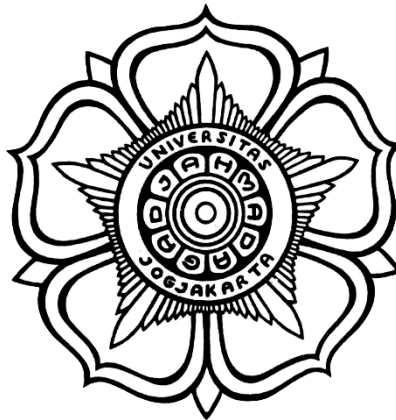


LAPORAN PRAKTIKUM
KEAMANAN INFORMASI 1
PERTEMUAN 6
SNORT DAN FIREWALL RULE



DISUSUN OLEH

Nama : Riva Mahyuli
NIM : (21/478709/SV/19365)
Kelas : R1AA

SARJANA TERAPAN TEKNOLOGI REKAYASA INTERNET
DEPARTEMEN TEKNIK ELEKTRO DAN INFORMATIKA

SEKOLAH VOKASI

2023/2024

Snort dan Firewall Rules

Alat dan Bahan

- Mesin virtual CyberOps Workstation
- Koneksi Internet

Langkah Kerja

1. Bagian 1: Mempersiapkan Lingkungan Virtual

Buka VM CyberOps Workstation, buka terminal dan konfigurasi jaringannya dengan menjalankan skrip `configure_as_dhcp.sh`. Karena skrip memerlukan hak pengguna super, berikan kata sandi untuk user `analyst`

```
[analyst@secOps ~]$ sudo ./lab.support.files/scripts/configure_as_dhcp.sh
[sudo] password for analyst:
Configuring the NIC to request IP info via DHCP...
Requesting IP information...
IP Configuration successful.
```

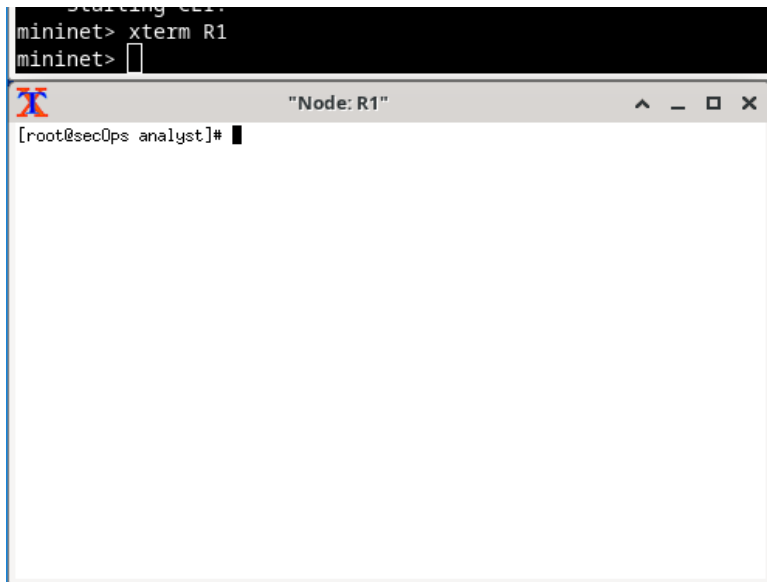
2. Bagian 2: Firewall and IDS Logs

Dari terminal VM CyberOps Workstation, jalankan skrip untuk memulai `mininet.mininet`.

```
[analyst@secOps ~]$ sudo ./lab.support.files/scripts/cyberops_extended_topo_no_fw.py
[sudo] password for analyst:
*** Adding controller
*** Add switches
*** Add hosts
*** Add links
*** Starting network
*** Configuring hosts
R1 R4 H1 H2 H3 H4 H5 H6 H7 H8 H9 H10 H11
*** Starting controllers
*** Starting switches
*** Add routes
*** Post configure switches and hosts
*** Starting CLI:
mininet>
```

