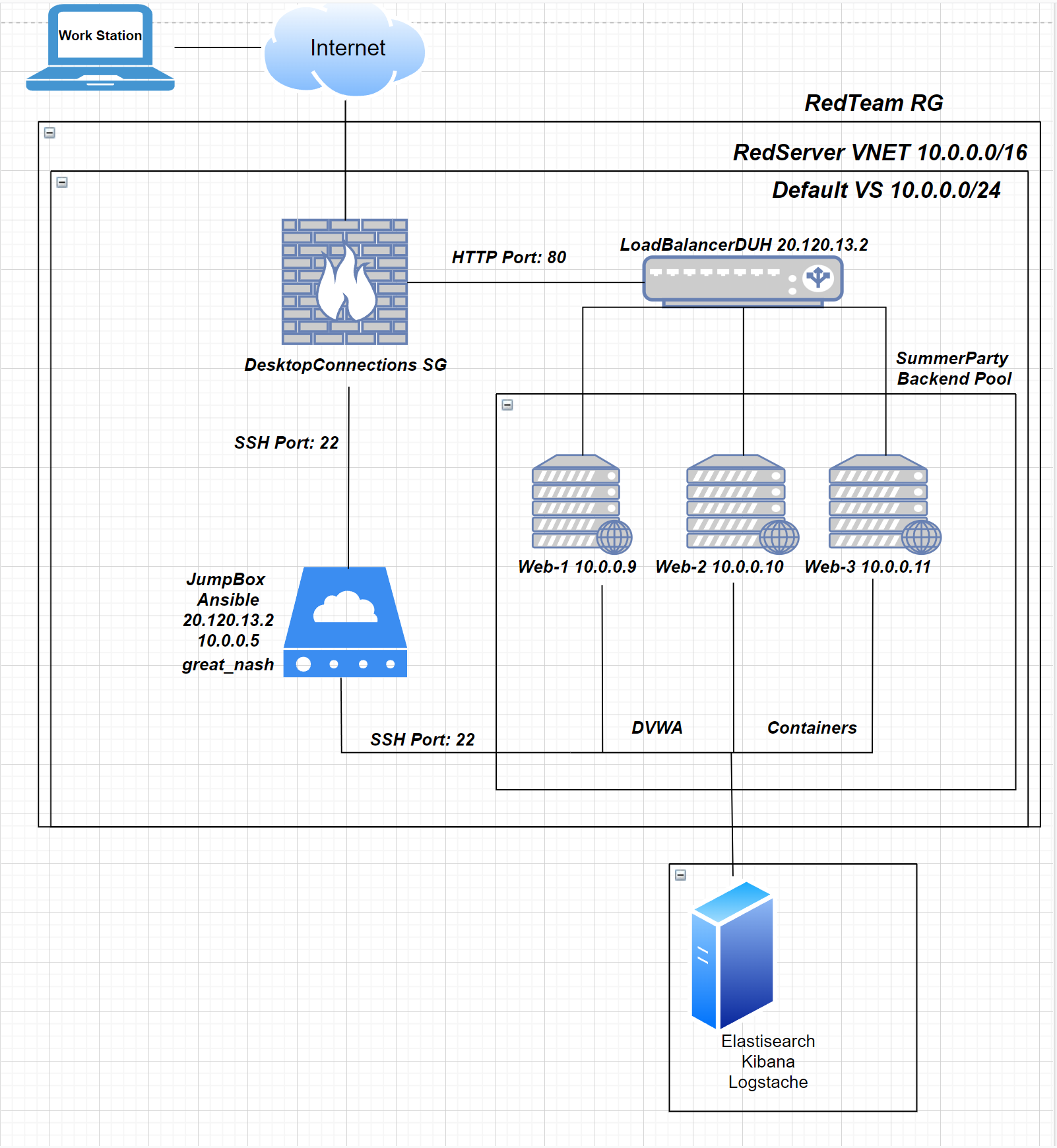
