**Automated ELK Stack Deployment**

The files in this repository were used to configure the network depicted below.

**Note**: The following image link needs to be updated. Replace diagram\_filename.png with the name of your diagram image file.

These files have been tested and used to generate a live ELK deployment on Azure. They can be used to either recreate the entire deployment pictured above. Alternatively, select portions of the **Deployment-playbook.yml** file may be used to install only certain pieces of it, such as Filebeat.

* *TODO: Enter the playbook file.*
* **https://github.com/gdollasign78/Vader/blob/main/Ansible/elk-playbook.yml**

This document contains the following details:

* Description of the Topologu
* Access Policies
* ELK Configuration
  + Beats in Use
  + Machines Being Monitored
* How to Use the Ansible Build

**Description of the Topology**

The main purpose of this network is to expose a load-balanced and monitored instance of DVWA, the D\*mn Vulnerable Web Application.

Load balancing ensures that the application will be highly **redundant**, in addition to restricting **access** to the network.

* *TODO: What aspect of security do load balancers protect? What is the advantage of a jump box?*
* **The Load Balancers allow access to the regularly not available Web-VM's that are loaded with the docker containers for the DVWA, Filebeat, and Metricbeat. Placing them behind the load balancer allows the load to be spread out across multiple machines in case a machine goes down.**
* **The jump Box limits access to the rest of the network as you have to ssh to the jump box, then move into the docker container, then move to any of the other machines in the network. This allows a single point of entry which makes it easier to monitor. It also allows more robust security procedures to be deployed to the jump box in order to make the entire network more secure.**

Integrating an ELK server allows users to easily monitor the vulnerable VMs for changes to the **network** and system **files**.

* *TODO: What does Filebeat watch for?*
* **Filebeat is used to monitor log data, events, and files, then forwards them to either Elasticsearch or Logstash.**
* *TODO: What does Metricbeat record?*
* **Metricbeat is used to collect different metrics from the OS and services running on the server.**

The configuration details of each machine may be found below. *Note: Use the* [*Markdown Table Generator*](http://www.tablesgenerator.com/markdown_tables) *to add/remove values from the table*.

| **Name** | **Function** | **IP Address** | **Operating System** |
| --- | --- | --- | --- |
| **Jump Box** | **Gateway** | **10.0.0.4** | **Linux** |
| **Web-1** | **Webserver** | **10.0.0.5** | **Linux** |
| **Web-2** | **Webserver** | **10.0.0.6** | **Linux** |
| **Web-3** | **Webserver** | **10.0.0.7** | **Linux** |

**Access Policies**

The machines on the internal network are not exposed to the public Internet.

Only the **Jump Box** machine can accept connections from the Internet. Access to this machine is only allowed from the following IP addresses:

* *TODO: Add whitelisted IP addresses*
* **40.121.90.148**

Machines within the network can only be accessed by the Jump Box’s IP address.

* *TODO: Which machine did you allow to access your ELK VM? What was its IP address?*
* **The Jump Box. 40.121.90.148**

A summary of the access policies in place can be found in the table below.

| **Name** | **Publicly Accessible** | **Allowed IP Addresses** |
| --- | --- | --- |
| **Jump Box** | **Yes** | **40.121.90.148** |
| **Web-1** | **No** | **10.0.0.5** |
| **Web-2** | **No** | **10.0.0.6** |

**Web-3 No 10.0.0.7**

**Load Balancer Yes 40.121.90.148**

**ELK Yes 40.121.90.148/10.0.0.4**

**Elk Configuration**

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

* *TODO: What is the main advantage of automating configuration with Ansible?*
* **You can easily launch an entire network in minimal time. This reduces downtime for the network and can help maximize efficiency and profit. It is also easily adaptable by adding the new IP's to the hosts file and then rerunning the playbook. This means you can add, modify, or remove machines quickly as your network demands change. Specifically for ELK, you can also add different modules to multiple machines with just a few keystrokes.**

The playbook implements the following tasks:

* *TODO: In 3-5 bullets, explain the steps of the ELK installation play. E.g., install Docker; download image; etc.*
* **Install Docker and Ansible on the Jump Box machine, enabling playbook automation to push software services to other machines**
* **Added ELK VM to Hosts file under separate Group header**
* **Run Playbook file via ansible-playbook command which pushed tasks to all machines assigned or configured to receive**

The following screenshot displays the result of running docker ps after successfully configuring the ELK instance.

**Note**: The following image link needs to be updated. Replace docker\_ps\_output.png with the name of your screenshot image file.

**Target Machines & Beats**

This ELK server is configured to monitor the following machines:

* *TODO: List the IP addresses of the machines you are monitoring*
* **Web-1 (10.0.0.5), Web-2 (10.0.0.6), Web-3 (10.0.0.7)**

We have installed the following Beats on these machines:

* *TODO: Specify which Beats you successfully installed*
* **Filebeat and Metricbeat**

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

* *TODO: In 1-2 sentences, explain what kind of data each beat collects, and provide 1 example of what you expect to see. E.g., Winlogbeat collects Windows logs, which we use to track user logon events, etc.*
* **Filebeat allows the ELK Server to monitor files being accessed whether by authorized or unauthorized users. It keeps track of access so if there were changes made to sensitive files, the user can figure out when this happened. Metricbeat allows the ELK Server to monitor valuable usage metrics like CPU usage, memory usage, and monitors what is coming in and out of the server.**

**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 **deployment-playbook** file to **/etc/ansible/roles**.
* Update the **ansible.cfg file** to include the **remote\_user=(USERNAME FOR ADMIN OF EACH VIRTUAL MACHINE)**
* Run the playbook, and navigate to **96.38.207.202:5601/app/kibana** to check that the installation worked as expected.

*TODO: Answer the following questions to fill in the blanks:*

* *Which file is the playbook? Where do you copy it?*
* **The playbook is the yml file. It’s copied into an ansible container**
* *Which file do you update to make Ansible run the playbook on a specific machine?*
* **Ansible.cfg and hosts file**
* *How do I specify which machine to install the ELK server on versus which to install Filebeat on?*
* Edit the /
* \_Which URL do you navigate to in order to check that the ELK server is running?
* **96.38.207.202:5601/app/kibana**

*As a* ***Bonus****, provide the specific commands the user will need to run to download the playbook, update the files, etc.*

**1. sudo docker container list -a**

**2. sudo docker start [container name]**

**3. sudo docker attach [container name]**

**4. nano /etc/ansible/playbook.yml**

**5. ansible-playbook /etc/ansible/playbook.yml**