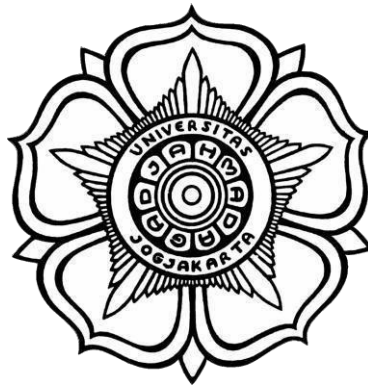


LAPORAN PRAKTIKUM
KEAMANAN INFORMASI 1
UNIT 6

“Snort dan Firewall Rules”



Disusun Oleh:

Nama	: Saadah Mardatillah
NIM	: 21/473000/SV/18792
Kelas	: Teknologi Rekayasa Internet A
Hari, Tanggal	: Selasa, 28 Maret 2023
Dosen Pengampu	: Anni Karimatul Fauziyyah, S.Kom.,M.Eng.
Asisten Praktikum	: 1. Annisa Nurul Ramadhani Novelika 2. Gabriella Alvera Chaterine

PROGRAM STUDI DIPLOMA IV TEKNOLOGI REKAYASA INTERNET

DEPARTEMEN TEKNIK ELEKTRO DAN INFORMATIKA

SEKOLAH VOKASI

UNIVERSITAS GADJAH MADA

2023

Unit 6

Snort dan Firewall Rules

I. Tujuan

1. Mahasiswa dapat mempersiapkan pembelajaran dari lingkungan virtual
2. Mahasiswa dapat melakukan analisis praktikum pada firewall dan log IDS
3. Mahasiswa dapat melakukan pemberhentian dan menghapus proses miniset

II. Alat dan Bahan

1. Mesin virtual CyberOps Workstation
2. Laptop
3. Koneksi Internet

III. Latar Belakang

SNORT adalah sistem deteksi intrusi (IDS) dan sistem pencegahan intrusi (IPS) sumber terbuka yang kuat yang menyediakan analisis lalu lintas jaringan secara real-time dan pencatatan paket data. SNORT menggunakan bahasa berbasis aturan yang menggabungkan metode anomali, protokol, dan pemeriksaan tanda tangan untuk mendeteksi aktivitas yang berpotensi berbahaya. Dengan menggunakan SNORT, admin jaringan dapat menemukan serangan denial-of-service (DoS) dan serangan DoS terdistribusi (DDoS), serangan Common Gateway Interface (CGI), buffer yang meluap, dan pemindaian port siluman. SNORT menciptakan serangkaian aturan yang mendefinisikan aktivitas jaringan yang berbahaya, mengidentifikasi paket-paket berbahaya, dan mengirimkan peringatan kepada pengguna. SNORT adalah perangkat lunak sumber terbuka yang bebas digunakan yang dapat digunakan oleh perorangan dan organisasi. Bahasa aturan SNORT menentukan lalu lintas jaringan mana yang harus dikumpulkan dan apa yang harus terjadi ketika mendeteksi paket berbahaya. Snort adalah Sistem Pencegahan Penyusupan (IPS) *Open Source* terkemuka di dunia. Snort IPS menggunakan serangkaian aturan yang membantu mendefinisikan aktivitas jaringan yang berbahaya dan menggunakan aturan-aturan tersebut untuk menemukan paket yang cocok dengan aturan tersebut dan menghasilkan peringatan bagi pengguna. Snort juga dapat digunakan secara *inline* untuk menghentikan paket-paket ini. Snort

memiliki tiga kegunaan utama: Sebagai sniffer paket seperti tcpdump, sebagai pencatat paket - yang berguna untuk debugging lalu lintas jaringan, atau dapat digunakan sebagai sistem pencegahan penyusupan jaringan yang lengkap. Snort bisa diunduh dan dikonfigurasi untuk penggunaan pribadi dan bisnis.

