\*\*\*After I connected to my jumpBox through

ssh azureuser@40.112.169.59

Then, I started my container and attached it to ansible

From cd /etc/ansible

nano hosts

[webservers]

## alpha.example.org

## beta.example.org

## 192.168.1.100

## 192.168.1.110

10.0.0.5 ansible\_python\_interpreter=/usr/bin/python3

10.0.0.6 ansible\_python\_interpreter=/usr/bin/python3

10.0.0.8 ansible\_python\_interpreter=/usr/bin/python3

[elk]

10.1.0.4 ansible\_python\_interpreter=/usr/bin/python3

\*\*\*\*After that I

cd ~

Where I was able to

nano install.yml

GNU nano 2.9.3 install.yml

---

- name: Install Web VMs

hosts: webservers

become: true

tasks:

- name: Install docker

apt:

update\_cache: yes

name: docker.io

state: present

- name: Install pip3

apt:

name: python3-pip

state: present

- name: Install Python docker

pip:

name: docker

state: present

- name: Install dvwa

docker\_container:

name: dvwa

image: cyberxsecurity/dvwa

state: started

restart\_policy: always

published\_ports: 80:80

Then I

nano ansible.cfg

# Controls if a missing handler for a notification event is an error or a warning

#error\_on\_missing\_handler = True

# change this for alternative sudo implementations

#sudo\_exe = sudo

# What flags to pass to sudo

# WARNING: leaving out the defaults might create unexpected behaviours

#sudo\_flags = -H -S -n

# SSH timeout

#timeout = 10

# default user to use for playbooks if user is not specified

# (/usr/bin/ansible will use current user as default)

remote\_user = azureuser

THESE ARE MY SCRIPTS

1. TO CONFIGURE THE WEB VM WITH DOCKER:

---

- name: Config Web VM with Docker

hosts: webservers

become: true

tasks:

- name: [docker.io](http://docker.io/)

apt:

update\_cache: yes

name: [docker.io](http://docker.io/)

state: present

- name: Install pip3

apt:

name: python3-pip

state: present

- name: Install Docker python module

pip:

name: docker

state: present

- name: download and launch a docker web container

docker\_container:

name: dvwa

image: cyberxsecurity/dvwa

state: started

restart\_policy: always

published\_ports: 80:80

- name: Enable docker service

systemd:

name: docker

enabled: yes

TO CONFIGURE THE WEB-ELK WITH DOCKER

---

- name: Configure Elk VM with Docker

hosts: elk

remote\_user: azureuser

become: true

tasks:

# Use apt module

- name: Install [docker.io](http://docker.io/)

apt:

update\_cache: yes

force\_apt\_get: yes

name: docker

state: present

# Use apt module

- name: Install python3-pip

apt:

force\_apt\_get: yes

name: python3

state: present

# Use pip module (It will default to pip3)

- name: Install Docker module

pip:

name: docker

state: present

# Use command module

- name: Increase virtual memory

command: sysctl -w vm.max\_map\_count=262144

# Use sysctl module

- name: Use more memory

sysctl:

name: memory

value: '4'

state: present

reload: no

# Use docker\_container module

- name: download and launch a docker elk container

docker\_container:

name: elk

image: sebp/elk:761

state: started

restart\_policy: always

# Please list the ports that ELK runs on

published\_ports:

- 5601:5601

- 9200:9200

- 5044:5044

# Use systemd module

- name: Enable service docker on boot

systemd:

name: docker

enabled: yes

WE RAN OUR ANSIBLE PLAYBOOK; WHICH MADE UP INVENTORY AND MADE DEPLOYMENT

-bash: cd: /etc/ansible: No such file or directory

azureuser@Jump-Box:~$ sudo docker attach naughty\_wilbur

root@9df70ec7cffb:~# cd /etc/ansible

root@9df70ec7cffb:/etc/ansible# ansible-playbook install-elk.yml

PLAY [Configure Elk VM with Docker] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

TASK [Gathering Facts] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [10.1.0.4]

TASK [Enable service docker on boot] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [10.1.0.4]

PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

10.1.0.4 : ok=8 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

TO INSTALL MY FILEBEAT THAT WILL MONITOR THE LOG FILES, COLLECT LOG EVENTS AND FORWARD THEM TO ELASTIC SEARCH OR LOGSTASH

---

- name: installing and launching filebeat

hosts: webservers

become: yes

tasks:

- name: download filebeat deb

command: curl -L -O <https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-7.4.0-amd64.deb>

- name: install filebeat deb

command: dpkg -i filebeat-7.4.0-amd64.deb

- name: drop in filebeat.yml

copy:

src: /etc/ansible/files/filebeat-config.yml

dest: /etc/filebeat/filebeat.yml

- name: enable and configure system module

command: filebeat modules enable system

- name: setup filebeat

command: filebeat setup

- name: start filebeat service

command: service filebeat start

- name: enable service filebeat on boot

systemd:

name: filebeat

enabled: yes

AFTER THIS WE INSTALLED METRICBEAT TO COLLECT METRICS FROM O.S, AND ALSO MONITOR OTHER BEATS AND ELK STACK ITSELF

---

- name: Install metric beat

hosts: webservers

become: true

tasks:

# Use command module

- name: Download metricbeat

command: curl -L -O <https://artifacts.elastic.co/downloads/beats/metricbeat/metricbeat-7.4.0-amd64.deb>

# Use command module

- name: install metricbeat

command: dpkg -i metricbeat-7.4.0-amd64.deb

# Use copy module

- name: drop in metricbeat config

copy:

src: /etc/ansible/files/metricbeat-config.yml

dest: /etc/metricbeat/metricbeat.yml

# Use command module

- name: enable and configure docker module for metric beat

command: metricbeat modules enable docker

# Use command module

- name: setup metric beat

command: metricbeat setup

# Use command module

- name: start metric beat

command: metricbeat -e

# Use systemd module

- name: enable service metricbeat on boot

systemd:

name: metricbeat

enabled: yes