CID Lab 4 - Intrusion Detection Systems Lab

Lab Goals:

1. To get you acquainted with different Intrusion Detection Systems (IDSs)
2. To install an IDS on a real machine
3. To use the IDS to detect Denial of Service Attacks
4. To create and study IDS rules
5. To study IDS logs and understand them

Please set up three SEED VMs in the same subnet as we did in other labs. Use these three VMs as Attacker, User, and Monitor. We will use the Monitor VM as the Victim for simplicity. Please check these three VMs can ping each other and list their IP in your report.

1. Installing Snort

Follow the below instructions to install Snort on your **Monitor/Victim** VM.

1.1 Installing from the source

Setting up Snort on Ubuntu from the source code consists of a set of steps: download the code, configure it, compile the code, install it to an appropriate directory, and lastly configure the detection rules.

Start by making a temporary download folder to your home directory and then changing into it with the command below.

```
mkdir ~/snort_src && cd ~/snort_src
```

Snort itself uses something called Data Acquisition library (DAQ) to make abstract calls to packet capture libraries. You can download the latest DAQ source package from the Snort website with the wget command underneath.

```
wget https://www.snort.org/downloads/snort/daq-2.0.7.tar.gz
```

The download will only take a few seconds. When complete, extract the source code and change into the new directory with the following commands.

```
tar -xvzf daq-2.0.7.tar.gz
cd daq-2.0.7
```

The latest version requires an additional step to auto-reconfigure DAQ before running the config. Use the command below which requires having autoconf and libtool installed.

```
autoreconf -f -i
```

Afterward, run the configuration script using its default values, then compile the program with make, and finally install DAQ.

```
./configure && make && sudo make install
```

With the DAQ installed you can get started with installing Snort. First, change back to the download folder.

```
cd ~/snort_src
```

Next, download the Snort source code with wget.

```
wget https://www.snort.org/downloads/snort/snort-2.9.16.tar.gz
```

Once the download is complete, extract the source and change into the new directory.

```
tar -xvzf snort-2.9.16.tar.gz
cd snort-2.9.16
```

Then configure the installation with Sourcefire enabled, run make and make install.

```
./configure --enable-sourcefire && make && sudo make install
```

With that done, continue to set up the configuration files.

1.2 Configuring Snort to run in the NIDS mode

Next, you will need to configure Snort for your system. This includes editing configuration files, downloading the rules that Snort will follow, and taking Snort for a test run.

Start with updating the shared libraries using the following command.

```
sudo ldconfig
```

Snort on Ubuntu gets installed to /usr/local/bin/snort directory. It is a good practice to create a symbolic link to /usr/sbin/snort.

```
sudo ln -s /usr/local/bin/snort /usr/sbin/snort
```

1.3 Setting up username and folder structure

To run Snort on Ubuntu safely without root access, you should create a new unprivileged user and a new user group to run the daemon.

```
sudo groupadd snort
sudo useradd snort -r -s /sbin/nologin -c SNORT_IDS -g snort
```

Then create the folder structure to house the Snort configuration. Just copy over the commands below.

```
sudo mkdir -p /etc/snort/rules
sudo mkdir /var/log/snort
sudo mkdir /usr/local/lib/snort_dynamicrules
```

Set the permissions for the new directories accordingly.

```
sudo chmod -R 5775 /etc/snort
sudo chmod -R 5775 /var/log/snort
sudo chmod -R 5775 /usr/local/lib/snort_dynamicrules
sudo chown -R snort:snort /etc/snort
sudo chown -R snort:snort /var/log/snort
sudo chown -R snort:snort /usr/local/lib/snort_dynamicrules
```

Now create new files for the whitelist and blacklist as well as local rules.

```
sudo touch /etc/snort/rules/white_list.rules
sudo touch /etc/snort/rules/black_list.rules
sudo touch /etc/snort/rules/local.rules
```

Then copy the configuration files from the download folder.

```
sudo cp ~/snort_src/snort-2.9.16/etc/*.conf* /etc/snort
sudo cp ~/snort_src/snort-2.9.16/etc/*.map /etc/snort
```

Next up, you will need to download the detection rules Snort will follow to identify potential threats. Snort provides three tiers of rule sets, community, registered and subscriber rules.

