## 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](Untitled Diagram.drawio)

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

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 Damn 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 do exactly what they say, they ballance the load across X machines to protect from excessive load like a DDoS

A jump box is super nice to have because it's your own personal secure back door to everywhere in your system.

Integrating an ELK server allows users to easily monitor the vulnerable VMs for changes to the log files and system services

| Name | Function | IP address | Operating system |
| --- | --- | --- | --- |
| jumpbox | gateway | 10.0.0.4 | linux |
| web1 | DVWA | 10.0.0.5 | linux |
| web2 | DVWA | 10.0.0.6 | linux |
| ELK | ELK | 10.1.0.4 | linux |

### Access Policies

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

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

- \_TODO: Add whitelisted IP addresses\_ IDK I let anyone connect to my load balancer, so \*

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 can get to it from the ansible container on my jumpbox 10.0.0.4 and my DVWAs report to it with filebeat and metricbeat

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

| name | Public access | Allowed IP |
| --- | --- | --- |
| Jump box | No | My personal ssh |
| Web1 and web2 | no | None, forwards from load balancer |
| elk | no | My personal ip |
| Load balancer | yes | all |

### 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?\_

Once you get it working you don't have to do it again, you just run the playbook again and boom you’re done

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
* Install python 3
* Force it to use all the memory
* Download and launch an elk container off an image
* Update systemctl to start elk on boot

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/elk docker ps screenshot)

### Target Machines & Beats

This ELK server is configured to monitor the following machines:

Web1 and Web2

We have installed the following Beats on these machines:

Filebeats and metricbeats

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.\_

Between the 2 of them you can watch machine performance, cpu usage, up time things like that and you can also see how many visitors you have to your site and where they are from and other personal information.

### 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 hosts file to include target IP

- 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?\_

- \_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?\_

- \_Which URL do you navigate to in order to check that the ELK server is running?

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

---

- name: Config Web VM with Docker

hosts: webservers

become: true

tasks:

- name: docker.io

apt:

update\_cache: yes

name: docker.io

state: present

- name: Install pip3

apt:

name: python3-pip

state: present

- name: Install Python Docker Module

pip:

name: docker

state: present

- name: download and launch a docker web container

docker\_container:

name: dvwa

image: cyberxsecurity/dvwa

state: started

restart\_policy: always

published\_ports: 80:80

- name: Enable docker service

systemd:

name: docker

enabled: yes