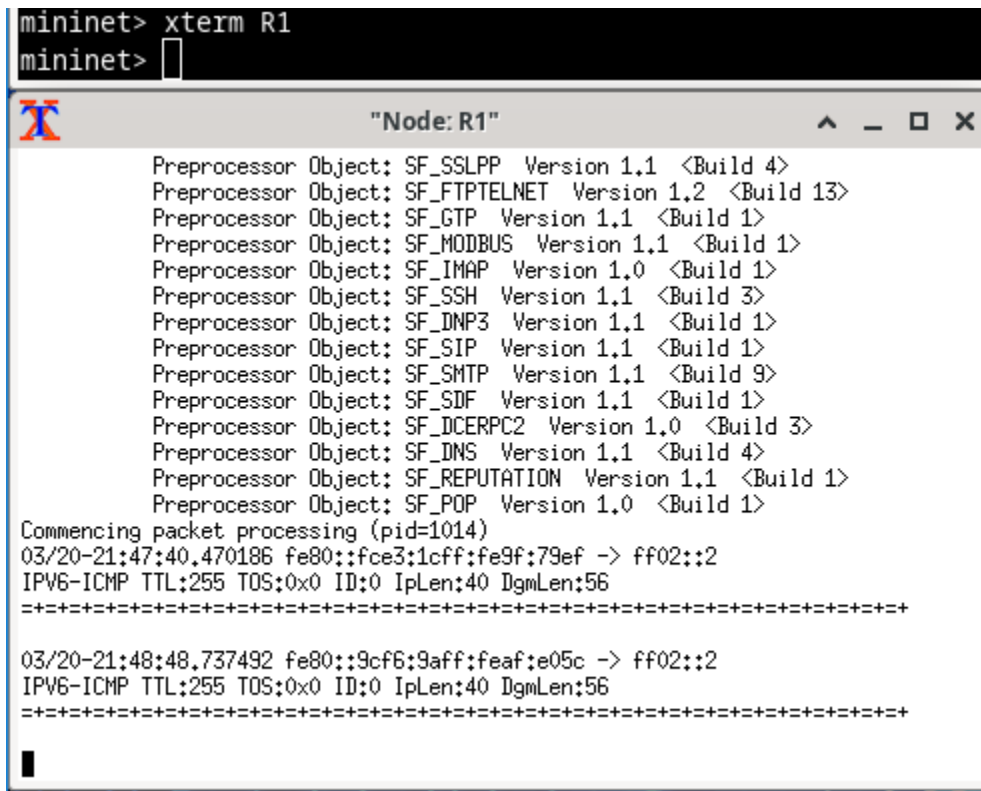
3. Dari prompt mininet, buka shell di R1 menggunakan perintah xterm R1

```
mininet> xterm R1
mininet> 
```



