

NIDS using SNORT - Part 1

Objectives

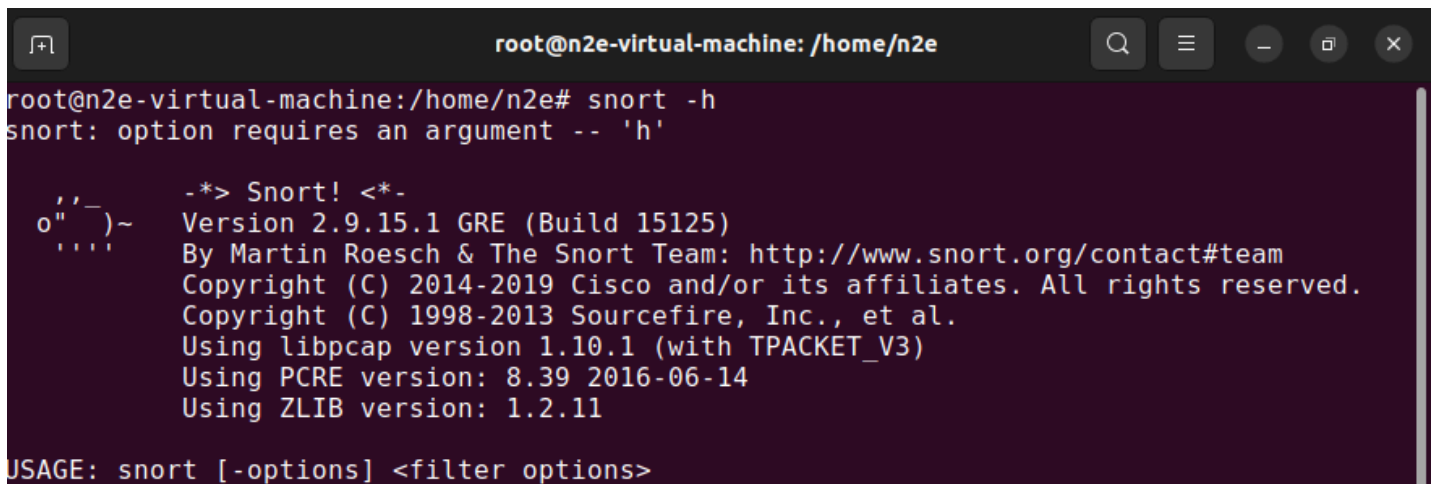
- Learn how to install and configure Snort.
- Explore Snort's modes and rule sets.
- Capture and analyze network traffic for intrusion detection.
- Write custom Snort rules for specific attack scenarios.

Prerequisites

- Wireshark or tcpdump for network traffic analysis (optional)

Snort Installation & Configuration

Using your package manager install snort v2, for Debian based distributions (eg. ubuntu) use: `sudo apt install snort`, test if its correctly installed using `snort --version`

A terminal window with a dark background and light text. The title bar shows 'root@n2e-virtual-machine: /home/n2e'. The prompt is 'root@n2e-virtual-machine:/home/n2e#'. The command 'snort -h' has been entered, and the output is displayed. The output includes a copyright notice for Snort 2.9.15.1 GRE (Build 15125) by Martin Roesch & The Snort Team, and lists dependencies like libpcap, PCRE, and ZLIB. It also shows the usage syntax: 'USAGE: snort [-options] <filter options>'.

```
root@n2e-virtual-machine:/home/n2e# snort -h
snort: option requires an argument -- 'h'

o"')~  -*> Snort! <*-
      ~  Version 2.9.15.1 GRE (Build 15125)
      ~  By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
      ~  Copyright (C) 2014-2019 Cisco and/or its affiliates. All rights reserved.
      ~  Copyright (C) 1998-2013 Sourcefire, Inc., et al.
      ~  Using libpcap version 1.10.1 (with TPACKET_V3)
      ~  Using PCRE version: 8.39 2016-06-14
      ~  Using ZLIB version: 1.2.11

USAGE: snort [-options] <filter options>
```

