**Mini-Project(cs-399)**

Under guidance of **Radhika MAm**

**Team members-**

Amit Prakhar Pandey(20bcs012)

Ashis J Kalathil(20bcs022)

Harsh Rawat(20bcs050)

Gaurav Agrawal(20bcs047)

**INTRUSION DETECTION SYSTEM**

**Set up an IDS with Filebeat Log Shipping**

**Intro:**

I don’t know about you, but I love analyzing network traffic. Especially when looking through the details for those little crumbs that lead to discovering a compromise by attackers. Building the skills needed for network log analysis takes time and practice. The best way to start is just put together a basic IDS(intrusion detection system) and start familiarizing yourself with the data you will encounter. After you feel comfortable, start running pcaps(Packet Captures) of known attacks so you get an idea of what they look like. This blog will show you how to set up that first IDS. Suricata will be used to perform rule-based packet inspection and alerts. Zeek will be included to provide the gritty details and key clues along the way. Finally, Filebeat will be used to ship the logs to the Elastic Stack. The scope of this blog is confined to setting up the IDS. I will provide links to a few tutorials on the Elastic Stack that will help you get you started if you are not familiar with it. Also, the end of this blog is only the beginning of the fun you can have with this project. Using Zeek, Suricata, and Elastic there are so many ways to expand the setup, literally, you could lose yourself for years.

**Requirements:**

● Ubuntu Server 22.04 was used when putting this together. Theoretically, the instructions can be adjusted for an earlier version of Ubuntu, or with heavy modifications, other Debian based Linux distributions. Feel free to ask for assistance if you follow this route and run into problems.

● Your server must have 2 ethernet adapters. Since I am using a virtual machine I set my second interface to use a host-only network.

● Whether built on a virtual machine(like mine) or on a physical box you will require 4GB of Ram and at least 60GB of hard drive space(you could use less, but it is risky. I used 32GB on a Raspberry Pi once).

● Please be sure to run an apt update and apt upgrade before starting so that your server is up to date. Always the best practice if you already don’t do so.

**Configuring the Sniffing Interface:**

An IDS always has at least 2 interfaces. The first one is called the management interface. The second interface is commonly known as the sniffing interface. Its job is to receive all packets sent across the network. To accomplish this the interface must be put in what is called Promiscuous mode. Usually, an interface will only receive packets that are addressed to it, but when configured in promiscuous mode the interface will receive all packets.

Ubuntu by default uses a network management system called Netplan. To my best knowledge, Netplan does not support the configuring of Promiscuous mode. To work around this we will install the older network management tools and remove NetPlan.

**Replace Netplan Instructions:**

**Enter the following Commands:**

ip addr (Note: write down the interface names, my VM used enp0s17 and enp0s8) sudo apt install -y ifupdown

**Create the Following File:**

sudo nano /etc/network/interfaces

**File text**( replace interface names with the ones you wrote down earlier if needed):

#loopback

auto lo

iface lo inet loopback

#management interface

allow-hotplug enp0s17

iface enp0s17 inet dhcp

#sniffing interface

allow-hotplug enp0s8

iface enp0s8 inet manual

up ifconfig enp0s8 promisc up

down ifconfig enp0s8 promisc down

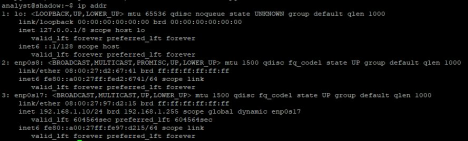
**Shut Down Network Services, remove netplan and reboot with the Following Commands:** sudo service systemd-networkd stop

sudo apt remove -y netplan

sudo apt install net-tools -y

sudo reboot now (Note: You could just restart the Interfaces, but I prefer to restart the whole thing.)

**Confirm that your interfaces Look like the image below with the following command:** ip addr



**Installing Suricata:**

Installing Suricata on Ubuntu is actually very simple. You do always have the option to build it from source, but with Ubuntu, there is also a repository that can be used to install a prepared binary of the latest release that we will use. For simplicity, I did not create a separate user for running Suricata. Other guides do because in production this is best practice, but for lab

purposes, I tend not to. The last command will install the emerging threats rule set.

**Enter the Following Commands:**

sudo apt install software-properties-common

sudo add-apt-repository ppa:oisf/suricata-stable

sudo apt update

sudo apt install -y suricata

sudo suricata-update

**Configuring Suricata:**

Suricata is fairly easy to configure as most of the default will be suitable for initial purposes. Using the command sudo nano /etc/suricata/suricata.yaml you can enter the configuration and make any necessary changes. The following set are the major things to accomplish.

● add any ip ranges to the home-net variable list that are not already covered ● In the section on logging outputs be sure that eve-log has a yes value for enabled ● Use the key combination ‘ctl \’ to bring up the find and replace option. Replace eth0 with the name of your sniffing interface( in my case I had enp0s8)

● Scroll down to checksum-validation under the stream section and set the value to no

**Create Suricata Service:**

To make it easier to run suricata we will create a service entry for it. Please enter the command, sudo nano /lib/systemd/system/suricata.service, once the blank file is open please copy the following text into it. Basically this file tells the system what commands and options to use when running suricata. This way we can run it in the background without specifying things like configuration paths each time.

