SNORT

This module will explore the 'foremost Open Source Intrusion Prevention System (IPS) in the world. Snort IPS uses a series of rules that help define malicious network activity and uses those rules to find packets that match against them and generates alerts for users'. (SNORT. N.d. para.1)

References

SNORT. N.d. 'What is Snort?'
https://www.snort.org/
Accessed: 19 April 2022

Q1

```
log tcp any any → any 22 (msg: "Someone's trying to use SSH!";)
```

What type of connection does this rule apply to? (include protocol name)

- The Snort manual defines four protocols that it analyses - TCP, and UDP. ICP and IP.
- TCP connections

What traffic is monitored? (include source, destination, ports, and directions)

The rule will monitor:

- Source IP - Any
- Source port - Any
- Dst IP - Any
- Dst port - 22
- Direction - Source to destination flow

Any additional requirements/characteristics in the traffic that the rule looks for?

- No

What happens when the rule is matched? (include action)

- *log* is used, which means the corresponding packet and the message is logged.

```
alert ip any any → any any (msg: "IP Packet detected"; sid:1000002; rev:0;)
```

What type of connection does this rule apply to? (include protocol name)

- The Snort manual defines four protocols that it analyses - TCP, and UDP. ICP and IP.
- In the above rule, the IP protocol is selected for analysis.

What traffic is monitored? (include source, destination, ports, and directions)

The rule will monitor:

- Source IP - any
- Source port - any
- Dst IP - any
- Dst port - any
- Direction - Source to destination flow

Any additional requirements/characteristics in the traffic that the rule looks for?

- No

What happens when the rule is matched? (include action)

- Alert is generated, meaning the *msg* is printed to the console. Next, the corresponding packet is logged. The message will also contain the SID and REV.
- SID - a keyword used to identify a snort rule. The SID can be used by network engineers to identify a particular rule. Locally defined rules must be denoted by values above 1,000,000.
- REV - a keyword used to identify revisions to a snort rule.

```
log udp any any → 192.168.1.0/24 1:1024
```

What type of connection does this rule apply to? (include protocol name)

- UDP

What traffic is monitored? (include source, destination, ports, and directions)

The rule will monitor:

- Source IP - any
- Source port - any
- DST IP - 192.168.1.1 - 192.168.1.255
- DST port - 1 a 1024
- Direction - Source to destination flow

- 192.168.1.0/24 represents a block of IP addresses using the Classless Inter-Domain Routing (CIDR) scheme. By this token, the rule applies to IP addresses from 192.168.1.1 to 192.168.1.255. CIDR designation offers a short-hand method to designate a large pool of addresses.
- Port ranges are denoted using ':' This means, the rule will apply to ports from 0 to 1024.

Any additional requirements/characteristics in the traffic that the rule looks for?

- No

What happens when the rule is matched? (include action)

- The Rule Header indicates what to do in the event that the rule is matched.
- In this case, *log* is used. This means the corresponding packet is logged.

```
alert tcp any any → any any (msg:"Possible exploit";
content:"|90|"; offset:40; depth:75";)
```

What type of connection this rule is applied to? (include protocol name)

- TCP

What traffic is monitored? (include source, destination, ports, and directions)

The rule will monitor:

- Source IP - any
- Source port - any
- DST IP - any
- DST port - any
- Direction - Source to destination flow

Any additional requirements/characteristics in the traffic that the rule looks for?

- The *content* parameter is a component of Payload Detection Rule Options.
- The rule allows users to search packets to find the specified content in the packet.
- In the rule above, hexadecimal numbers are enclosed within the pipe (|) operators. It represents binary data.

- *Offset* modifies how the rule operates.
- An offset of 40 informs Snort to search for patterns after the 45th byte of the payload.

- *Depth* specifies how far into a packet to search for the pattern.
- A *depth* of 75 informs Snort to search for patterns within the first 45 bytes,

What happens when the rule is matched? (include action)

- Alert is generated, meaning the *msg* is printed to the console. Next, the corresponding packet is logged.

```
alert any any → any any (flags: SF; msg: "Possible SYN FIN scan";)
```

What type of connection does this rule apply to? (include protocol name)

- TCP

What traffic is monitored? (include source, destination, ports, and directions)

The rule will monitor:

- Source IP - any
- Source port - any
- DST IP - any
- DST port - any
- Direction - Source to destination flow

Any additional requirements/characteristics in the traffic that the rule looks for?

- The *flags* parameter is a component of Non-Payload Detection Rule Options.
- It is used to identify if specific TCP flag bits are present in a packet.
- The above rule defines flags that are SYN and FIN.

