Network Intrusion Detection System Using Snort

- Commomly called as "NIDS-Network Intrusion Detection system".
- A Network Intrusion Detection System (NIDS) is a security solution designed to monitor and analyze network traffic for signs of suspicious activity and potential threats.
- NIDS operates by inspecting incoming, outgoing, and internal traffic on a network to detect malicious actions, policy violations, or security breaches.
- When a threat is detected, NIDS can alert administrators, log the event, and sometimes take automated action to mitigate the threat.

- Common examples of NIDS include Snort and Suricata.

Steps to build an NIDS using Snort:

1. **Update System and Install Prerequisites**:

```
sudo apt-get update
```

```
sudo apt-get install -y build-essential libpcap-dev libpcre3-dev libdnet-dev bison flex zlib1g-dev
```

2. **Download and Extract Snort**:

```
wget https://www.snort.org/downloads/snort/snort-2.9.20.tar.gz
```

```
tar -xvzf snort-2.9.20.tar.gz
```

```
cd snort-2.9.20
```

3. **Compile and Install Snort**:

```
sudo apt-get install -y snort
```

4. **Verify Installation**:

```
snort -V
```

5. Configure Snort:(setting up rules)

Create Necessary Directories:

```
 sudo mkdir -p /etc/snort/rules                                   //Store Snort rule files
sudo mkdir /var/log/snort                                         //to store snort log files
sudo mkdir /usr/local/lib/snort_dynamicrules          // used to store dynamically loaded rule files
```

Create Configuration Files:

```
sudo touch /etc/snort/snort.conf
sudo touch /etc/snort/rules/blacklist.rules
sudo touch /etc/snort/rules/whitelist.rules
sudo touch /etc/snort/rules/local.rules     //it is for customized rules.you can specify your own rules in the configuration
```

Edit ‘`snort.conf’(if required`):

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

***lets see the ftp rules:**

```
 sudo gedit /etc/snort/rules/ftp.rules
```

Test the Configuration:

To check whether the snort configuration is valid or not.Enter the below command:

```
   sudo snort -T -c /etc/snort/snort.conf -i wlp2s0
```

Here ‘wlps20’ is a network interface of my computer.

You can check the available network interfaces on your computer by

ip link show

or

ipconfig

commands running in the terminal

Run Snort with the Interface:

-Start Snort, specifying the interface to monitor

```
sudo snort -A console -q -u snort -g snort -c /etc/snort/snort.conf -i wlp2s0
```

Testing and Verifying Alerts

- Once Snort is running and the rule is in place, perform network scans or traffic generation that matches your rules.
- Observe the console output for alerts or check the Snort logs typically located in `/var/log/snort/`.

Visualizing the detected attacks:

- print alerts directly to the console

Example output explanation:

```
05/29-07:20:53.058339 [**] [1:1917:6] SCAN UPnP service discover attempt
[**] [Classification: Detection of a Network Scan] [Priority: 3] {UDP}
192.168.220.63:40188 -> 239.255.255.250:1900
```

The alert indicates that a device with the IP address '`192.168.220.63`' sent a UDP packet from port `40188` to the multicast address '`239.255.255.250`' on port `1900`. This is typically associated with UPnP (Universal Plug and Play) service discovery attempts. UPnP is used by network devices to automatically discover and interact with each other on a network.

lets verify the trafic in the network:

```
sudo tcpdump -i wlp2s0
```

lets generate some traffic:

```
ping -c 4 8.8.8.8
```

lets run the snort again and see if it captures the generated traffic or not.

```
sudo snort -A console -q -u snort -g snort -c /etc/snort/snort.conf -i wlp2s0
```

As we can see that,it captured the traffic succesfully and showed as an alert by displaying it in the monitor.

THANKYOU...