**Automated ELK Stack Deployment**

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

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.

* *my-playbook.yml* -To configure WebVMs
* *elk-playbook.yml* -To configure ELK
* *filebeat-playbook.yml* -To configure Filebeat
* *metricbeat\_playbook.yml* -To configure 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 accessible, in addition to restricting access to the network.

* Using a load balancer increases the availability of a network by providing a defence against a DoS attack. While using a jumpbox allows one secure point off access to the network to run administrative tasks for your machines/servers.

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

* Filebeat watches for any changes in log that the user has specified and forwards them onward to the ELK stack
* Metricbeat collects metric data from any specified servers and forwards them onward to the ELK stack

| **Name** | **Function** | **IP Address** | **Operating System** |
| --- | --- | --- | --- |
| Jump-Box | Gateway | 10.0.0.4 | Linux Ubuntu |
| Web-1 | Webserver | 10.0.0.5 | Linux Ubuntu |
| Web-2 | Webserver | 10.0.0.6 | Linux Ubuntu |
| ELK-Proj-VM | Monitoring webserver | 10.1.0.4 | Linux Ubuntu |

**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:

* My public IP address.

Machines within the network can only be accessed by the Jump-Box.

* The ELK-Proj-VM is only accessible through the Jump-Box with the IP address 40.83.151.123

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

| **Name** | **Publicly Accessible** | **Allowed IP Addresses** |
| --- | --- | --- |
| Jump Box | Yes, by a set IP | <my-public-IP> |
| Web-1 | No | 10.0.0.4, 10.0.0.6, 10.1.0.4 |
| Web-2 | No | 10.0.0.4, 10.0.0.5, 10.1.0.4 |
| ELK-Proj-VM | No | 10.0.0.4, 10.0.0.5, 10.0.0.5 |

**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 these tasks is to reduce the amount of daily tasks an administrator will need to action. Making it simple by only running one paybook instead of multiple commands.

The playbook implements the following tasks:

* Install Docker.io
* Install plython3
* Install Docker module
* Increase virtual memory
* Use more memory (memory set in previous command)
* Download and launch ELK container
* Enable service on boot.

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

A screenshot of a computer

Description automatically generated with medium confidence

**Target Machines & Beats**

This ELK server is configured to monitor the following machines:

* *10.0.0.5* and *10.0.0.6*

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 all changes in the log files the user specifies e.g. audit logs, whereas Metricbeat collects metric data e.g. CPU usage, inbound traffic. This data will then be pushed out into the ELK using interface to easily read the collected data.

**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 hosts file to include web servers IP address.
* Run the playbook and navigate to <http://[your.IP]:5601> (in my case it was http://20.36.40.78:5601 to check that the installation worked as expected.