## Automated ELK Stack Deployment

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

**Images/docker-ps.png**

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

- \_TODO: **install-elk.yml**

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 **sufficient**, in addition to restricting **traffic** to the network.

- \_TODO: What aspect of security do load balancers protect? What is the advantage of a jump box?\_**It can prevent ddos by shifting attack traffic from the corporate to a public cloud, it is a secure computer.**

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

- \_TODO: What does Filebeat watch for? **Filebeat watches specified log data or locations.**

- \_TODO: What does Metricbeat record?\_**it records metrics and satistics**

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 | |**

**|------------|---------------------------|---------------|------------------|---|**

**| jumpbox | gateway | 104.211.23.11 | linux | |**

**| webserver1 | store,process,distributes | 10.0.0.6 | cloud | |**

**| webserver2 | store,process,distributes | 10.0.0.8 | cloud | |**

**| elk server | monitors | 23.100.66.196 | cloud | |**

### Access Policies

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

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

- **104.211.23.11**

Machines within the network can only be accessed by **jumpbox**.

- \_TODO: Which machine did you allow to access your ELK VM? What was its IP address?\_**my jump box and the ip was 104.211.23.11**

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

| Name | Publicly Accessible | Allowed IP Addresses |

|----------|---------------------|----------------------|

| Jump Box | Yes/No | 10.0.0.1 10.0.0.2 |

| | | |

| | | |

### 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?\_**it makes things faster and more simpler**

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.io**

**- install python3-pip**

**- install docker module**

**- download the image and start it**

- ...

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

**Images/docker-ps.png**

### Target Machines & Beats

This ELK server is configured to monitor the following machines:

**Webserver1**

**Webserver2**

We have installed the following Beats on these machines:

**- I have installed filebeat**

These Beats allow us to collect the following information from each machine: **filebeat collects logs and locations so specific things. Metricbeats collects metric and statistics a example of this will be seeing the amount of logins into the machine**

### 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 **elk-playbook.yml** file to **/etc/ansible/**

- Update the **ansible-playbook** file to include...

- Run the playbook, and navigate to **http://23.100.67.196:5601/app/kibana** to check that the installation worked as expected.

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