Firewall rules adalah proses peninjauan dan pengoptimalan aturan firewall secara berkala. Proses ini melibatkan hal-hal berikut:

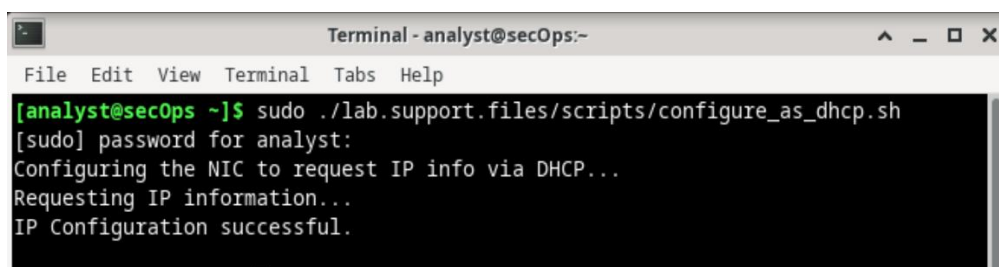
1. Menganalisis anomali aturan yang memengaruhi kinerja firewall.
2. Menyusun ulang aturan yang ada untuk meningkatkan kinerja aturan.
3. Mengidentifikasi dan menghapus aturan yang tidak digunakan.
4. Menganalisis dampak aturan baru terhadap kumpulan aturan yang ada sebelum membuatnya aktif di firewall.

Mengapa *tools firewall rules* penting?

Firewall memberikan perlindungan dari ancaman eksternal dengan melindungi jaringan Anda dan mencegah lalu lintas internet yang berbahaya. Dengan seperangkat aturan dan kebijakan yang stabil dapat menjaga perusahaan tetap aman dari peretas. Namun, melacak kebijakan keamanan firewall merupakan tantangan tersendiri; organisasi kecil dapat memiliki ratusan aturan untuk dikelola, sementara organisasi yang lebih besar mungkin memiliki ribuan aturan. Banyak dari aturan-aturan ini sudah ada sejak lebih dari lima hingga sepuluh tahun yang lalu, dan sering kali ada kekurangan kesinambungan dalam mendefinisikan aturan-aturan baru karena sebagian besar diwarisi dari para pendahulunya. Salah kelola aturan ini sangat memengaruhi kinerja firewall, membuat jaringan dapat rentan terhadap pelanggaran keamanan.

IV. Data Instruksi Kerja

1. Konfigurasi jaringan dengan menjalankan skrip `configure_as_dhcp.sh`.



```
Terminal - analyst@secOps:~
File Edit View Terminal Tabs Help
[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.
```