- Snort configuration is located at `/etc/snort/snort.conf`
- Snort can work either online and offline
- Online => listening to incoming traffic of a net interface

```
sudo snort -A <out_mode> -i <if> -c <config_path> -l <out_path>
```

- Offline => analysis of pcaps

```
sudo snort -r <pcap_path> -A <out_mode> -i <if> -c <config_path> -l <out_path>
```

- Also, we can run snort as service

```
sudo service snort start
```

```
root@n2e-virtual-machine: /home/n2e
● snort.service - LSB: Lightweight network intrusion detection system
   Loaded: loaded (/etc/init.d/snort; generated)
   Active: active (running) since Sun 2024-10-20 08:19:44 IST; 21min ago
     Docs: man:systemd-sysv-generator(8)
    Tasks: 2 (limit: 2213)
   Memory: 1.9M
      CPU: 962ms
   CGroup: /system.slice/snort.service
           └─1489 /usr/sbin/snort -m 027 -D -d -l /var/log/snort -u snort -g snort --pid-path /run/snort/ -c /etc/snort.conf

Oct 20 08:19:44 n2e-virtual-machine snort[1489]:      Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>
Oct 20 08:19:44 n2e-virtual-machine snort[1489]:      Preprocessor Object: SF_SSH Version 1.1 <Build 3>
Oct 20 08:19:44 n2e-virtual-machine snort[1489]:      Preprocessor Object: SF_DCERPC2 Version 1.0 <Build 3>
Oct 20 08:19:44 n2e-virtual-machine snort[1489]:      Preprocessor Object: SF_GTP Version 1.1 <Build 1>
Oct 20 08:19:44 n2e-virtual-machine snort[1489]:      Preprocessor Object: SF_SMTP Version 1.1 <Build 9>
Oct 20 08:19:44 n2e-virtual-machine snort[1489]:      Preprocessor Object: SF_POP Version 1.0 <Build 1>
Oct 20 08:19:44 n2e-virtual-machine snort[1489]:      Preprocessor Object: SF_FTPTELNET Version 1.2 <Build 13>
Oct 20 08:19:44 n2e-virtual-machine snort[1489]:      Preprocessor Object: SF_DNS Version 1.1 <Build 4>
Oct 20 08:19:44 n2e-virtual-machine snort[1489]:      Preprocessor Object: appid Version 1.1 <Build 5>
Oct 20 08:19:44 n2e-virtual-machine snort[1489]: Commencing packet processing (pid=1489)
```

Snort Default Rules

Snort is a signature-based IDS, and it defines rules to detect the intrusions. All rules of Snort are stored under `/etc/snort/rules` directory. The screenshot below shows the files that contain rules of Snort.

```
root@n2e-virtual-machine: /etc/snort/rules
root@n2e-virtual-machine:/etc/snort/rules# ls
attack-responses.rules      community-smtp.rules        icmp.rules                  shellcode.rules
backdoor.rules              community-sql-injection.rules  imap.rules                  smtp.rules
bad-traffic.rules           community-virus.rules         info.rules                  snmp.rules
chat.rules                  community-web-attacks.rules    local.rules                 sql.rules
community-bot.rules          community-web-cgi.rules        misc.rules                  telnet.rules
community-deleted.rules      community-web-client.rules     multimedia.rules            tftp.rules
community-dos.rules          community-web-dos.rules        mysql.rules                 virus.rules
community-exploit.rules      community-web-iis.rules        netbios.rules               web-attacks.rules
community-ftp.rules           community-web-misc.rules        nntp.rules                  web-cgi.rules
community-game.rules          community-web-php.rules         oracle.rules                 web-client.rules
community-icmp.rules          ddos.rules                    other-ids.rules              web-coldfusion.rules
community-imap.rules          deleted.rules                  p2p.rules                   web-frontpage.rules
community-inappropriate.rules dns.rules                      policy.rules                 web-iis.rules
community-mail-client.rules   dos.rules                      pop2.rules                   web-misc.rules
community-misc.rules           experimental.rules             pop3.rules                   web-php.rules
community-nntp.rules           exploit.rules                   porn.rules                    x11.rules
community-oracle.rules         finger.rules                   rpc.rules                    rservices.rules
community-policy.rules         ftp.rules                      rservices.rules              scan.rules
community-sip.rules            icmp-info.rules                scan.rules
```

