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

Elk Github repository: **https://github.com/brnakj78/Josh\_Brnak\_Project\_1**

**The network diagram for this project is located in the shared GitHub repository provided above as: ELK\_Stack\_Cloud\_Diagram.**

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 \_\_\_\_\_ file may be used to install only certain pieces of it, such as Filebeat.

* *TODO: Enter the playbook file.* **Please see the Playbooks folder in the GitHub repository above for copies of these files.**

**ansible.cfg – This is the configuration file for ansible. For the ELK stack we need to make sure the ELK server username is set as a remote users. For this project the ELK server username was set the same as the DVWA web servers’ usernames.**

**ansible hosts file – Allows you to set up groups with IP addresses that can be used in YAML files in the hosts line in the header section. In our instance there is a webservers group that has the IP addresses of the three DVWA web servers, and another group, elkservers, that has the IP address of the ELK server on which to install an ELK stack.**

**ELK\_VM\_Config\_with\_Docker.yml – Installs Docker and the ELK Docker container on the ELK server**

**filebeat-configuration.yml – The configuration file used by the Filebeat installation playbook.**

**filebeat-installation.yml – Filebeat installation playbook. Installs Filebeat, copies the filebeat config file to the servers being monitored (DVWA web servers in our instance), and then starts Filebeat.**

**metricbeat-configuration.yml – The configuration file used by Metricbeat installation playbook.**

**metricbeat-installation.yml - Metricbeat installation playbook. Installs metricbeat, copies the Metricbeat config file to the servers being monitored (DVWA web servers in our instance), and then starts Metricbeat.**

This document contains the following details:

* Description of the Topology
* 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.

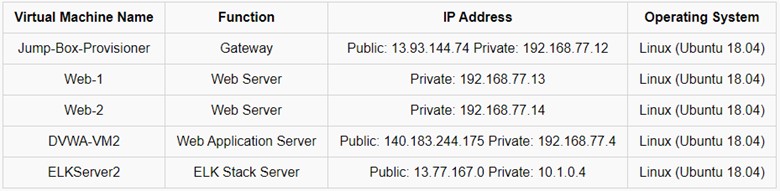
Load balancing ensures that the application will be highly \_\_\_\_\_, in addition to restricting \_\_\_\_\_ to the network.

* *TODO: What aspect of security do load balancers protect? What is the advantage of a jump box?* **Load balancers protect against denial of service attacks by maintaining the availability of, in our instance, a web app’s servers. The advantage of a jump box is it acts as a single point gateway to access the networks servers. The jump box can then be easily configured to restrict access to the servers.**

Integrating an ELK server allows users to easily monitor the vulnerable VMs for changes to the \_\_\_\_\_ and system \_\_\_\_\_.

* *TODO: What does Filebeat watch for?* **Filebeat monitors log files and log events of specified locations (the DVWA web server in our instance) that are captured in Elasticsearch. Filebeat watches for curios behavior across the log files and log events (ie high volumes of files/events) and can adjust its monitoring parameters accordingly.**
* *TODO: What does Metricbeat record?* **Metricbeat records specified metrics such as CPU usage, memory, and load an operating system has. In our case metrics associated with the DVWA web servers.**

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*.

****

**Access Policies**

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

Only the \_\_\_\_\_ machine can accept connections from the Internet. Access to this machine is only allowed from the following IP addresses:

* *TODO: Add whitelisted IP addresses:* **My personal public IP address, 73.229.238.100, is a whitelisted IP address.**

Machines within the network can only be accessed by \_\_\_\_\_.

* *TODO: Which machine did you allow to access your ELK VM? What was its IP address?* **I allowed the jump box virtual machine to access my ELK virtual machine. Its IP address is 192.168.77.12.**

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



**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?* **The main advantages of automating configuration of the ELK machine with Ansible is that changes to the configuration can be implemented quickly from one source, user’s config files are consistent, and new machines can easily be added and configured as other machines are configured. Automating configuration also reduces the chance of typo errors.**

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 repository**
* **Install latest version of Docker Engine**
* **Install Python 3 package installer and then Python Docker module**
* **Increase virtual memory of ELK server**
* **Install and launch an ELK Docker container using an ELK container image and publish that elk container to http ports 5601, 9200, 5044**

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

**Please see the images folder in the Github repository for a screenshot of “Successful\_Config\_Install\_of\_ELK\_Instance”**

**Target Machines & Beats**

This ELK server is configured to monitor the following machines:

* *TODO: List the IP addresses of the machines you are monitoring.* **The machines in which the ELK server is configured to monitor are Web-1 webserver IP address: 192.168.77.13, Web-2 webserver IP address: 192.168.77.14, and DVWA webserver IP address: 192.168.77.4**

We have installed the following Beats on these machines:

* *TODO: Specify which Beats you successfully installed.* **Filebeat and Metricbeat were successfully installed on the above machines.**

**Please see Filebeat\_Install\_Successful\_GUI and Metricbeat\_Install\_Successful\_GUI in the images folder in Github repository to view that Filebeat and Metricbeat are installed and running and capturing the appropriate data.**

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 collects log files and log events which we can use to track system log events such as Web Server 1 timing out. Metricbeat collects metrics associated with different operating systems. With metricbeat we would be able to see the CPU usage for our different DVWA webservers.**

**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 \_\_\_\_\_ file to \_\_\_\_\_.
* Update the \_\_\_\_\_ file to include...
* Run the playbook, and navigate to \_\_\_\_ 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 to install filebeat is filebeat-installation.yml and it is located under /etc/ansible/roles. Within the filebeat-installation playbook there is a module in which the filebeat configuration file we created on the ELK server is copied to the webservers.**
* *Which file do you update to make Ansible run the playbook on a specific machine? How do I specify which machine to install the ELK server on versus which to install Filebeat on?* **In order to specify which machine a particular playbook will run on you need to update the Ansible hosts file. In the hosts file you can set up groups (webservers and elkservers in our case) with IP addresses in that group. Then in the install playbook in the header you can dictate which group (hosts) to run the playbook on.**
* *Which URL do you navigate to in order to check that the ELK server is running?* **To check the ELK server is running you can navigate to the URL http://ELK\_server\_Public\_IP:5601. In this case the URL to check the ELK server is running would be http://13.77.167.0:5601**

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