Jeff King

cyber Security Project 1

The main purpose of the network is to expose a load-balanced and monitored instance of DVWA, the Damn Vulnerable Web Application.

Load balancing ensures that the application will be highly available, in addition to restricting access to the network. Load balancers help ensure environment availability through distribution of incoming data to web servers. Jump boxes allow for more easy administration of multiple systems and provide an additional layer between the outside and internal assets.

Integrating an ELK server allows users to easily monitor the vulnerable VMs for changes to the event logs & system metrics.

File beats watch for log directories or specific log files.

Jumpbox Gateway 10.0.0.4 Linux Ubuntu (All)

|  |
| --- |
| Web 1 Server 10.0.0.5 |
| Web 2 Server 10.0.0,6 |
| Web 3 DVWA 10.0.0.7 |
| ELK Kibana 10.3.0.4 |

**Access Policies**

Machines on this internal network will not be exposed to the public Internet.

Only the jump box provisioner machine can accept connections from the Internet. Access to this machine is only allowed from the following IP addresses:

-Personal IP address 73.223…………….etc

Machines within the network can only be accessed via the Jump Box. Elk machine can have access from personal IP address through port 5601.

Jump is Public Access via Personal IP SSH

Load Balancer is open

|  |
| --- |
| Web 1 is not Public Accessible 10.0.0.5 |
| Web 2 is not Public Accessible 10.0.0.6 |
| Web 3 is not Public Accessible 10.0.0.7 |
| ELK VM is Public Access via Personal IP SSH |

**ELK Configuration**

Ansible was used to automate configuration of the ELK machine. No configuration was performed manually, which is advantageous because...

* The main advantage of automating the installation process is that we could deploy multiple servers easily and quickly without having to physically touch each server.

The playbook implements the following tasks:

1. Published Ports are set
2. Download & Configure ELK docker Container
3. Increase VM Memory
4. Install Docker.io & pip3
5. OS & Software Updates

**Target Machines & Beats**

This ELK server is configured to monitor the following machines:

* 10.0.0.5,10.0.0.6, 10.0.0.7

We have installed the following Beats on these machines:

* Filebeat and Metricbeat

These Beats allow us to collect the following information from each machine:

* Filebeat collects the logs for each virtual machine, you should be able to see information like how many visitors you've had and where they are located. Along with seeing if any users experienced any errors like 404 or 503 and when they happened.
* Metricbeat provides metric logs for each virtual machine, you will be able to see information like CPU Usage and Memory Usage.

**Using the Playbook**

In order to use the playbook, you will need to have an Ansible control node already configured. Assuming you have such a control node provisioned:

SSH into the control node and follow the steps below:

Copy the filebeat-playbook.yml and metricbeat-playbook.yml files to your ansible directory (~/etc/ansible).

Update the hosts file to include your private IP addresses for the machines you want for each service. DVWA machines are under webservers while the Elk stack is under Elk. In turn you can then update each playbook file with the host to tell ansible which machines to run the playbook on.

Run the playbooks, and navigate to the Public IP address of the Elk-VM on port 5601 to check that the installation worked as expected.

Install Docker, Python, Python Module, Download DVWA Container, Docker Service

The playbook is filebeat-config.yml and curl command is needed to get the filebeat-config to forward it into the correct directory on the Jump-Box Provisioner.

Copy playbook file to Ansible Docker container.

Use curl

Update Ansible hosts file via /etc/ansible/hosts

[webservers]

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

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

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

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

[elkservers]

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

Update Ansible Config file /etc/ansible/ansible.cfg

Ssh with Jumpbox ssh [azureuserJK@20.83.250.128](mailto:azureuserJK@20.83.250.128)

Start Ansible docker container sudo docker start strange knuth

Attach shell to Ansible docker container sudo docker attach strange knuth

Run playbook:

Ansible-playbook /etc/ansible/pentest.yml

Ansible-playbook /etc/ansible/install-elk.yml

Ansible-playbook /etc/ansible/roles/filebeat-playbook.yml

Navigate to Kibana http://20.119.38.232:5601/app/kibana

**Ansible Build**

nano ansible.cfg

nano metricbeat-config.yml

nano filebeat-config.yml

Use the nano command to update your hosts file to your private IP address of your machines

nano hosts

Run the pentest playbook to get your docker containers running for web 1/2 VMs

ansible-playbook pentest.yml

Run the metricbeat and filebeat playbooks to get both services running on your Elk-VM

ansible-playbook metricbeat-playbook.yml && ansible-playbook filebeat-playbook.yml

**Pentest.yml**

---

- name: Config Web VM with Docker

hosts: webservers

become: true

tasks:

- name: docker.io

apt:

force\_apt\_get: yes

update\_cache: yes

name: docker.io

state: present

- name: Install pip3

apt:

force\_apt\_get: yes

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

published\_ports: 80:80

- name: Enable docker service

systemd:

name: docker

enabled: yes

Steps to set up ELK-server

1. Open Git Bash command prompt
2. ssh username@Jump-BoxJames'IP
3. sudo docker container list -a
4. sudo docker start dreamy\_satoshi && sudo docker attach dreamy\_satoshi dreamy\_satoshi
5. nano /etc/ansible/hosts (add the internal private IP from the Elk-Server VM machine) [elk] 10.3.0.4 ansible\_python\_interpreter=/usr/bin/python3
6. nano /etc/ansible/ansible.cfg (update remote user to the user name that you created for the public key for Microsoft Azure based on the Elk-Server VM machine)
7. nano /etc/ansible/elk-install.yml (create the YML file for the ELK installation)
8. install-elk.yml

ansible all -m ping (to ensure configurations and connection to the ELK-Server VM is successful)

1. ansible-playbook /etc/ansible/install-elk.yml (Deploy the elk installation to the internal private VM machines)
2. Type http://[PuplicIPfortheElkServer]:5601/app/kibana (Welcome to Kibana page should appear on the browser, that is successful setup)

---

- name: Configure Elk VM with Docker

hosts: elkservers

remote\_user: elk

become: true

tasks:

# Use apt module

- name: Install docker.io

apt:

update\_cache: yes

force\_apt\_get: yes

name: docker.io

state: present

# Use apt module

- name: Install python3-pip

apt:

force\_apt\_get: yes

name: python3-pip

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: vm.max\_map\_count

value: 262144

state: present

reload: yes

# 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

**filebeat-playbook.yml**

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES 2e98ef1c6eab cyberxsecurity/ansible:latest "/bin/sh -c /bin/bas…" 47 hours ago Exited (137) 10 hours ago dreamy\_satoshi d354a79394f6 cyberxsecurity/ansible:latest "/bin/sh -c /bin/bas…" 47 hours ago Exited (137) 44 hours ago distracted\_saha

sudo docker start dreamy\_satoshi && sudo docker attach dreamy\_satoshi dreamy\_satoshi

---

- 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

Text

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

Project 1

ELK Stack Deployment

A screenshot of a computer screen

Description automatically generated with medium confidence