Automated ELK Stack Deployment

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

![TODO: Update the path with the name of your diagram](Images/*Project 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 _*Yaml*_ file may be used to install only certain pieces of it, such as Filebeat.

- _TODO: Enter the playbook file._
  - *Install-elk.yml*
  - *filebeat-playbook.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 _*available*_, 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?_*Load balancers protect the Availability of, in our case, webservers. The advantage of a jump box is we are creating a single point where we are able to do admin tasks on many machines*.

Integrating an ELK server allows users to easily monitor the vulnerable VMs for changes to the _*Logs*_ and system _*files*_.
- _TODO: What does Filebeat watch for? *Filebeat watches log files for log events*
- _TODO: What does Metricbeat record? *Metricbeat watches system and services resources that are running.*

The configuration details of each machine may be found below.

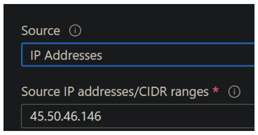
_Note: Use the [Markdown Table Generator](http://www.tablesgenerator.com/markdown_tables) to add/remove values from the table_.

Name	Function	Ip Address	Operating System
Jump Box	Gateway	10.0.0.4	Linux
Web1	Server	10.0.0.5	Linux
Web2	Server	10.0.0.6	Linux
ELK	Log Server	10.1.0.4	Linux

Access Policies

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

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

- _TODO: 

Source ⓘ

IP Addresses

Source IP addresses/CIDR ranges * ⓘ

45.50.46.146

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? *To access the ELK VM the jumpbox is allowed. To access the ELK Server you use the home IP address and port 5601.*

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

Name	Publicly Accessible	Allowed IP Addresses
Jump Box	Yes	Home IP Address
Web1	No	10.0.0.4
Web2	No	10.0.0.4
ELK	Yes	Home IP Address

Elk Configuration

Ansible was used to automate configuration of the ELK machine. No configuration was performed manually, which is advantageous because *it allows for equal deployment of the same system, updates and applications needed. The ability to replicate the same machine instances allows for less variables and saves on time to deploy.*
- _TODO: What is the main advantage of automating configuration with Ansible? *The main advantage is the process is streamlined and saves time. This can also reduce human error with forgetting to install some items as they will all have the same items.*

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._

- *Set the max memory for the VM*
- *Install Docker.io*
- *Install Python3*
- *Install Python Docker Module*
- *Download and launch Web 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/Screenshot_Sudo_Docker_PS)

Target Machines & Beats
This ELK server is configured to monitor the following machines:
- _TODO: List the IP addresses of the machines you are monitoring
    - *Web1 - 10.0.0.5*
    - *Web2 - 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:
- _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 will allow us to collect logs from either a specified location or you can let it filter all logs. Metricbeat will allow you to collect data on system or application resource*

usage. Both of these can be forwarded to Logstash for logging that can be viewed at a later time or based on alerts can be brought to the monitors attention.

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 *install-elk.yml* file to */etc/ansible/roles/install-elk.yml*.
- Update the *hosts* file to include *the [ELK] server group and 10.1.0.4 ansible_python_interpreter=/usr/bin/python3*

- Run the playbook, and navigate to *ELK server ip address: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?
  - *You will copy /etc/ansible/files/filebeat-config.yml to /etc/filebeat/filebeat.yml*

- _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?_

  - *You will update the Hosts file to specify which machine will get the playbook*
  - *In your yaml code you will put which hosts you would like it to be installed on*

- _Which URL do you navigate to in order to check that the ELK server is running?
  - *ELK server ip address:5601/app/kibana*

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