[Unit]

Description=Suricata Intrusion Detection Service

After=syslog.target network-online.target

[Service]

# Environment file to pick up $OPTIONS. On Fedora/EL this would be

# /etc/sysconfig/suricata, or on Debian/Ubuntu, /etc/default/suricata.

#EnvironmentFile=-/etc/sysconfig/suricata

#EnvironmentFile=-/etc/default/suricata

ExecStartPre=/bin/rm -f /var/run/suricata.pid

ExecStart=/usr/bin/suricata -c /etc/suricata/suricata.yaml --pidfile /var/run/suricata.pid --af-packet ExecReload=/bin/kill -USR2 $MAINPID

[Install]

WantedBy=multi-user.target

**Installing Zeek:**

The second network monitoring tool we will install is called Zeek. For our purposes, Zeek is being used to provide a rich level of metadata of the monitored traffic. There are a variety of logs that Zeek will populate that will carry information about everything from connection records to application layer details. Zeek itself is highly extensible. Scripts can be written that perform special detections, preprocessing, or even an entirely new protocol analyzer. As with Suricata, I have skipped the step of creating a dedicated Zeek user to simplify installation, though in some settings you would want to do this.

**Enter the Following Commands:**

sudo apt install git -y (skip if git is already installed, it is not installed by default on Ubuntu) sudo apt install -y cmake make gcc g++ flex bison libpcap-dev libssl-dev python-dev swig zlib1g-dev sudo git clone --recursive https://github.com/zeek/zeek

cd zeek

./configure

make

sudo make install

**Add Zeek to System Path:**

We want to be able to run Zeek without navigating to its directory. To do this we will add it to the system’s path variable. Basically this is the list of approved directories that the system will search when a program is called. On most occasions, we would add the new path to the file /etc/environment so it applies to all users. However, on Ubuntu at least, we need to add it to the secured path list(https://help.ubuntu.com/community/EnvironmentVariables#sudo\_caveat) in our current setup.

Enter the command, sudo visudo to enter the file

Once the file is open please add the following line to the secured paths list: :/usr/local/zeek/bin

**Configuring Zeek:**

Zeek has three separate files that contain the configuration information. The first one is the configuration of what networks Zeek will monitor. Here you can specify what IP ranges you want Zeek to cover. By default most private IP ranges are covered, but you can add any that might apply to your environment. Enter, sudo nano /usr/local/zeek/etc/networks.cfg, to edit the file.

Next enter the command, sudo nano /usr/local/zeek/etc/node.cfg, to modify the setup of the Zeek node itself. For our purposes leave the default values except the interface name, which you should change to reflect the name of your sniffing interface(in my case it is enp0s8). Notice that it is in this file that a cluster of Zeek nodes can be configured to work together.

We will then enter the command, sudo nano /usr/local/zeek/etc/zeekctl.cfg, to configure the rest of the zeek management settings. Please make the following changes as you move through the configuration file.

● Make sure all mail settings are set to 0 as we will not be utilizing these features ● Change the log rotation values to a low setting. Since our nodes have a small amount of space I always set my LogExpireInterval to 48hrs.

● Same with Stats log Interval, set it something small like 1 or 2 days

● Finally set the log path for Zeek to /var/log/zeek/logs

**Change Output to JSON:**

We need to change the output format of the logs to json

sudo nano /usr/local/zeek/share/zeek/site/local.zeek

Enter the lines:

# Output to JSON

@load policy/tuning/json-logs.zeek

**Create ZeekCtl Cron Job:**

ZeekCtl requires an addition to the crontab to setup its regular housekeeping activities like log rotation. Please enter the command, crontab -e, then add the following line to the bottom of the file: \*/5 \* \* \* \* /usr/local/zeek/bin/zeekctl cron

echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee

/etc/apt/sources.list.d/elastic-7.x.list

**Installing Filebeat:**

The best way to install Elastic software is to add their repository to your server. This way you can install any component you wish, even those you might not have originally planned.

wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add - sudo apt-get install apt-transport-https

(this command is one line) echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee /etc/apt/sources.list.d/elastic-7.x.list

sudo apt update

sudo apt install filebeat

**Creating Filebeat Keystore:**

For my Elastic work I use a cluster that is hosted by Elastic Cloud. The cluster is protected by the common username:password requirement that is also now available on the free tier of locally hosted clusters. Best practice when entering these values in the configuration of Filebeat is to use a keystore. I also prefer to store the elastic host information here too because of the fact that my cluster is hosted online. It also makes it easy to change the value later on. Since I am using Elastic Cloud I could actually just use my Cloud ID instead of providing the host information, and I should add you could also replace user:password with an API code. The following instructions will demonstrate how to create one using the simplest setup. **If you do not require a username:password to access your Elastic Stack then you may skip this section.**

sudo filebeat keystore create

sudo filebeat keystore add es\_host (When prompted enter the value for your host) sudo filebeat keystore add kibana\_host (When prompted enter the value for your kibana host) sudo filebeat keystore add es\_user (When prompted enter the user name)

sudo filebeat keystore add es\_pwd (When prompted enter the password)

