PRACTICAL SESSION FOR SNORT (NIDS)

Introduction

Credentials

IPFire – SSH - Id: root
IPFire – SSH - Password: .
(yes, just a dot)

Web UI - Id: admin
Web UI -Password: .

Kali Linux - Id: root
Kali Linux - Password: toor

Look here for an full listing of options and a bit about the header:
http://manual.snort.org/node27.html
Commands to be run in the command line will be in a different font, as well as bolded. Text to be added to a file will be in a different font.

Basic command

Restart snort
Through web UI on Kali box: Uncheck Snort -> Save -> Check Snort again -> Save
Through command line on IPFire: **`/etc/init.d/snort restart`**
Fully reconfigure IPFIre
Through command line on IPFire: **`setup`**
If Snort cannot be started check running issue
Through command line on IPFire: **`tail -f /var/log/messages`**
To follow the Snort alert log
Through command line on IPFire: **`tail -f /var/log/snort/alert`**

Default tutorial setup

Task 0: Load the VMs and view Snort's config file.
Note: When opening the VM, it'll prompt you, asking if you moved or copied the vm. Select the **I moved it** option, so set up is easier.

1. Open the IPFire vm first and log in with the credentials listed in the introduction.
2. Start up the Kali linux vm and log in with the credential listed in the introduction.
3. Run **`ifconfig`** in Kali must have eth1 (192.168.1.x for eth1) and eth0 (ip depend on VMware local setup)
4. Run **`ifconfig`** in IPFire must have red0 (ip depend on VMware local setup) and green0 (192.168.1.1 for green)
5. There is a change that Kali box IP id not correct. If the network is not functional, run **`ifdown eth1`** followed by **`ifup eth1`** from Kali box command line. Try to do this a couple time until **`eth1`** from Kali box is correct as describe in **Task 0.4**. If you accidentally pressed **I copied it,** you might have to reconfigure the IPFire (check basic command)
6. Snort should be installed in IPFire. Therefore, there should be a Snort config file, which you can view and edit on IPFire.
   `vi /etc/snort/snort.conf`

7. Look in the configuration file at line 74. Here you see the $HOME_NET and $EXTERNAL_NET variables mentioned in the presentation example. Ideally, you would change them, but for this tutorial, we won't bother.

Task 1: Create a rule to notify you of an incoming ICMP packet.

1. Create a rule file on IPFire
   ```
   vi /etc/snort/rules/rules.file
   ```
2. With a rule to notify you when you get an incoming response from an outgoing ping:
   ```
   alert icmp $EXTERNAL_NET any -> $HOME_NET any
   (msg:"Incoming ICMP packet"; sid:1000001;)
   ```
3. Include your new rules file within the configuration file. Open the config file, check line 74
4. Add to the bottom of the config file
   ```
   include $RULE_PATH/rules.file
   ```
5. Turn on Snort through WebUI on Kali box
   https://192.168.1.1:444/cgi-bin/index.cgi
   Service Intrusion Detection
   Red alert check Save
   https://192.168.1.1:444/cgi-bin/services.cgi to check Status
   If Snort cannot be started check
6. To follow the alert log from IPFire
   ```
   tail -f /var/log/snort/alert
   ```
7. To test your rule, use the Kali box to ping IPFire.
   ```
   ping <ip address of red0 interface of ipfire > -c 1
   ```

Task 2: Modify the rule above to notify you when the sequence number is 2.

1. Reopen the rule file
   ```
   vi /etc/snort/rules/rules.file
   ```
2. Add new rule or replace old rule with
   ```
   alert icmp $EXTERNAL_NET any -> $HOME_NET any
   (msg:"Incoming ICMP packet"; sid:1000001;icmp_seq:2)
   ```
3. Restart Snort so it is aware of the new rules.
4. Ping from Kali while looking at log.
   ```
   ping <ip address of red0 interface of ipfire> -c 1
   ```

Task 3: Make a dynamic rule that has state.

1. Add dynamic rules to the rule file.
   ```
   alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:
   "Receiving student number."; flowbits:set, studentnumber;
   sid:1000001; content:"student number")

   alert tpc $EXTERNAL_NET any -> $HOME_NET any (msg: "Already
   got the sutdent number. Now here is the name.";
   flowbits:isset, studentnumber; sid:1000002; content:"name")
   ```
2. Restart Snort so it is aware of the new rules.
3. Install netcat to IPFire through WEB ui on Kali box
   IPFire → Parfire → Available Addons: → Look for Netcat → + → →
   If there is no option press **Update**
4. Connect Kali with IPFire
   Objective: Run a server on IPFire and connect a Client from Kali

From IPFire connect to Kali box at port 3000 (Kali) through port 3001 (IPFire)

nc <ip address of eth0 interface of Kali> 3000 -p 3001

From Kali connect to IPFire box at port 3001 (IPFire) through port 3001 (Kali)

nc <ip address of red0 interface of ipfire> 30001 -p 3000

5. On IPFire send the process to background

   Suspend process

   Ctrl + z

   Send the process to the background

   bg

6. To follow the Snort alert log
7. From Kali send : **name**

   Sending a packet with content "name" without having sent a packet with content "student number" doesn't trigger any rules
8. From Kali send : **student number**
9. From Kali send : **name**
10. See in the snort log that an alert showed up for the student number packet and then the name packet.

Task 4: Update Snort rules with online Snort rules.

On kali linux through web ui

1. Service → Intrusion Detection → Select Snort rules update Community rules → Download new rules

   Oinkcode: **f034c8e610fcf4a9e2b31d7d946976ec1d69ad15**
2. Select all rules

   You can use this command in DevConsole of IceWeasel (press **F12** to open)

   $("input[type=checkbox]").attr('checked', true);

   Note: You can uncheck all rules with

   $("input[type=checkbox]").attr('checked', false); to

   Press **Update**
3. Restart snort
4. Check with nmap on Kali

   nmap <ip address of red0 interface of ipfire > -A

Task 5: Install Guardian and enable Guardian

1. On kali linux through IPFire's web UI

   IPFire → Parfire → Available Addons: → Look for Guardian → + → →

Task 6: Use Guardian to block an IP address.

1. See that you can ping your IPFire vm on Kali box

   ping <ip address of red0 interface of ipfire > -c 1
2. Mount your "attack" using nmap. Guardian will end up blocking this attack after a period of time.

   nmap <ip address of red0 interface of ipfire > -A
3. See that Guardian blocked the offending IP address.

   ping <ip address of red0 interface of ipfire > -c 1