- Community rules are freely available although slightly limited.
- By registering for free on their website you get access to your Oink code, which lets you download the registered users rule sets.
- Lastly, subscriber rules are just available to users with an active subscription to Snort services.

Underneath you can find instructions for downloading community rules.

1.4 Using community rules

If you just want to quickly test out Snort, grab the community rules using wget with the command below.

```
wget https://www.snort.org/rules/community -O ~/community.tar.gz
```

Extract the rules and copy them to your configuration folder.

```
sudo tar -xvf ~/community.tar.gz -C ~/
sudo cp ~/community-rules/* /etc/snort/rules
```

By default, Snort on Ubuntu expects to find a number of different rule files which are not included in the community rules. You can easily comment out the unnecessary lines using the sed command below.

```
sudo sed -i 's/include $RULE_PATH/#include $RULE_PATH/' /etc/snort/snort.conf
```

1.5 Configuring the network and rule sets

With the configuration and rule files in place, edit the snort.conf to modify a few parameters. Open the configuration file in your favorite text editor, for example using nano with the command below.

```
sudo nano /etc/snort/snort.conf
```

Find these sections shown below in the configuration file and change the parameters to reflect the examples here. Replace the server_public_ip with the IP of the Monitor/Victim VM.

```
# Setup the network addresses you are protecting
ipvar HOME_NET server_public_ip/32

# Set up the external network addresses. Leave as "any" in most situations
ipvar EXTERNAL_NET !$HOME_NET

# Path to your rules files (this can be a relative path)
var RULE_PATH /etc/snort/rules
var SO_RULE_PATH /etc/snort/so_rules
var PREPROC_RULE_PATH /etc/snort/preproc_rules

# Set the absolute path appropriately
```

```
var WHITE_LIST_PATH /etc/snort/rules
var BLACK_LIST_PATH /etc/snort/rules
```

In the same snort.conf file, scroll down to section 6 and set the output for unified2 to log under the filename of snort.log like below.

```
# unified2
# Recommended for most installs
output unified2: filename snort.log, limit 128
```

Lastly, scroll down towards the bottom of the file to find the list of included rule sets. You will need to uncomment the local.rules to allow Snort to load any custom rules.

```
include $RULE_PATH/local.rules
```

If you are using the community rules, add the line below to your ruleset as well, for example just below your local.rules line.

```
include $RULE_PATH/community.rules
```

Once you are done with the configuration file, save the changes and exit the editor.

1.6 Validating settings

Your Snort should now be ready to run. Test the configuration using the parameter -T to enable test mode.

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

After running the Snort configuration test, you should get a message like this example below.

```
        --== Initialization Complete ==--

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.16 GRE (Build 118)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014-2020 Cisco and/or its affiliates. All rights rese
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using libpcap version 1.8.1
           Using PCRE version: 8.39 2016-06-14
           Using ZLIB version: 1.2.11

           Rules Engine: SF_SNORT_DETECTION_ENGINE  Version 3.1
           Preprocessor Object: SF_DCERPC2  Version 1.0
           Preprocessor Object: SF_SSH  Version 1.1
           Preprocessor Object: SF_FTPTELNET  Version 1.2
           Preprocessor Object: SF_SDF  Version 1.1
           Preprocessor Object: SF_DNP3  Version 1.1
           Preprocessor Object: SF_REPUTATION  Version 1.1
           Preprocessor Object: SF_IMAP  Version 1.0
           Preprocessor Object: SF_SMTP  Version 1.1
           Preprocessor Object: SF_GTP  Version 1.1
           Preprocessor Object: appid  Version 1.1
           Preprocessor Object: SF_MODBUS  Version 1.1
           Preprocessor Object: SF_POP  Version 1.0
           Preprocessor Object: SF_DNS  Version 1.1
           Preprocessor Object: SF_SSLPP  Version 1.1
           Preprocessor Object: SF_SIP  Version 1.1

Snort successfully validated the configuration!
```

In case you get an error, the printout should tell you what the problem is and where to fix it. Most likely problems are missing files or folders, which you can usually resolve by either adding any files you might have missed in the setup above, or by commenting out unnecessary inclusion lines in the snort.conf file. So check the configuration steps and try again.