4. Dari shell R1, jalankan IDS berbasis Linux, Snort..

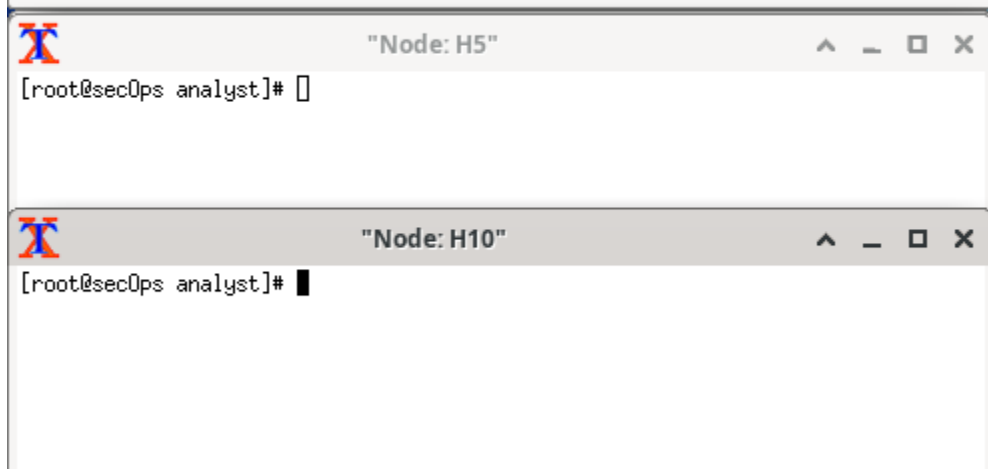
```
mininet> xterm R1
mininet> 
```



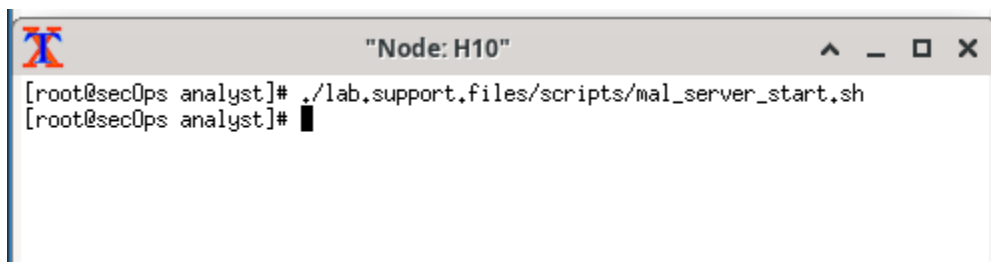
```
Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>
Preprocessor Object: SF_FTPTELNET Version 1.2 <Build 13>
Preprocessor Object: SF_GTP Version 1.1 <Build 1>
Preprocessor Object: SF_MODBUS Version 1.1 <Build 1>
Preprocessor Object: SF_IMAP Version 1.0 <Build 1>
Preprocessor Object: SF_SSH Version 1.1 <Build 3>
Preprocessor Object: SF_DNP3 Version 1.1 <Build 1>
Preprocessor Object: SF_SIP Version 1.1 <Build 1>
Preprocessor Object: SF_SMTP Version 1.1 <Build 9>
Preprocessor Object: SF_SDF Version 1.1 <Build 1>
Preprocessor Object: SF_DCERPC2 Version 1.0 <Build 3>
Preprocessor Object: SF_DNS Version 1.1 <Build 4>
Preprocessor Object: SF_REPUTATION Version 1.1 <Build 1>
Preprocessor Object: SF_POP Version 1.0 <Build 1>
Commencing packet processing (pid=1014)
03/20-21:47:40.470186 fe80::fce3:1cff:fe9f:79ef -> ff02::2
IPV6-ICMP TTL:255 TOS:0x0 ID:0 IpLen:40 DgmLen:56
=====
03/20-21:48:48.737492 fe80::9cf6:9aff:feaf:e05c -> ff02::2
IPV6-ICMP TTL:255 TOS:0x0 ID:0 IpLen:40 DgmLen:56
=====
```

4. Dari prompt mininet CyberOps Workstation VM, buka shell untuk host H5 dan H10.

```
mininet> xterm H5
mininet> xterm H10
mininet> 
```



5. H10 akan mensimulasikan server di Internet yang menghosting malware. Pada H10, jalankan skrip mal_server_start.sh untuk memulai server



```
[root@secOps analyst]# ./lab.support.files/scripts/mal_server_start.sh
[root@secOps analyst]# 
```

6. Pada H10, gunakan netstat dengan opsi -tunpa untuk memverifikasi bahwa server web sedang berjalan. Saat digunakan seperti yang ditunjukkan di bawah ini, netstat mencantumkan semua port yang saat ini ditetapkan ke layanan:

```
[root@secOps analyst]# netstat -tunpa
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
PID/Program name
tcp        0      0 0.0.0.0:6666             0.0.0.0:*               LISTEN
1075/nginx: master
[root@secOps analyst]# 
```

7. Buka terminal R1 lain dengan memasukkan xterm R1 lagi di jendela terminal VM CyberOps Workstation. jalankan perintah tail dengan opsi -f untuk memantau file /var/log/snort/alert secara real-time. File ini adalah tempat snort dikonfigurasi untuk merekam peringatan

```
Try 'tail --help' for more information.
[root@secOps analyst]# tail -f /var/log/snort/alert
```

8. Dari H5, gunakan perintah wget untuk mengunduh file bernama W32.Nimda.Amm.exe

```
"Node: H5"
[root@secOps analyst]# wget 209.165.202.133:6666/W32.Nimda.Amm.exe
--2023-03-20 21:59:30-- http://209.165.202.133:6666/W32.Nimda.Amm.exe
Connecting to 209.165.202.133:6666... connected.
HTTP request sent, awaiting response... 200 OK
Length: 345088 (337K) [application/octet-stream]
Saving to: 'W32.Nimda.Amm.exe.1'

W32.Nimda.Amm.exe.1 100%[=====>] 337,00K --.-KB/s in 0.02s

2023-03-20 21:59:30 (17.1 MB/s) - 'W32.Nimda.Amm.exe.1' saved [345088/345088]

[root@secOps analyst]#
```

9. Saat file berbahaya sedang transit R1, IDS, Snort, dapat memeriksa muatannya. Payload cocok dengan setidaknya satu tanda tangan yang dikonfigurasi di Snort dan memicu peringatan di jendela terminal R1 kedua (tab tempat tail -f berjalan). Entri peringatan ditunjukkan di bawah ini. Stempel waktu Anda akan berbeda

```
"Node: R1"
[root@secOps analyst]# tail -f/var/log/snort/alert
tail: invalid option -- '/'
Try 'tail --help' for more information.
[root@secOps analyst]# tail -f /var/log/snort/alert
03/20-21:59:30.773070 00000000 [1:1000003:0] Malicious Server Hit! [0000] [Priority:
0] {TCP} 209.165.200.235:49672 -> 209.165.202.133:6666
```