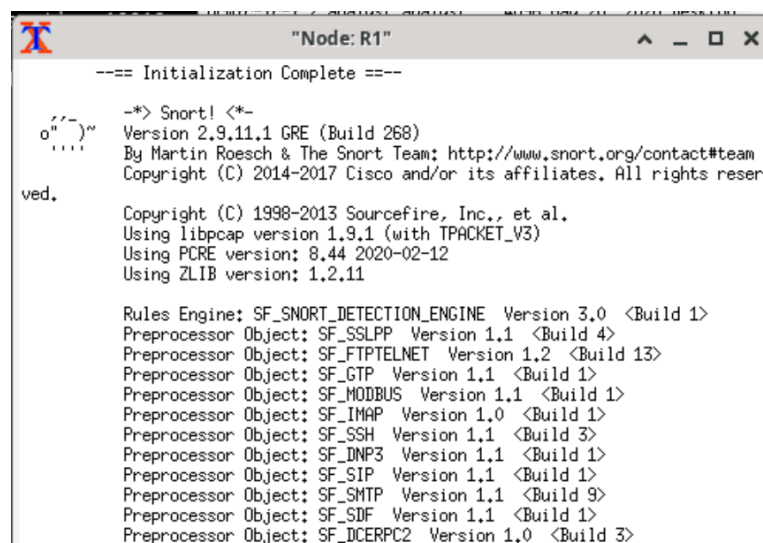
2. Melakukan ping ke www.cisco.com.

```
[analyst@secOps ~]$ ping www.cisco.com
PING e2867.dsca.akamaiedge.net (104.93.203.251) 56(84) bytes of data.
64 bytes from 104.93.203.251: icmp_seq=1 ttl=55 time=29.3 ms
64 bytes from 104.93.203.251: icmp_seq=2 ttl=55 time=59.4 ms
64 bytes from 104.93.203.251: icmp_seq=3 ttl=55 time=25.1 ms
64 bytes from 104.93.203.251: icmp_seq=4 ttl=55 time=36.6 ms
64 bytes from 104.93.203.251: icmp_seq=5 ttl=55 time=99.1 ms
64 bytes from 104.93.203.251: icmp_seq=6 ttl=55 time=46.9 ms
64 bytes from 104.93.203.251: icmp_seq=7 ttl=55 time=42.4 ms
64 bytes from 104.93.203.251: icmp_seq=8 ttl=55 time=43.3 ms
64 bytes from 104.93.203.251: icmp_seq=9 ttl=55 time=59.4 ms
64 bytes from 104.93.203.251: icmp_seq=10 ttl=55 time=47.8 ms
64 bytes from 104.93.203.251: icmp_seq=11 ttl=55 time=58.2 ms
64 bytes from 104.93.203.251: icmp_seq=12 ttl=55 time=348 ms
64 bytes from 104.93.203.251: icmp_seq=13 ttl=55 time=198 ms
^C
--- e2867.dsca.akamaiedge.net ping statistics ---
13 packets transmitted, 13 received, 0% packet loss, time 21036ms
rtt min/avg/max/mdev = 25.086/84.159/348.399/87.641 ms
```

3. Menjalankan skrip miniset.

```
[analyst@secOps ~]$ sudo ./lab.support.files/scripts/cyberops_extended_topo_no_f
w.py
*** 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> xterm R1
```

4. Menjalankan perintah dari prompt miniset untuk buka shell R1.



```
Node: R1
--== Initialization Complete ==--

--*) Snort! <*-
Version 2.9.11.1 GRE (Build 268)
By Martin Roesch & The Snort Team; http://www.snort.org/contact#team
Copyright (C) 2014-2017 Cisco and/or its affiliates. All rights reserved.

Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using libpcap version 1.9.1 (with TPACKET_V3)
Using PCRE version: 8.44 2020-02-12
Using ZLIB version: 1.2.11

Rules Engine: SF_SNORT_DETECTION_ENGINE Version 3.0 <Build 1>
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_SHTP Version 1.1 <Build 9>
Preprocessor Object: SF_SDF Version 1.1 <Build 1>
Preprocessor Object: SF_DCERPC2 Version 1.0 <Build 3>
```

5. Membuka shell host H5 dan H10.

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

6. H10 melakukan simulasi server di internet yang melakukan hosting malware.

```
"Node: H10"
[root@secOps analyst]# ./lab.support.files/scripts/mal_server_start.sh
[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
927/nginx: master p...
```

7. Pada tab terminal R1 yang baru, menjalankan perintah tail dengan opsi -f untuk memantau file /var/log/snort secara real-time.

```
"Node: R1"
[root@secOps analyst]# tail -f /var/log/snort/alert
```

Note: diawal config terminal masih kosong karena belum ada aktivitas untuk perhitungan real-time.

8. Pada terminal H5, menggunakan 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
--2023-03-21 11:29:13-- 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.7'

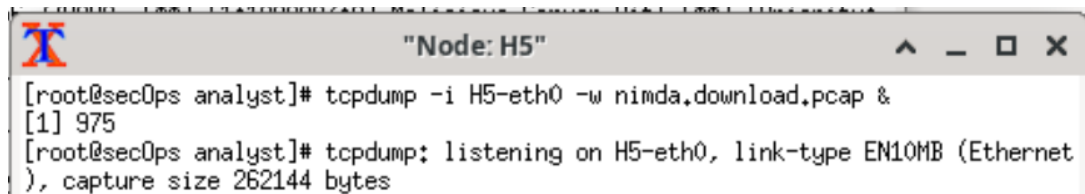
W32.Nimda.Amm.exe.7 100%[=====>] 337.00K --.-KB/s in 0.007s

