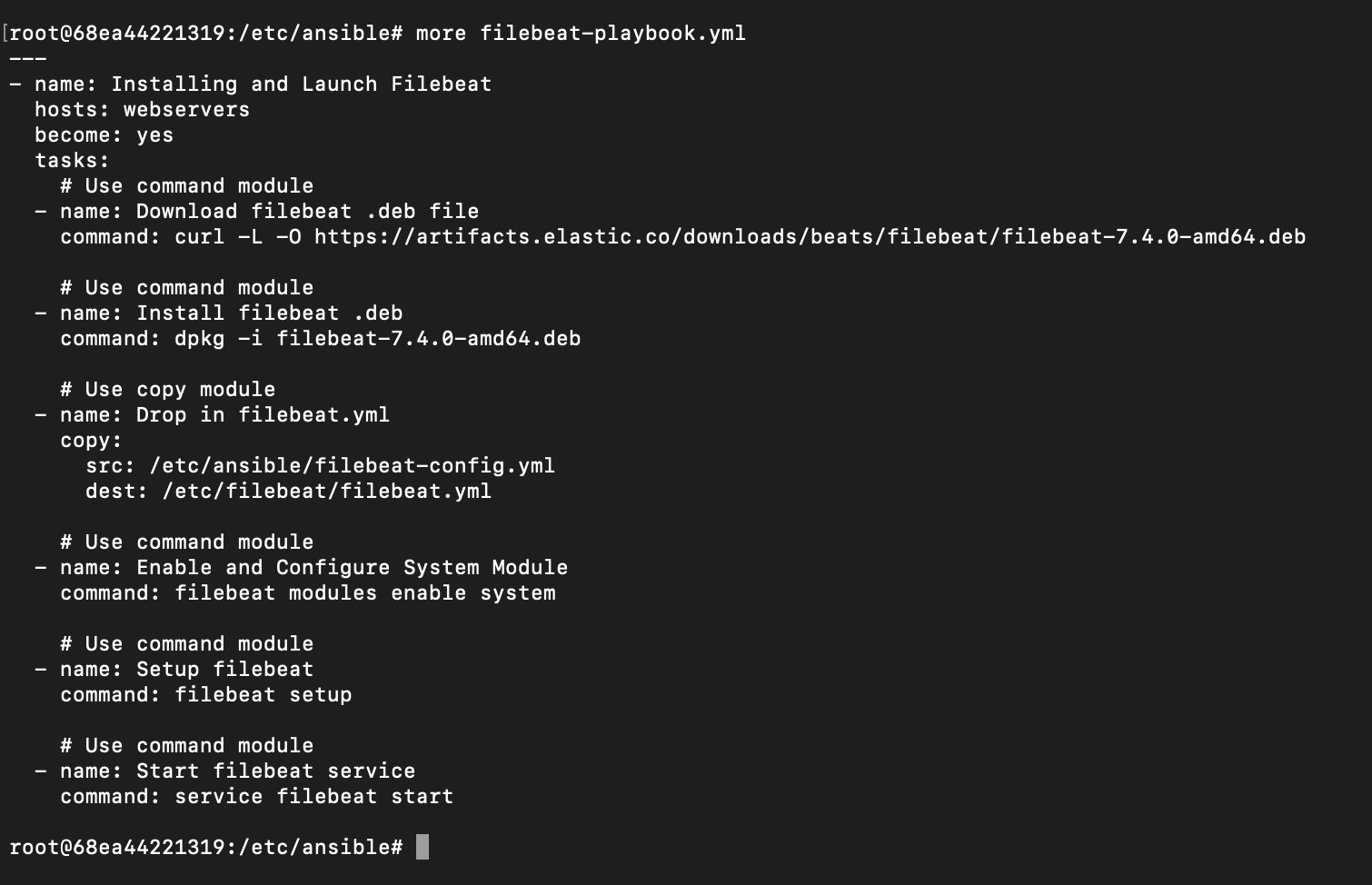
## Automated ELK Stack Deployment

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

- \_TODO: Enter the playbook file.\_



This document contains the following details:

- Description of the Topologu

- 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?\_

Integrating an ELK server allows users to easily monitor the vulnerable VMs for changes to the \_\_LOG DATA + FILES\_\_\_ and system \_METRIC\_\_\_\_.

- \_TODO: What does Filebeat watch for?\_(13.1 Student Guide) “collects data about the file system”

- \_TODO: What does Metricbeat record?\_(13.1 Student Guide) “collects machine metrics, such as uptime

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 | 10.0.0.6 | Linux |

| TODO | WEB | 10.0.0.7 | LINUX |

| TODO | WEB | 10.0.0.8 | LINUX |

| TODO | MONITOR | 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\_ 110.175.42.235

Machines within the network can only be accessed by \_\_OTHER VMS\_\_\_.

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

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

| Name | Publicly Accessible | Allowed IP Addresses |

|----------|---------------------|----------------------|

| Jump Box | Yes/No | |

| VM-3 | NO | |

| VM-4 | NO | |

| VM-5. | NO | |

| LOAD BAL.| YES | |

ALLOWED IP ADDRESSES:

JUMP BOX: 10.0.0.7, 10.0.0.8, 10.1.0.4, 13.75.168.142

VM-3: 10.0.0.6, 10.0.0.8, 10.1.0.4, 13.75.168.142

VM-4: 10.0.0.6, 10.0.0.7, 10.1.0.4, 13.75.168.142

VM-5: 10.0.0.6, 10.0.0.7, 13.75.168.142

LOAD BALANCER: 10.0.0.6, 10.0.0.7, 10.0.0.8, 10.1.0.4

### 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?\_Every machine will be configured within minutes, avoiding human error and any issues can be reverted via IaC

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.\_

- Create new vNet (same resource group + new region)

- Create new VM (SSH key from Ansible, place new VM in Hosts file)

- Download contatiner (sebp/elk:761)

- NSG: Incoming rule for new VM port

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\_

10.0.0.7, 10.0.0.8

We have installed the following Beats on these machines:

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

FILEBEAT AND 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: DETECTS CHANGES IN FILES, FOR EXAMPLE LOGS

METRICBEAT: DETECS CHANGES IN SYSTEM METRICS, FOR EXAMPLE CPU PERFORMANCE

### 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 \_CONFIGURATION\_\_\_\_ file to \_ANSIBLE CONTAINER\_\_\_\_.

- Update the \_\_YAML\_\_\_ file to include...

- Run the playbook, and navigate to \_\_server GUI\_\_ 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?\_YAML, /etc/ansible

- \_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?\_Hosts file on Ansible  
[ELK]

10.0.0.7 (example)

- \_Which URL do you navigate to in order to check that the ELK server is running? http://[your.VM.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.\_