netfilter/group__nfacct.html
3	Nfacct clone source code	git clone git://git.netfilter.org/nfacct
4	Sequence diagram	http://charette.no-ip.com:81/programming/doxygen/netfilter/dir_9780868c6b71f5d64c6e4d0f1808a326.html
5	Nfacct vs ulogd2	https://home.regit.org/2012/07/flow-accounting-with-netfilter-and-ulogd2/