2023-03-21 11:29:13 (45.9 MB/s) - 'W32.Nimda.Amm.exe.7' saved [345088/345088]
```

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:

```
03/21-11:28:56.778080  [**] [1:1000003;0] Malicious Server Hit! [**] [Priority:
0] {TCP} 209.165.200.235:52798 -> 209.165.202.133:6666
03/21-11:29:13.453925  [**] [1:1000003;0] Malicious Server Hit! [**] [Priority:
0] {TCP} 209.165.200.235:52800 -> 209.165.202.133:6666
03/21-11:32:49.324072  [**] [1:1000003;0] Malicious Server Hit! [**] [Priority:
0] {TCP} 209.165.200.235:52802 -> 209.165.202.133:6666
03/21-11:33:23.771505  [**] [1:1000003;0] Malicious Server Hit! [**] [Priority:
0] {TCP} 209.165.200.235:52804 -> 209.165.202.133:6666
```

10. Pada H5, gunakan perintah tcpdump untuk merekam peristiwa dan mengunduh file malware lagi sehingga Anda dapat merekam transaksi. Keluarkan perintah berikut di bawah ini mulai pengambilan paket:



```
"Node: H5"
[root@secOps analyst]# tcpdump -i H5-eth0 -w nimda.download.pcap &
[1] 975
[root@secOps analyst]# tcpdump: listening on H5-eth0, link-type EN10MB (Ethernet
), capture size 262144 bytes
```

11. Sekarang tcpdump menangkap paket, unduh malware lagi. Pada H5, jalankan kembali perintah atau gunakan panah atas untuk memanggilnya kembali dari fasilitas riwayat perintah.

```
tcpdump -i H5-eth0 -w nwget 209.165.202.133:6666/W32.Nimda.Amm.exe
--2023-03-21 11:33:23-- 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.8'

W32.Nimda.Amm.exe.8 100%[=====>] 337.00K --.-KB/s in 0.005s

2023-03-21 11:33:23 (60.0 MB/s) - 'W32.Nimda.Amm.exe.8' saved [345088/345088]
```

12. Hentikan pengambilan dengan membawa tcpdump ke latar depan dengan perintah fg. Karena tcpdump adalah satu-satunya proses yang dikirim ke latar belakang, PID tidak perlu ditentukan.

```
[root@secOps analyst]# fg
tcpdump -i H5-eth0 -w nimda.download.pcap

58 packets captured
58 packets received by filter
0 packets dropped by kernel
```

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

```
^C[root@secOps analyst]# ls -l
total 12128
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 229695 Feb 27 08:15 httpdump.pcap
-rw-r--r-- 1 root root 8677167 Feb 27 08:30 httpsdump.pcap
drwxr-xr-x 9 analyst analyst 4096 Mar 13 22:01 lab.support.files
-rw-r--r-- 1 root root 350210 Mar 21 11:34 nimda.download.pcap
drwxr-xr-x 2 analyst analyst 4096 Mar 21 2018 second_drive
drwxr-xr-x 2 analyst analyst 4096 Feb 27 08:37 Tugas_Unit3_SaadahM
-rw-r--r-- 1 analyst analyst 345088 Mar 13 21:58 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
-rw-r--r-- 1 root root 345088 Mar 23 2018 W32.Nimda.Amm.exe.3
-rw-r--r-- 1 root root 345088 Mar 23 2018 W32.Nimda.Amm.exe.4
-rw-r--r-- 1 root root 345088 Mar 23 2018 W32.Nimda.Amm.exe.5
-rw-r--r-- 1 root root 345088 Mar 23 2018 W32.Nimda.Amm.exe.6
-rw-r--r-- 1 root root 345088 Mar 23 2018 W32.Nimda.Amm.exe.7
-rw-r--r-- 1 root root 345088 Mar 23 2018 W32.Nimda.Amm.exe.8
[root@secOps analyst]#
```

14. Di VM CyberOps Workstation, mulai jendela terminal R1 ketiga.

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

Di jendela terminal R1 baru, gunakan perintah iptables untuk membuat daftar rantai dan aturannya yang sedang digunakan:

```
Node: R1
[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
```

15. Koneksi ke server menghasilkan paket yang harus melintasi firewall iptables di R1. Paket yang melintasi firewall ditangani oleh aturan FORWARD dan oleh karena itu, rantai itulah yang akan menerima aturan pemblokiran. 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. VM CyberOps Workstation mungkin memerlukan beberapa detik untuk menghasilkan output.