Writing your first Rule

Now, we will write a simple snort rule to alerts on ICMP messages (ping). the following is the rule:

```
alert icmp any any -> any any (msg:"ICMP Packet found"; sid:1000001; rev:1;)
```

Basically, this rule defines that an alert will be logged if an ICMP packet is found. The ICMP packet could be from any IP address and the rule ID is 1000001. Make sure to pick a SID greater 1000000 for your own rules.

- Put your rule in `/etc/snort/rules/local.rules` and comment all default rules (keep only local-rule file as shown the image bellow)

```
GNU nano 6.2                                snort.conf
#
# If you install the official VRT Sourcefire rules please review this
# configuration file and re-enable (remove the comment in the first line) those
# rules files that are available in your system (in the /etc/snort/rules
# directory)

# site specific rules
include $RULE_PATH/local.rules

# The include files commented below have been disabled
# because they are not available in the stock Debian
# rules. If you install the Sourcefire VRT please make
# sure you re-enable them again:

#include $RULE_PATH/app-detect.rules
#include $RULE_PATH/attack-responses.rules
#include $RULE_PATH/backdoor.rules
#include $RULE_PATH/bad-traffic.rules
#include $RULE_PATH/blacklist.rules
#include $RULE_PATH/botnet-cnc.rules
#include $RULE_PATH/browser-chrome.rules
#include $RULE_PATH/browser-firefox.rules
#include $RULE_PATH/browser-ie.rules
#include $RULE_PATH/browser-other.rules
#include $RULE_PATH/browser-plugins.rules
#include $RULE_PATH/browser-webkit.rules
#include $RULE_PATH/chat.rules
#include $RULE_PATH/content-replace.rules
#include $RULE_PATH/ddos.rules
#include $RULE_PATH/dns.rules
#include $RULE_PATH/dos.rules
#include $RULE_PATH/experimental.rules
#include $RULE_PATH/exploit-kit.rules
#include $RULE_PATH/exploit.rules
#include $RULE_PATH/file-executable.rules
#include $RULE_PATH/file-flash.rules
#include $RULE_PATH/file-identify.rules
#include $RULE_PATH/file-image.rules
#include $RULE_PATH/file-multimedia.rules
#include $RULE_PATH/file-office.rules
```

- Test your configuration with

```
sudo snort -T -c /etc/snort/snort.conf
```

The -T option is used to verify the configuration file. it should show success message:

```

--== Initialization Complete ==--

o"~)~
'-'~
'-'~

-*> Snort! <*-
Version 2.9.15.1 GRE (Build 15125)
By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
Copyright (C) 2014-2019 Cisco and/or its affiliates. All rights reserved
Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using libpcap version 1.10.1 (with TPACKET_V3)
Using PCRE version: 8.39 2016-06-14
Using ZLIB version: 1.2.11

Rules Engine: SF_SNORT_DETECTION_ENGINE Version 3.1 <Build 1>
Preprocessor Object: SF_SDF Version 1.1 <Build 1>
Preprocessor Object: SF_DNP3 Version 1.1 <Build 1>
Preprocessor Object: SF_IMAP Version 1.0 <Build 1>
Preprocessor Object: SF_MODBUS Version 1.1 <Build 1>
Preprocessor Object: SF_SIP Version 1.1 <Build 1>
Preprocessor Object: SF_REPUTATION Version 1.1 <Build 1>
Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>
Preprocessor Object: SF_SSH Version 1.1 <Build 3>
Preprocessor Object: SF_DCERPC2 Version 1.0 <Build 3>
Preprocessor Object: SF_GTP Version 1.1 <Build 1>
Preprocessor Object: SF_SMTP Version 1.1 <Build 9>
Preprocessor Object: SF_POP Version 1.0 <Build 1>
Preprocessor Object: SF_FTPTELNET Version 1.2 <Build 13>
Preprocessor Object: SF_DNS Version 1.1 <Build 4>
Preprocessor Object: appid Version 1.1 <Build 5>

Snort successfully validated the configuration!
Snort exiting