1.7 Testing the configuration

To test if Snort is logging alerts as intended, add a custom detection rule alert on incoming ICMP connections to the local.rules file. Open your local rules in a text editor.

```
sudo nano /etc/snort/rules/local.rules
```

Then add the following line to the file.

```
alert icmp any any -> $HOME_NET any (msg:"ICMP test"; sid:10000001; rev:001;)
```

This rule consists of the following parts:

- action for traffic matching the rule, alert in this case
- traffic protocol like TCP, UDP, or ICMP here
- the source address and port, simply marked as any to include all addresses and ports
- the destination address and port, $HOME_NET as declared in the configuration and any for port
- some additional bits
    - log message
    - unique rule identifier (sid) which for local rules needs to be 1000001 or higher
    - rule version number.

Save and exit the editor.

Start Snort with -A console option to print the alerts to stdout. You will need to select the correct network interface with the public IP address of your server, for example, eth0.

```
sudo snort -A console -i eth0 -u snort -g snort -c /etc/snort/snort.conf
```

With Snort up and running, ping your cloud server from any other computer. You should see a notice for each ICMP call in the terminal running Snort.

```
07/12-11:20:33.501624  [**] [1:10000001:1] ICMP test [**] [Priority: 0] {ICMP}
83.136.252.119 -> 80.69.173.202
```

After the alerts show up you can stop Snort with ctrl+C.

2. Running an Internal Attack

This means that a local node in the network has been compromised, either because it was vulnerable, i.e., without the latest patches, with weak passwords, etc., or because a social engineering attack took place. We will be running such an attack and then write a custom rule to detect it with Snort IDS.

To start Snort, you need to open another terminal on the Monitor and enter the following command:

```
sudo snort -c /etc/snort/snort.conf -A fast -i eth1
```

Now, we need to simulate normal traffic on our network to which we can compare the DoS attack. In a real network, regular traffic such as email exchange, web browsing, etc., takes place at the same time as an attack. This makes it more difficult to distinguish the attack traffic from network traffic, and thus monitoring with accurate, robust, signature or behavior-based rules is important. To simulate traffic we will use iPerf. iPerf requires a Sever and a Client to generate

traffic, so we will use the Victim as the Server and the User as the Client to simulate traffic. To start the server on the Victim, execute the following command:

```
iperf -s
```

To start the traffic simulation, we need to make the iperf client start communicating with the Server. We will start a communication that will last two minutes (120 seconds) with the following command on the User:

```
iperf -c victim -t 120
```

Finally, it is time to run the attack! To run the attack, go to the terminal of the Attacker, and execute the following command:

```
sudo timeout -sHUP 20s hping3 -i ul -S --flood --rand-source -p 80 victim
```

3. Get more familiar with Snort commands

3.1 Based on the command you run above, you shall not be able to see alerts on the console prompt to run Snort. The actual alert log file will be stored under the /var/log/snort directory. However, if you try to read that log file, you will see some gibberish instead of some meaningful alert messages. Do some googling or read through online tutorials about Snort to see if you're able to figure this out.

3.2 Try to add a new option in the Snort command to make the stored alert log files being readable by a human being, a.k.a. ASCII mode.

3.3 Modify the Snort command so that the alert messages can be shown on the console prompt instead of being stored.

3.4 Take a look at the pre-installed Snort rules (/etc/snort/rules). Find out where the rule is that can detect SYN flooding attacks and explain your finding and this rule. Please test it to detect the attack above.

4. Create new rules

If you want to create new rules, these rules should be stored in /etc/snort/rules/local.rules. Once you update the rules, always run the command "sudo snort -c /etc/snort/snort.conf -T -i eth1" to

make sure there is no syntax error. Remember that you should always test how these rules work and include necessary screenshots.

4.1 Please create a custom rule that can detect SSH connections on the Monitor node.

4.2 Install "nmap" on the Attacker node. Try two different types of nmap scanning of the Victim node, and create customized Snort rules to detect such nmap scanning.

4.3 Are you able to create a Snort rule to detect the "failed login attempt of SSH connection"? Clearly explain why or why not