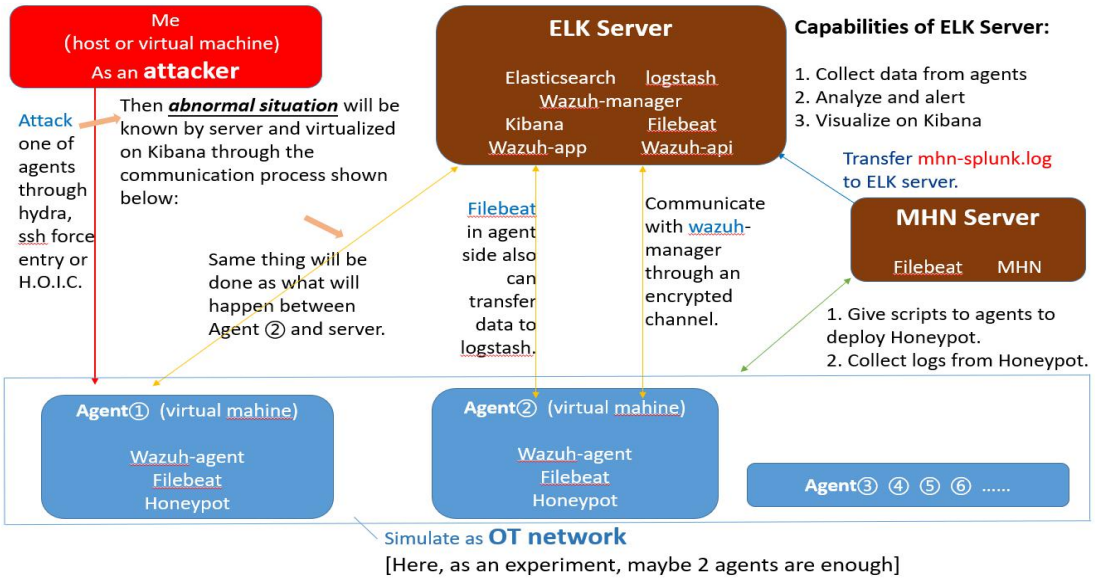
OT Honeynet

All those are installed on Ubuntu 18.04.

The design is shown below:



In 'MHN-server' virtual machine, MHN and Filebeat should be deployed. ELK and Wazuh server should be installed in 'ELK-server' virtual machine. All agents should equip with Filebeat, Wazuh-agent and Honeypot.

MHN:

Install:

```
cd /opt
git clone https://github.com/threatstream/mhn.git
cd mhn/
sudo ./install.sh [Until "Successfully installed MHN" appears]
```

Start all and check status:

```
sudo supervisorctl start all        sudo supervisorctl status
```

If *'mhn-celery-worker' FATAL*, run 'chmod 777 -R /var/log/mhn/mhn.log'.

If *'honeymap' FATAL*, delete the old go package and redownload. And run:

```
cd /opt/honeymap/server
export GOPATH=/opt/honeymap/server
go get github.com/golang/net
mkdir -p golang.org/x
cp -rf src/github.com/golang/net/ ./golang.org/x/
cp -rf golang.org/ /usr/local/go/src/
go build
sudo supervisorctl restart all
```

The successful UI should like below, and then access *http://127.0.0.1*

```
root@ubuntu:/home/student# supervisorctl status all
geoloc                           RUNNING   pid 19475, uptime 0:00:27
honeymap                         RUNNING   pid 19479, uptime 0:00:27
hpfeeds-broker                   RUNNING   pid 19473, uptime 0:00:27
hpfeeds-logger-splunk            RUNNING   pid 19472, uptime 0:00:27
mhn-celery-beat                  RUNNING   pid 19471, uptime 0:00:27
mhn-celery-worker                RUNNING   pid 19477, uptime 0:00:27
mhn-collector                    RUNNING   pid 19478, uptime 0:00:27
mhn-uwsgi                        RUNNING   pid 19476, uptime 0:00:27
mnemosyne                        RUNNING   pid 19474, uptime 0:00:27
```

If run 'sudo supervisorctl start all' encounter error likes '*Unit all.service not found*', then run:

```
student# sudo chmod 777 /run
student# sudo chmod 777 /var/log
student# sudo touch /var/run/supervisor.sock
student# sudo chmod 777 /var/run/supervisor.sock
```

Honeypot:

Using scripts shown in your MHN Deployment sector and change the IP address in the given script to your MHN-server's IP.