```

- Retsrat you snort service (use `status` command to check snort service after restarting)

```
sudo service snort restart
```

Triggering an Alert for the New Rule

To trigger an alert for the new rule, you only need to send an ICMP message to the VM image where snort runs. First, you need to find the IP address of the VM (`ipconfig` or `ip a`), then send a ping request (ICMP n 8 message) to the VM IP (`ping <VM_IP>`)

After you send the ping messages, the alerts should be triggered and you can find the log messages in `/var/log/snort/`.

```

root@n2e-virtual-machine:/var/log/snort# cat snort.alert.fast
10/20-09:12:50.761057  [**] [1:1000001:1] ICMP Packet found [**] [Priority: 0] {ICMP} 192.168.102.1 -> 192.168.102.128
10/20-09:12:50.761095  [**] [1:1000001:1] ICMP Packet found [**] [Priority: 0] {ICMP} 192.168.102.128 -> 192.168.102.1
10/20-09:12:51.776263  [**] [1:1000001:1] ICMP Packet found [**] [Priority: 0] {ICMP} 192.168.102.1 -> 192.168.102.128
10/20-09:12:51.776306  [**] [1:1000001:1] ICMP Packet found [**] [Priority: 0] {ICMP} 192.168.102.128 -> 192.168.102.1
10/20-09:12:52.786125  [**] [1:1000001:1] ICMP Packet found [**] [Priority: 0] {ICMP} 192.168.102.1 -> 192.168.102.128
10/20-09:12:52.786191  [**] [1:1000001:1] ICMP Packet found [**] [Priority: 0] {ICMP} 192.168.102.128 -> 192.168.102.1
10/20-09:12:53.797155  [**] [1:1000001:1] ICMP Packet found [**] [Priority: 0] {ICMP} 192.168.102.1 -> 192.168.102.128
10/20-09:12:53.797197  [**] [1:1000001:1] ICMP Packet found [**] [Priority: 0] {ICMP} 192.168.102.128 -> 192.168.102.1
root@n2e-virtual-machine:/var/log/snort#

```

Pactices

Try to writing snort rules to detect the following activities

1. HTTP traffic on port 80
2. An FTP connection to the server
3. SYN connection
4. Bad login (530) FTP attempt
5. TELNET connection
6. Email containing credit card information

- Consultate the snort documentation for details on rule options [snort manual](#)
- Locate pcaps in <https://github.com/AbdelliNasredine/IT-D> , lab2 and inspect the traffic using wireshark to extract signature

Offline mode

- Close this repo <https://github.com/AbdelliNasredine/IT-D>
- Go to lab2 folder and local `scanning.dump` file
- Run snort (offline mode) to analyze the dump file

```
sudo snort -r <pcapfile> -c /etc/snort/snort.conf -l ./snort-output
```

Take Home Exercise

Generate you own attack traffic and try to detect intrusion attempts using snort

- steps:
 - use tools for execution of attacks/intrusion attempts
 - scanning (`nmap`)
 - exploitation (`metasploit`)
 - capture network traffic: `sudo tcpdump -i <netif> -w /path/to/dumpfile.pcap`
 - run snort (offline mode) on you pcaps