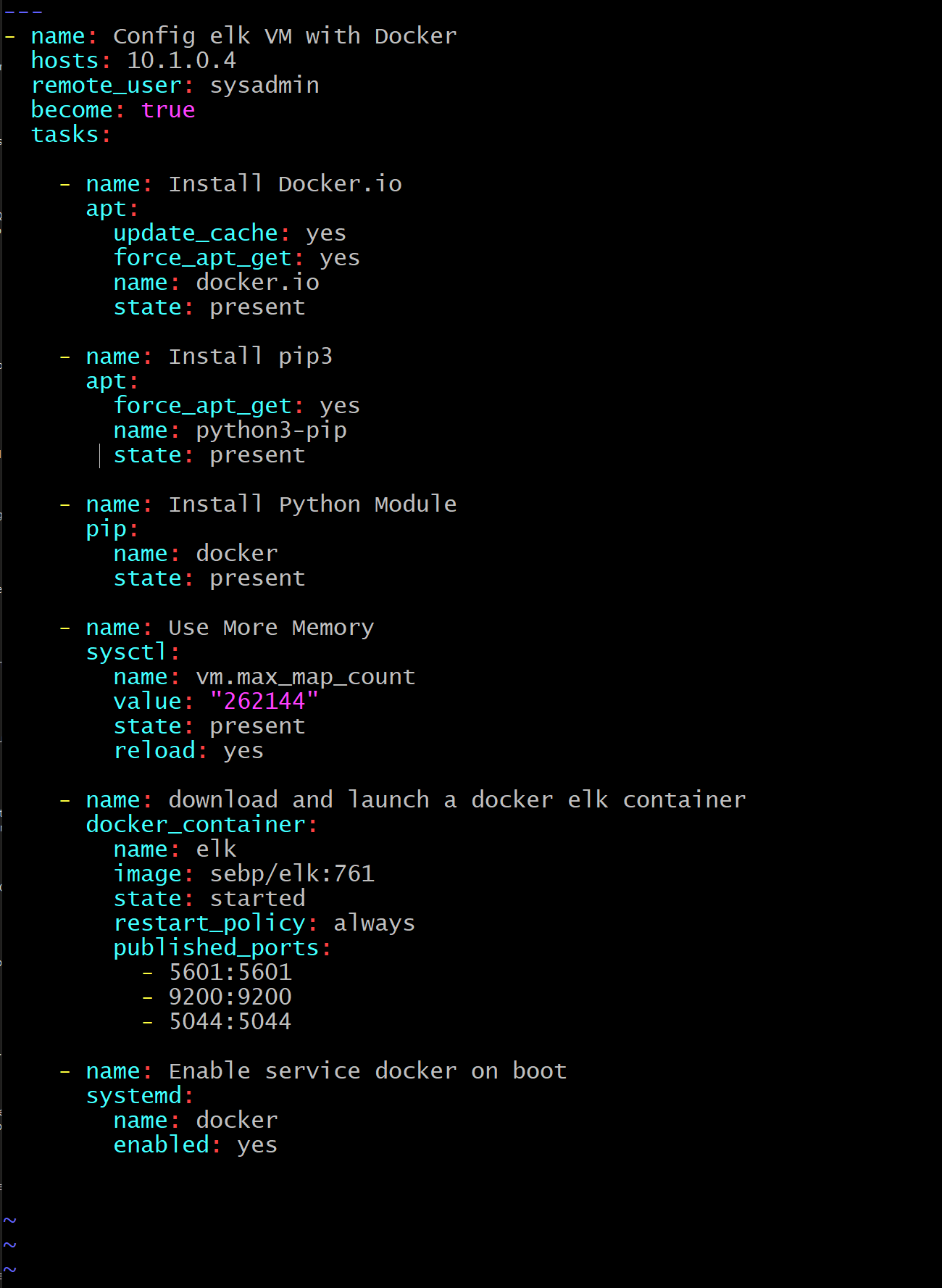
**Automated ELK Stack Deployment**