10. Pada H5, gunakan perintah tcpdump untuk merekam peristiwa dan mengunduh file malware lagi sehingga Anda dapat merekam transaksi.

```
[root@secOps analyst]# tcpdump -i H5-eth0 -w nimda.download.pcap &
[1] 1137
[root@secOps analyst]# tcpdump: listening on H5-eth0, link-type EN10MB (Ethernet
), capture size 262144 bytes
SS
```

11. Pada H5, jalankan kembali perintah atau gunakan panah atas untuk memanggilnya kembali dari fasilitas riwayat perintah

```
[root@secOps analyst]# wget 209.165.202.133:6666/W32.Nimda.Amm.exe
--2023-03-20 22:04:53-- http://209.165.202.133:6666/W32.Nimda.Amm.exe
Connecting to 209.165.202.133:6666... connected.
HTTP request sent, awaiting response... 200 OK
Length: 345088 (337K) [application/octet-stream]
Saving to: 'W32.Nimda.Amm.exe.2'

W32.Nimda.Amm.exe.2 100%[=====>] 337,00K --.-KB/s in 0.02s

2023-03-20 22:04:54 (21.2 MB/s) - 'W32.Nimda.Amm.exe.2' saved [345088/345088]

[root@secOps analyst]#
```

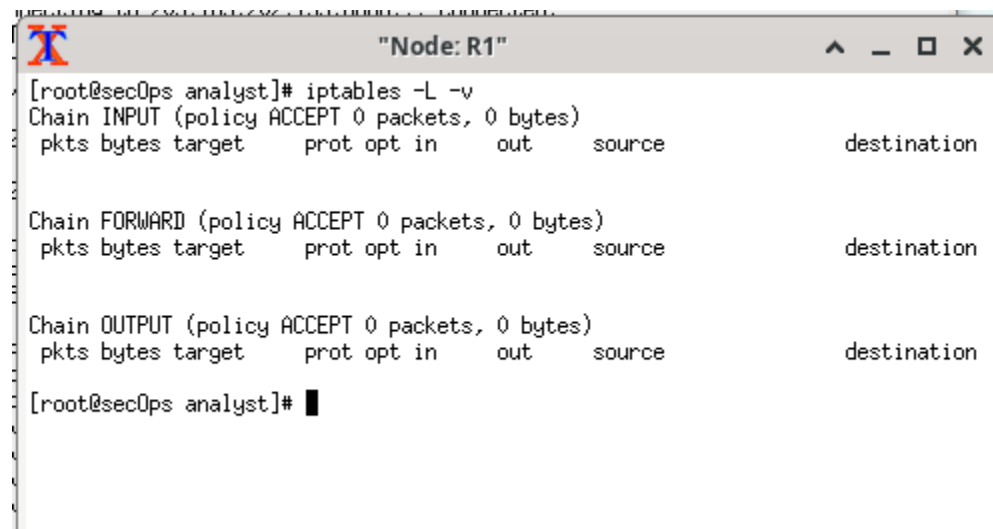
12. Hentikan pengambilan dengan membawa tcpdump ke latar depan dengan perintah fg.

```
[root@secOps analyst]# fg
tcpdump -i H5-eth0 -w nimda.download.pcap
^C56 packets captured
56 packets received by filter
0 packets dropped by kernel
[root@secOps analyst]#
```

13. Pada H5, Gunakan perintah ls untuk memverifikasi file pcap sebenarnya disimpan ke disk dan memiliki ukuran lebih besar dari nol:

```
[root@secOps analyst]# ls -l
total 4292
drwxr-xr-x 2 analyst analyst 4096 Mar 13 21:54 analyst
drwxr-xr-x 2 analyst analyst 4096 May 20 2020 Desktop
drwxr-xr-x 3 analyst analyst 4096 Apr 2 2020 Downloads
-rw-r--r-- 1 root root 9238 Feb 20 20:41 httpdump.pcap
-rw-r--r-- 1 root root 2965504 Feb 20 21:48 httpsdump.pcap
drwxr-xr-x 9 analyst analyst 4096 Jul 15 2020 lab.support.files
-rw-r--r-- 1 root root 350046 Mar 20 22:07 nimda.download.pcap
drwxr-xr-x 2 analyst analyst 4096 Mar 21 2018 second_drive
-rw-r--r-- 1 analyst analyst 345088 Mar 13 21:55 W32.Nimda.Amm.exe
-rw-r--r-- 1 root root 345088 Mar 23 2018 W32.Nimda.Amm.exe.1
-rw-r--r-- 1 root root 345088 Mar 23 2018 W32.Nimda.Amm.exe.2
[root@secOps analyst]#
```

