**## Automated ELK Stack Deployment**

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

**TO DO:** [**Virtual Network Diagram**](https://github.com/navarrofox21/UT-Cyber-Bootcamp-/tree/main/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 config file may be used to install only certain pieces of it, such as Filebeat.**

**TODO:** [**Ansible Files (Playbooks)**](https://github.com/navarrofox21/UT-Cyber-Bootcamp-/tree/main/Ansible)

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

**What aspect of security do load balancers protect? Load balancers allow for traffic to be re-routed in the event a server is infiltrated.**

**What is the advantage of a jump box? A jump box allows for logging to be conducted in a single area. Other VM’s can be accessed and rules can be maintained from the provisioner.**

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

**What does Filebeat watch for? Log files.**

**What does Metricbeat record? Metrics.**

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 |**

**| ELK-SERVER| Ubuntu | 10.1.0.4 |Linux |**

**| Web-1 | Ubuntu | 10.0.0.5 |Linux |**

**| Web-2 | Ubuntu | 10.0.0.6 |Linux |**

**### Access Policies**

**The machines on the internal network are not 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:**

* **Allow SSH from my public IP address 69.xxx.xxx.xx**
* **Jump box Access 22 from source 10.0.0.4**

**Machines within the network can only be accessed by the jump box provisioner.**

**Which machine did you allow to access your ELK VM? What was its IP address?** **The jump box provisioner IP 10.0.0.4**

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

**| Name | Publicly Accessible | Allowed IP Addresses |**

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

**| Jump Box | Yes | My personal IP 69 |**

**| Web-1 | No | 10.0.0.4 Port 22 |**

**| Web-2 | No | 10.0.0.4 Port 22 | | ELK-SERVER| No | My personal IP |**

**### Elk Configuration**

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

**What is the main advantage of automating configuration with Ansible? Deployment is playbooks can be completed in a timely manner.**

**The playbook implements the following tasks:**

**In 3-5 bullets, explain the steps of the ELK installation play. E.g., install Docker; download image; etc.\_**

1. **Install Docker.io**
2. **Install Python.pip**
3. **Download/Launch ELK Container**

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

**TODO: Update the path with the name of your screenshot of docker ps output](Images/docker\_ps\_output.png)**

Text

Description automatically generated

**### 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: IP 10.0.0.5**
* **Web-2: IP 10.0.0.6**

**We have installed the following Beats on these machines:**

**TODO: Specify which Beats you successfully installed**

* **Filebeat**
* **Metricbeat**

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

* **Metricbeat allows for the virtual machines core statistics to be monitored while Firebeat collects specific log files.**

**### 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 yml file to the ansible directory.**
* **Update the “config” file to include ports/users.**
* **Run the playbook and navigate to Kibana to check that the installation worked as expected.**

**Answer the following questions to fill in the blanks:\_**

* **Which file is the playbook? Where do you copy it? Run: curl https://gist.githubusercontent.com/slape/5cc350109583af6cbe577bbcc0710c93/raw/eca603b72586fbe148c11f9c87bf96a63cb25760/Filebeat >> /etc/ansible/filebeat-config.yml. This will be ran from the ansible container to create the filebeat configuration.**
* 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?\_**Elk Installation and VM Configuration.**
* **Which URL do you navigate to in order to check that the ELK server is running? Navigate to http://[your.VM.IP]:5601/app/kibana**