```
[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
```

17. Pada H5, coba unduh file kembali

```
[root@secOps analyst]# wget 209.165.202.133:6666/W32.Nimda.Amm.exe
--2023-03-21 12:05:24-- http://209.165.202.133:6666/W32.Nimda.Amm.exe
Connecting to 209.165.202.133:6666... failed: Connection timed out.
Retrying.
```

Uji coba firewall yang sedang berjalan:

```
[root@secOps analyst]# iptables -L -v
Chain INPUT (policy ACCEPT 2 packets, 204 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

6        360 DROP        tcp  --  any    any     anywhere          209.165.202.133
tcp dpt:6666

Chain OUTPUT (policy ACCEPT 2 packets, 148 bytes)
 pkts bytes target      prot opt in      out     source            destination
```


18. Menghentikan dan menghapus proses miniset

```
[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 udpbwtest mnexec ivs 2> /dev/null
killall -9 controller ofprotocol ofdatapath ping nox_core lt-nox_core ovs-openfl
owd ovs-controller udpbwtest mnexec ivs 2> /dev/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 ~]$
```

V. Analisis

Analisis pada pembahasan praktikum ini adalah dengan menjawab pertanyaan yang diberikan pada setiap kasus percobaan langkah – langkah praktikum.

Langkah pertama yang dilakukan pada praktikum adalah mempersiapkan lingkungan virtual dengan menjalankan oracle virtualbox pada laptop dengan membuka terminal dan melakukan konfigurasi jaringan dan menjalankan skrip configure. Pada kali ini mahasiswa menjalankan konfigurasi menggunakan VM CyberOps Workstation. Konfigurasi pertama dengan menjalankan perintah **sudo ./lab.support.files/scripts/configure_as_dhcp.sh** yang akan digunakan untuk memverifikasi CyberOps pada VM untuk memiliki alamat IP jaringan lokal. Pengujian konektivitas pada uji coba ini dengan melakukan ping server web publik yaitu www.cisco.com. Bukti uji coba dapat berjalan terlampir pada data instruksi kerja, dimana program dapat dijalankan.

Bagian kedua adalah firewall dan IDS Logs. Dimana firewall dan IDS digunakan dalam melakukan otomatisasi Sebagian tugas dari pemantauan lalu lintas. Pada bagian kedua ini kita memulai dengan menjalankan skrip mininet. Dimana program dapat dijalankan sesuai dengan terlampir pada data instruksi kerja. Penyelesaian soal adalah sebagai berikut:

Soal 1:

Shell R1 terbuka di jendela terminal dengan teks hitam dan latar belakang putih.

Pengguna apa yang masuk ke shell itu? Ini indikatornya apa??

Penyelesaian:

Pengguna yang masuk ke shell adalah pengguna root. Ditunjukkan oleh tanda # setelah prompt.

Disini kita tidak hanya menjalankan pada shell R1 namun menjalankan perintah pada host H5 dan H10. Pada H10, menggunakan netstat dengan opsi -tunpa untuk memverifikasi bahwa server web sedang berjalan. Saat digunakan seperti, netstat mencantumkan semua port yang saat ini ditetapkan ke layanan.

Soal 2:

- a. Port apa yang digunakan saat berkomunikasi dengan server web malware? Apa indikatornya?
- b. Apakah file telah diunduh sepenuhnya?
- c. Apakah IDS menghasilkan peringatan yang terkait dengan unduhan file?

Penyelesaian:

- a. Port 666. Port ini ditentukan dalam URL, setelah tanda pemisah :
- b. File sepenuhnya telah diunduh
