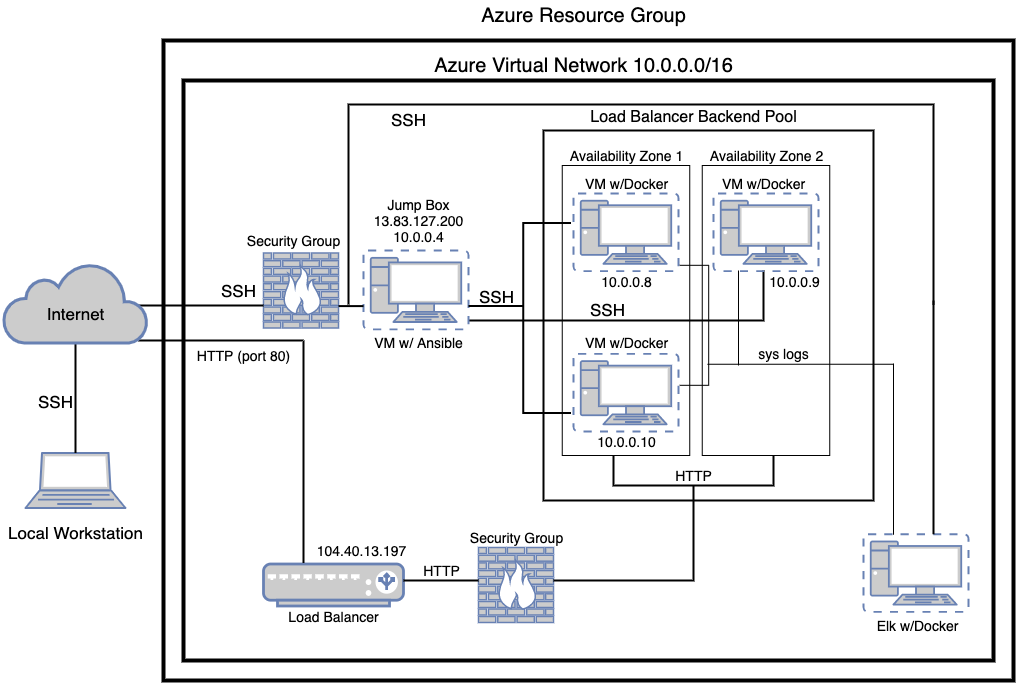
## Automated ELK Stack Deployment

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

![TODO: Update the path with the name of your diagram](Images/diagram\_filename.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 elk.yml file may be used to install only certain pieces of it, such as Filebeat.

- \_TODO: Enter the playbook file.\_

filebeat.yml

filebeat-config.yml

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 available, in addition to restricting inbound access to the network.

- \_TODO: What aspect of security do load balancers protect? What is the advantage of a jump box?\_ The load balancer makes it so that processing the incoming traffic is shared by both vulnerable web servers. The advantage of a jump box is that only authorized users will be able to SSH into the web servers.

Integrating an ELK server allows users to easily monitor the vulnerable VMs for changes to the file systems of the VMs on the network and system metrics.

- \_TODO: What does Filebeat watch for?\_ It is used to collect Apache logs and detect changes to the filesystem.

- \_TODO: What does Metricbeat record?\_ Metricbeat is used to record system metrics such as CPU usage. We can also use it to detect SSH login attempts, failed sudo escalations, and CPU/RAM statistics.

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 |

| Web 1 | Web Server | 10.0.0.8 | Linux |

| Web 2 | Web Server | 10.0.0.9 | Linux |

| Web 3 | Web Server | 10.0.0.10 | Linux |

| ELK Server | Monitoring | 10.1.0.4 | Linux |

### 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:

- \_TODO: Add whitelisted IP addresses\_ 216.110.198.35, 63.155.87.80

Machines within the network can only be accessed by each other.

- \_TODO: Which machine did you allow to access your ELK VM? What was its IP address?\_

The Jump Box has access to the ELK VM, it’s IP address is 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 | 216.110.198.35, 63.155.87.80 |