**Configure Filebeat:**

To configure filebeat you must replace default values with those we created in the keystore. The format for using the values is the same as environment variables on a Unix based operating system. You will need to scroll through the configuration file and replace the following default values.

sudo nano /etc/filebeat/filebeat.yml

● Uncomment and replace the kibana host value with ${kibana\_host}

● If you did not create a keystore you can just type in the host and port being used.(default value is localhost:5601)

● If you are using a Kibana Space be sure to uncomment that line and add the ID of the space

● Uncomment and replace the Elasticsearch host value with ${es\_host} ● If you did not create a keystore you can just type in the host and port being used(default value is localhost:9200)

● Uncomment and replace the username value with ${es\_user}

● Uncomment and replace the password value with ${es\_pwd}

**Activate and Configure Modules:**

We will be using the built in modules to process the logs. The modules have pre-defined ingest pipelines that will translate the logs into Elastic Common Schema.

sudo filebeat modules enable suricata zeek

sudo nano /etc/filebeat/modules.d/zeek.yml

● Uncomment var paths and please enter: /var/log/zeek/logs/current/\*.log

**Turn it all on and check if it is running:**

sudo service suricata start

sudo zeekctl deploy

sudo service filebeat start

sudo service suricata status

sudo zeekctl status

sudo service filebeat status (If filebeat is not running the most common problem is a spacing or syntax mistake in the yaml file)

**Send some logs to Elasticsearch:**

One very useful tool to test an IDS is tcp replay. It allows you to pass a recorded packet capture through your sniffing interface. Let’s use one of the provided pcaps to test it out. You should see a lot of packet traffic whizzing across your screen. When it is done go to Kibana and enjoy exploring the data. I recommend the SIEM application as a good starting point.

sudo apt install tcpreplay -y

wget https://s3.amazonaws.com/tcpreplay-pcap-files/bigFlows.pcap

sudo tcpreplay -t -v -i enp0s8 bigFlows.pcap(my sniffer is enp0s8, replace with your interface name)

**Conclusion:**

Congratulations on setting up your Intrusion Detection System. The logs provided by Zeek and Suricata will make it possible for you to finally have true visibility of what is going over a network. It is critical to understand the nuances of network traffic in order to find adversaries that are attempting to attack your network. This guide is truly only a beginning though. There are so many avenues you might take this knowledge. To help you take your next steps I am providing a reference section below with some of the best places to learn more and decide where your network monitoring journey will go next.

**How to use Zeek for Pcap analysis-**

echo 'deb http://download.opensuse.org/repositories/security:/zeek/xUbuntu\_22.04/ /' | sudo tee /etc/apt/sources.list.d/security:zeek.list

curl -fsSL https://download.opensuse.org/repositories/security:zeek/xUbuntu\_22.04/Release.key | gpg --dearmor | sudo tee /etc/apt/trusted.gpg.d/security\_zeek.gpg > /dev/null

sudo apt update

sudo apt install zeek

**Then after completion run**

cd/opt/zeek/bin/

ls

**Then download the file for pcap analysis from**

**malware-traffic-analysis.net/….**

**Change the location to your file path like**

cd-/downloads/

unzip

ls

**Change the folder**

cd/tmp/

mkdir zeek

cd zeek/

ls

/opt/zeek/bin/zeek -r -/Downloads/2020-10-12-Lokiboat-infection-traffic.pcap

ls

less -S conn.log

less -S dns.log

less -S http.log

For getting a unique list

cat dns.log | /opt/zeek/bin/zeek-cut query |sort | uniq -c

less -S pe.log

grep (copy a uid from above file and paste it here) \*.log

less -S conn.log

grep (copy a uid from above file and paste it here) \*

ls

Now after this whole process we are able to understand how we can analyze the network traffic through pcap files .

**References:**

Introduction to the Elastic Stack https://www.youtube.com/watch?v=GvzosoaOaIQ&t=4s Hunters Forge: Project Mordor: https://github.com/hunters-forge/mordor

Security Onion: https://securityonion.net/

RockNSM: <https://rocknsm.io/>

[Zeek in Action, Video 1, Suspected Malware Compromise](https://youtu.be/xpPEHtACrek)

[Zeek in Action: Introduction and How to Set Up a Windows Workstation Using Brim Security.](https://youtu.be/nsFDquaanZ8)

<https://www.google.com/url?q=https://www.malware-traffic-analysis.net/&hl=en&sa=D&source=meet>

<https://talkgadget.google.com/linkredirect?dest=https%3A%2F%2Fwww.youtube.com%2Fwatch%3Fv%3DtC53oreKfZo>

[How to install Zeek (aka bro) - Virtually Testing](https://youtube.com/watch?v=4b_dW5JdE5U&si=EnSIkaIECMiOmarE)

[12 Days of Defense - Day 2: How to use Zeek for PCAP Analysis](https://www.youtube.com/watch?v=bznH1yMyjjo&ab_channel=JohnHubbard)