- c. IDS telah menghasilkan peringatan yang memiliki kaitan dengan unduhan file

Soal 3:

- a. Berdasarkan peringatan yang ditunjukkan di atas, apa alamat IPv4 sumber dan tujuan yang digunakan dalam transaksi?
- b. Berdasarkan alert di atas, port sumber dan tujuan apa yang digunakan dalam transaksi?
- c. Berdasarkan peringatan yang ditunjukkan di atas, kapan pengunduhan dilakukan?
- d. Berdasarkan peringatan yang ditunjukkan di atas, apa pesan yang direkam IDS signature?

Penyelesaian:

- a. Alamat IP sumber adalah: 209.165.200.235 dan alamat IP tujuan adalah: 209.165.202.133
- b. Port sumber adalah: 52804 dan port tujuan adalah 666
- c. Pengunduhan dilakukan pada tanggal 21 Maret 2023, pukul 11:29
- d. Pesan yang direkam IDS signature adalah Malicious Server Hit!

Soal 4:

Bagaimana file PCAP ini berguna bagi analis keamanan?

Penyelesaian:

File PCAP berisi paket-paket yang terkait dengan lalu lintas yang dilihat oleh NIC yang menangkap. Dengan begitu, PCAP sangat berguna untuk menelusuri kembali peristiwa jaringan seperti komunikasi ke titik akhir yang berbahaya. Alat seperti Wireshark dapat digunakan untuk memfasilitasi analisis PCAP.

Soal 5:

Rantai apa yang saat ini digunakan oleh R1?

Penyelesaian:

INPUT, FORWARD, OUTPUT

Soal 6:

- a. Apakah unduhan berhasil kali ini? Jelaskan
- b. Apa pendekatan yang lebih agresif tetapi juga valid saat memblokir server yang melanggar?

Penyelesaian:

Dalam proses menentukan IP, protokol, dan port, sebuah aturan dapat dengan mudah memblokir alamat IP server. Ini akan memutus akses ke server tersebut dari jaringan internal.

VI. Kesimpulan

Dari proses praktikum ini dapat disimpulkan bahwa:

1. Snort dapat memberikan peringatan adanya sebuah serangan keamanan, sehingga dapat meningkatkan keamanan jaringan. Dapat atau tidaknya sebuah serangan terdeteksi oleh Snort IDS tergantung dari ada tidaknya rule dengan jenis signature pada sebuah pola serangan.
2. Salah satu cara untuk meningkatkan keamanan dalam jaringan adalah dengan mengimplementasikan *Intrusion Detection System* (IDS), merupakan sebuah sistem yang digunakan untuk melakukan deteksi adanya usaha-usaha penyusupan terhadap sebuah sistem dengan melakukan pengamatan trafik secara *real-time*.

DAFTAR PUSTAKA

- [1] Afandi, A., & Fatkhuroyan, F. (2019). Sistem Deteksi Intrusi dengan Snort. *Jurnal Telematika Komunikasi dan Informatika*, 7(1), 25-30. <https://doi.org/10.13140/RG.2.2.25139.46887>
- [2] Arfianto, M. A., & Arymurthy, A. M. (2019). Implementasi Snort sebagai Alat Pendeteksi Intrusi pada Jaringan Komputer. *Jurnal Ilmiah DASI (Datalogi dan Sistem Informasi)*, 20(2), 123-130. <https://doi.org/10.30743/JID.V20I2.1267>

- [3] Configure Snort firewalls. (n.d.). Retrieved September 2021, from <https://www.manageengine.com/products/firewall/help/configure-snort-firewalls.html>
- [4] Firewall Rule Management. (n.d.). Retrieved September 2021, from <https://www.manageengine.com/products/firewall/firewall-rule-management.html>
- [5] Snort. (n.d.). Retrieved September 2021, from <https://www.fortinet.com/resources/cyberglossary/snort>

LINK GITHUB

https://github.com/saadahmardatillah/Unit6_KeamananInformasi1