What happens when the rule is matched? (include action)

- Alert is generated, meaning the *msg* is printed to the console. Next, the corresponding packet is logged.

```
log tcp any :1024 → 192.168.1.0/24 500:
```

What type of connection does this rule apply to? (include protocol name)

- TCP

What traffic is monitored? (include source, destination, ports, and directions)

The rule will monitor:

- Source IP - any
- Source port - 0-1024
- DST IP - 192.168.1.1 - 192.168.1.255
- DST port - 500 - 65535
- Direction - Source to destination flow

Any additional requirements/characteristics in the traffic that the rule looks for?

- No

What happens when the rule is matched? (include action)

- *log* is used, which means the corresponding packet is logged.

```
log tcp any any → 192.168.1.0/24 !6000:6010
```

What type of connection does this rule apply to? (include protocol name)

- TCP

What traffic is monitored? (include source, destination, ports, and directions)

The rule will monitor:

- Source IP - any
- Source port - any
- DST IP - 192.168.1.1 - 192.168.1.255
- DST port - all ports except ports 6000 to 6010.
- Direction - Source to destination flow

Port negation is indicated by an asterisk (!).

Any additional requirements/characteristics in the traffic that the rule looks for?

- No

What happens when the rule is matched? (include action)

- *log* is used, which means the corresponding packet is logged.

```
alert tcp !192.168.1.0/24 any → 192.168.1.0/24 !:1024
```

What type of connection does this rule apply to? (include protocol name)

- TCP

What traffic is monitored? (include source, destination, ports, and directions)

The rule will monitor:

- Source IP - any IP address, except 192.168.1.1 to 192.168.1.255.
- Source port - any
- DST IP - 192.168.1.1 - 192.168.1.255
- DST port - all ports, except ports 0 to 1024.
- Direction - Source to destination flow

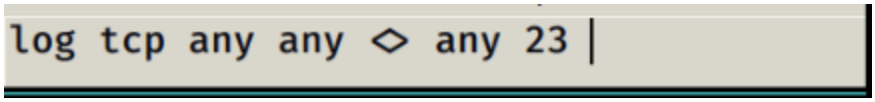
Port negation is indicated by an asterisk (!).

Any additional requirements/characteristics in the traffic that the rule looks for?

- No

What happens when the rule is matched? (include action)

- Alert is generated. Next, the corresponding packet is logged.



What type of connection does this rule apply to? (include protocol name)

- TCP

What traffic is monitored? (include source, destination, ports, and directions)

The rule will monitor:

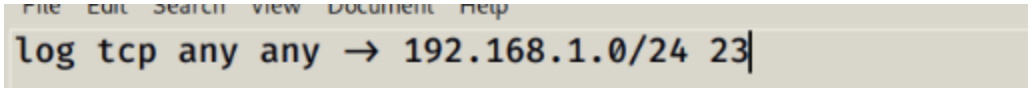
- Source IP - any
- Source port - any
- DST IP - any
- DST port - 23
- Direction - Bidirectional network flow (inbound and outbound network traffic)

Any additional requirements/characteristics in the traffic that the rule looks for?

- No

What happens when the rule is matched? (include action)

- *log* is used, which means the corresponding packet is logged.



What type of connection does this rule apply to? (include protocol name)

- TCP

What traffic is monitored? (include source, destination, ports, and directions)

The rule will monitor:

- Source IP - any
- Source port - any
- DST IP - 192.168.1.1 - 192.168.1.255
- DST port - 23
- Direction - Source to destination flow

Any additional requirements/characteristics in the traffic that the rule looks for?

- No

What happens when the rule is matched? (include action)

- *log* is used, which means the corresponding packet is logged.

```
alert tcp $HOME_NET any <> $EXTERNAL_NET 6666:7000 (msg:"CHAT IRC message"; flow:established;
content:"PRIVMSG "; nocase; classtype:policy-violation; sid:1463; rev:6;)
```

What type of connection does this rule apply to? (include protocol name)

- TCP

What traffic is monitored? (include source, destination, ports, and directions)

The rule will monitor:

- Source IP - $HOME_NET (denotes IP addresses belonging to the LAN)
- Source port - any
- DST IP - $EXTERNAL_NET (denotes IP addresses outside the LAN)
- DST port - 6666 - 7000
- Direction - bidirectional

Any additional requirements/characteristics in the traffic that the rule looks for?

- Flow: established (triggers only on established TCP connections)
- Content: “PRIVMSG” (triggers when the packet contains the specified string)
- Nocase (Snort searches for a specific pattern, however it ignores case)
- Class type: policy-violation (Used to categorise as detecting an attack ‘an attack that is part of a more general type of attack class’. (Snort. N.d. para.1)
- SID:1463 - a keyword used to identify a snort rule.
- Rev:6 - keyword used to identify revisions to a snort rule.

What happens when the rule is matched? (include action)

- Alert is generated, meaning the *msg* is printed to the console. Next, the corresponding packet is logged.

Reference

Snort. N.d. ‘3.4.6 classtype’.
http://manual-snort-org.s3-website-us-east-1.amazonaws.com/node31.html#SECTION00446000000000000000.
Accessed 18 April 2022