14. Di VM CyberOps Workstation, mulai jendela terminal R1 ketiga. Di terminal R1 baru, gunakan perintah iptables untuk membuat daftar rantai dan aturannya yang sedang digunakan:




```
[root@secOps analyst]# iptables -L -v
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source destination

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

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

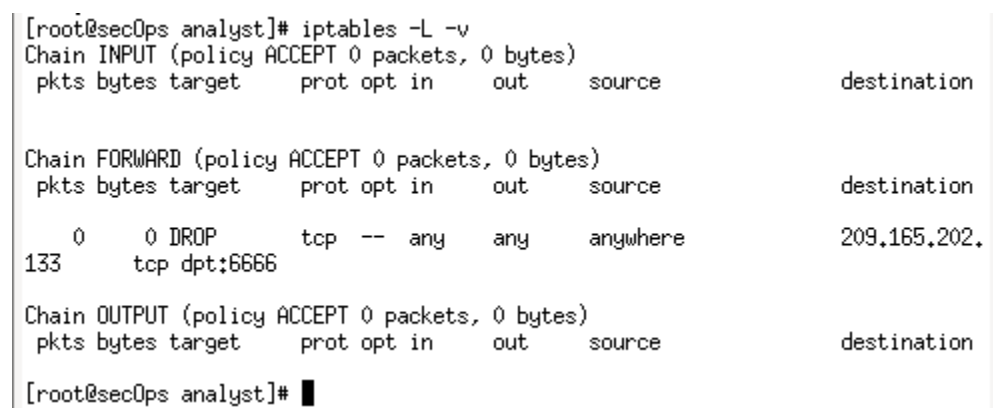
[root@secOps analyst]#
```

15. Agar komputer pengguna tidak terhubung ke server yang diidentifikasi di Langkah 1, tambahkan aturan berikut ke rantai FORWARD di R1:



```
[root@secOps analyst]# iptables -I FORWARD -p tcp -d 209.165.202.133 --dport 6666 -j DROP
[root@secOps analyst]#
```

16. Gunakan perintah iptables lagi untuk memastikan aturan telah ditambahkan ke rantai FORWARD



```
[root@secOps analyst]# iptables -L -v
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source destination
    0    0 DROP      tcp  --  any    any    anywhere    209.165.202.133 tcp dpt:6666

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

[root@secOps analyst]#
```

17. Pada H5, coba unduh file lagi

```
[root@secOps analyst]# wget 209.165.202.133:6666/W32.Nimda.Amm.exe
--2023-03-20 22:17:33-- http://209.165.202.133:6666/W32.Nimda.Amm.exe
Connecting to 209.165.202.133:6666... failed: Connection timed out.
Retrying.

--2023-03-20 22:19:44-- (try: 2) http://209.165.202.133:6666/W32.Nimda.Amm.exe
Connecting to 209.165.202.133:6666... failed: Connection timed out.
Retrying.

--2023-03-20 22:21:55-- (try: 3) http://209.165.202.133:6666/W32.Nimda.Amm.exe
Connecting to 209.165.202.133:6666... ^C
[root@secOps analyst]#
```

18. Hentikan dan Hapus Proses Mininet

```
[analyst@secOps ~]$ sudo mn -c
[sudo] password for analyst:
*** Removing excess controllers/ofprotocols/ofdatapaths/pings/noxes
killall controller ofprotocol ofdatapath ping nox_core lt-nox_core ovs-openflowd ovs-controller udph
ull
killall -9 controller ofprotocol ofdatapath ping nox_core lt-nox_core ovs-openflowd ovs-controller u
v/null
pkill -9 -f "sudo mnexec"
*** Removing junk from /tmp
rm -f /tmp/vconn* /tmp/vlogs* /tmp/*.out /tmp/*.log
*** Removing old X11 tunnels
*** Removing excess kernel datapaths
ps ax | egrep -o 'dp[0-9]+' | sed 's/dp/nl:/'
*** Removing OVS datapaths
ovs-vsctl --timeout=1 list-br
ovs-vsctl --timeout=1 list-br
*** Removing all links of the pattern foo-ethX
ip link show | egrep -o '([_-[:alnum:]]+-eth[[:digit:]]+)'
ip link show
*** Killing stale mininet node processes
pkill -9 -f mininet:
*** Shutting down stale tunnels
pkill -9 -f Tunnel=Ethernet
pkill -9 -f .ssh/mn
rm -f ~/.ssh/mn/*
*** Cleanup complete.
[analyst@secOps ~]$
```