| Web 1 | No | 10.0.0.1-254 |

| Web 2 | No | 10.0.0.1-254 |

| Web 3 | No | 10.0.0.1-254 |

### 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 allows us to automate daily tasks and save time.

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

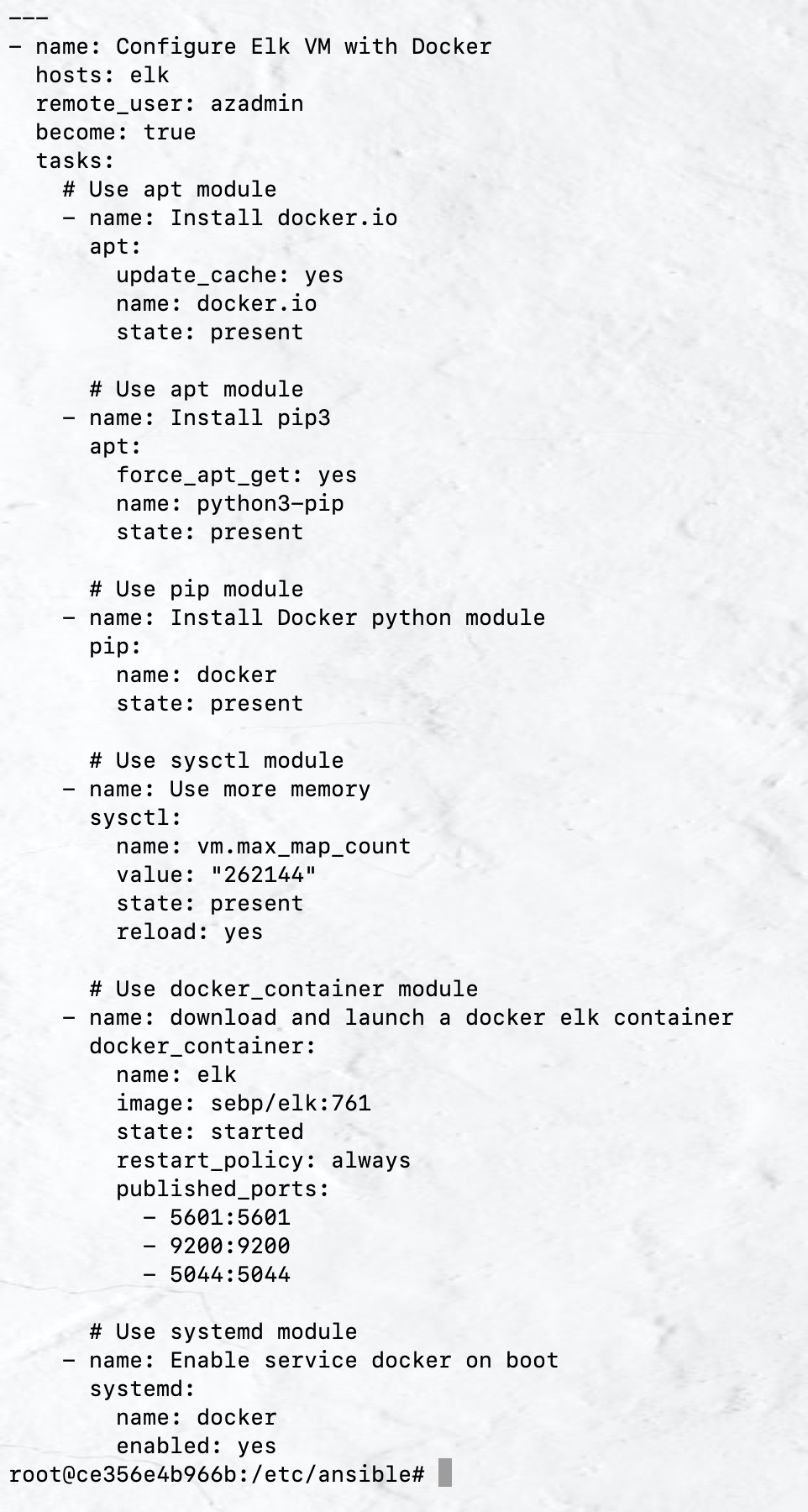
- … Install Python

- … Install Docker Python module

- … Download and install a docker elk container

- … Enable service docker 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/docker\_ps\_output.png)

### 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 - 10.0.0.8

Web 2 - 10.0.0.9

Web 3 - 10.0.0.10

We have installed the following Beats on these machines:

- \_TODO: Specify which Beats you successfully installed\_ Filebeat

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 - It is used to collect Apache logs and detect changes to the filesystem.

### 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: We use the Jump Box for this.

SSH into the control node and follow the steps below:

- Copy the playbook file to the ansible control node.

- Update the hosts file to include...which VMs to run the playbook on.

- Run the playbook, and navigate to 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?\_ The playbook file is elk.yml, you copy it in /etc/ansible/elk.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 update /etc/ansible/hosts and add the IP address of the machine you want to install Filebeat on under [webservers] and you add the IP address of the machine you want to install the ELK server on under [elk].

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

(IP address of the machine):5601 EXAMPLE: 10.0.0.8:5601

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

ansible-playbook install\_elk.yml elk

ansible-playbook install\_filebeat.yml webservers

To update the hosts file run this command in /etc/ansible/ - nano hosts