Run 'sudo supervisorctl start all'

Other methods to install Conpot:

Via a pre-built image:

docker pull honeynet/conpot

docker run -it -p 80:80 -p 102:102 -p 502:502 -p 161:161/udp --network=bridge honeynet/conpot:latest /bin/sh

Build docker image from source:

git clone https://github.com/mushorg/conpot.git

sudo make run-docker

Directly install:

sudo git clone https://github.com/mushorg/conpot.git

cd conpot

sudo python setup.py

Build from source and run with docker-compose:

git clone https://github.com/mushorg/conpot.git

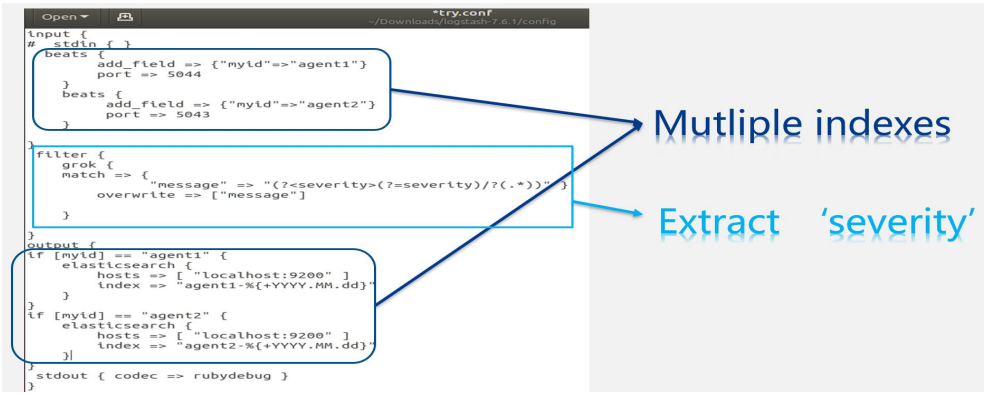
cd conpot/docker

sudo docker-compose build sudo docker-compose up

Filebeat and ELK Stack:

Download from official website and unzipe.

Config filebeat.yml and a Logstash config file like below:



Change the IP address in elasticsearch.yml to your ELK-server's IP.

Start them:

sudo systemctl start elasticsearch

sudo systemctl start Kibana

./filebeat -e -c ./filebeat.yml

bin/logstash -f config.conf --config.reload.automatic

If *Elasticsearch cannot be started*, there are three situations:

Memory is limited: Increasing the memory of the virtual machine to

4G.

Time out: Repeatly run ‘sudo deamon-reload’ and ‘sudo supervisorctl restart elasticsearch’.

Status is ‘active’ but no response: Reinstall.

During running ELK may meet ‘*Java memory error when building with Ubuntu*’

Change content in /etc/elasticsearch/jvm.options to ‘-Xms512m’ and ‘-Xmx512m’.

There two situations when ‘*Kibana server is not ready yet*’ appears:

Need long time to prepare: Just wait.

"port 5601 is already in use. Another instance of Kibana may be running."

Run ‘ps -a | grep apt’ and then kill all process.

When meet ‘*Could not find logstash.yml*’ but the path is right, can run like below:

bin/logstash --path.settings /etc/logstash/conf.d/ -f conf.d/try.conf

If the same file has been passed to Logstash, then *‘data/repository’* should be delete at the Filebeat side.

If error like: ‘Logstash could not be started because there is already another instance using the configured data directory.’

cd /usr/share/logstash/ ls -lah

And then run ‘rm -rf .lock’ and restart.

Do not change ‘*gemspec*’ in usr/share/logstash.

Visualize on *http:127.0.0.1:5601*

Wazuh Server:

(related codes are in https://documentation.wazuh.com/3.11/installation-guide/index.html)

apt-get install python gcc make libc6-dev curl policycoreutils automake

autoconf libtool

curl -Ls https://github.com/wazuh/wazuh/archive/v3.11.4.tar.gz | tar zx

cd wazuh-* ./install.sh entry ‘manger’ during deployment.

systemctl start wazuh-manager

curl -sL https://deb.nodesource.com/setup_10.x | sudo -E bash -

apt-get install -y nodejs npm config set user 0

curl -s -o install_api.sh https://raw.githubusercontent.com/wazuh/wazuh-api/v3.11.4/install_api.sh && bash ./install_api.sh download

systemctl status wazuh-api

cd /usr/share/kibana/

sudo -u kibana bin/kibana-plugin install https://packages.wazuh.com/wazuhapp/wazuhapp-3.11.4_7.6.1.zip

A strange situation may encounter is that *Wazuh-app cannot be opened in Kibana*, but status of Wazuh-app, Wazuh-api and Wazuh-manager is active.

Then try to change another browser.

Wazuh-agent:

curl -Ls https://github.com/wazuh/wazuh/archive/v3.11.4.tar.gz | tar zx

cd wazuh-* ./install.sh entry ‘agent’ during deployment.

Register Wazuh agent：

Run ‘/var/ossec/bin/manage_agents’ at Wazuh-manager side, and add the IP of agent virtual machine. Then get the key.

Run ‘/var/ossec/bin/manage_agents’ at Wazuh-agent side, and paste the key got from manager.

Check agent status through Kibana.