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. Alternatively, select portions of the \_YAML\_ file may be used to install only certain pieces of it, such as Filebeat.

ELK Config

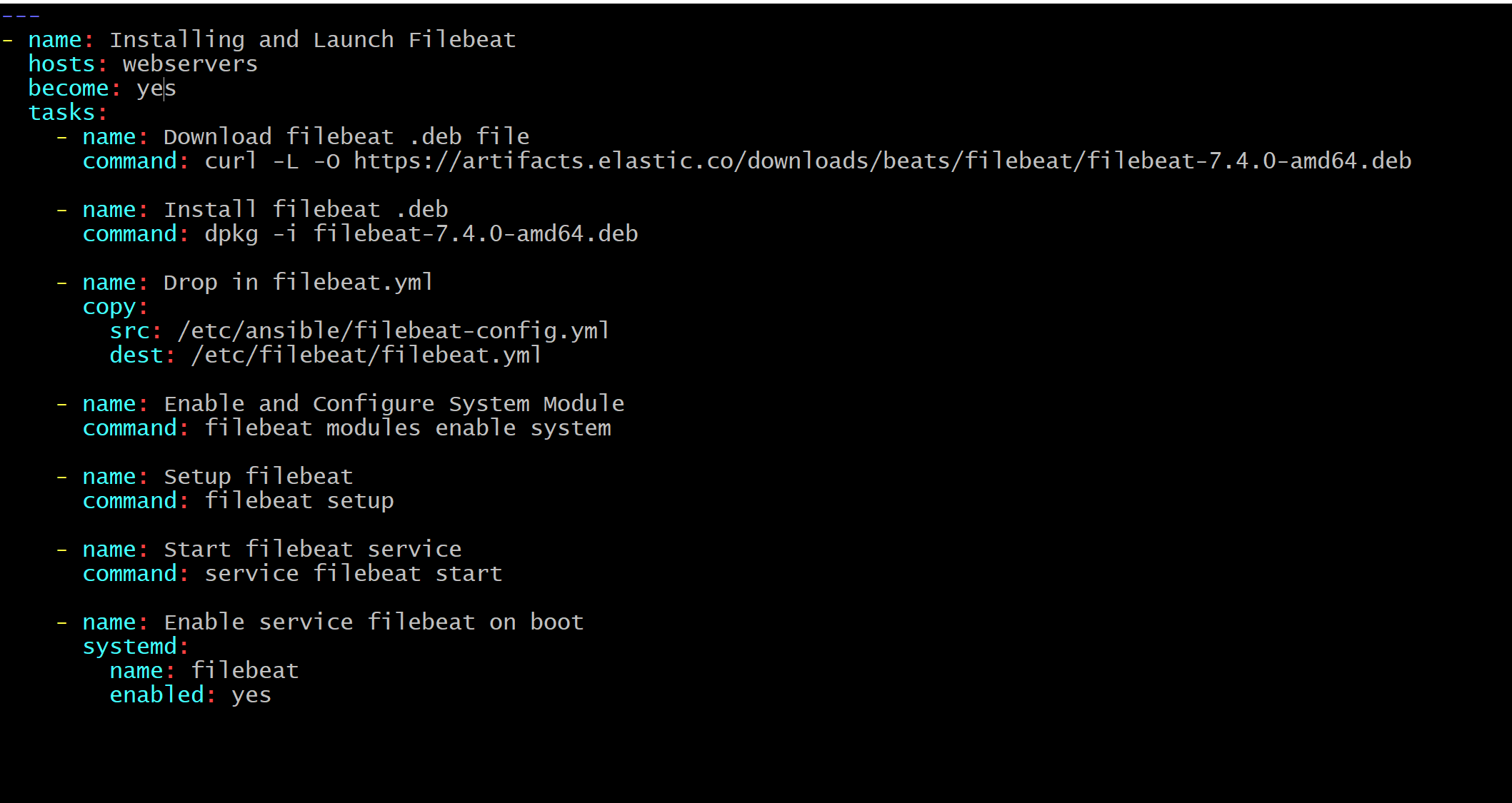


Metricbeat

Text

Description automatically generated

Filebeat



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 \_too much traffic\_ to the network.

- *\_TODO: What aspect of security do load balancers protect? What is the advantage of a jump box?\_*

*LoadBalancers make sure the servers are always available. Jumpboxes allow you to access your server through the backend for maintenance and updates.*

Integrating an ELK server allows users to easily monitor the vulnerable VMs for changes to the \_CPU\_ and system \_\_Logs\_\_\_.

- *\_TODO: What does Filebeat watch for?\_ system logs*

- *\_TODO: What does Metricbeat record?\_ CPU Usage*

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 | 10.0.0.5 | Linux |

| Web-1 | Server | 10.0.0.9 | Linux |

| Web-2 | Server | 10.0.0.10 | Linux |

| Web-3 | Server | 10.0.0.11 | 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: Add whitelisted IP addresses\_*

108.58.146.214

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?\_*

My JumpBox and its IPv4 (private) ip is 10.0.0.5

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

| Name | Publicly Accessible | Allowed IP Addresses |

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

| Jump Box | NO | 108.58.146.214|

| ELK | Front-End Yes (108.58.146.214:5601) Back-End No 10.0.0.5 | 10.0.0.5 (Back-End) 108.58.146.214 (Front-End Port:5601) |

| Web-1, 2, & 3 |Front-End Yes (Port:80) Back-End No 10.0.0.5| 10.0.0.5 (Back-End) Any IP (Front-End Port:80) |

### **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?\_*

More Efficient and less error prone if working on multiple machines at once

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 the VM in Azure in different region and peer it to V-Net of the 3 webservers

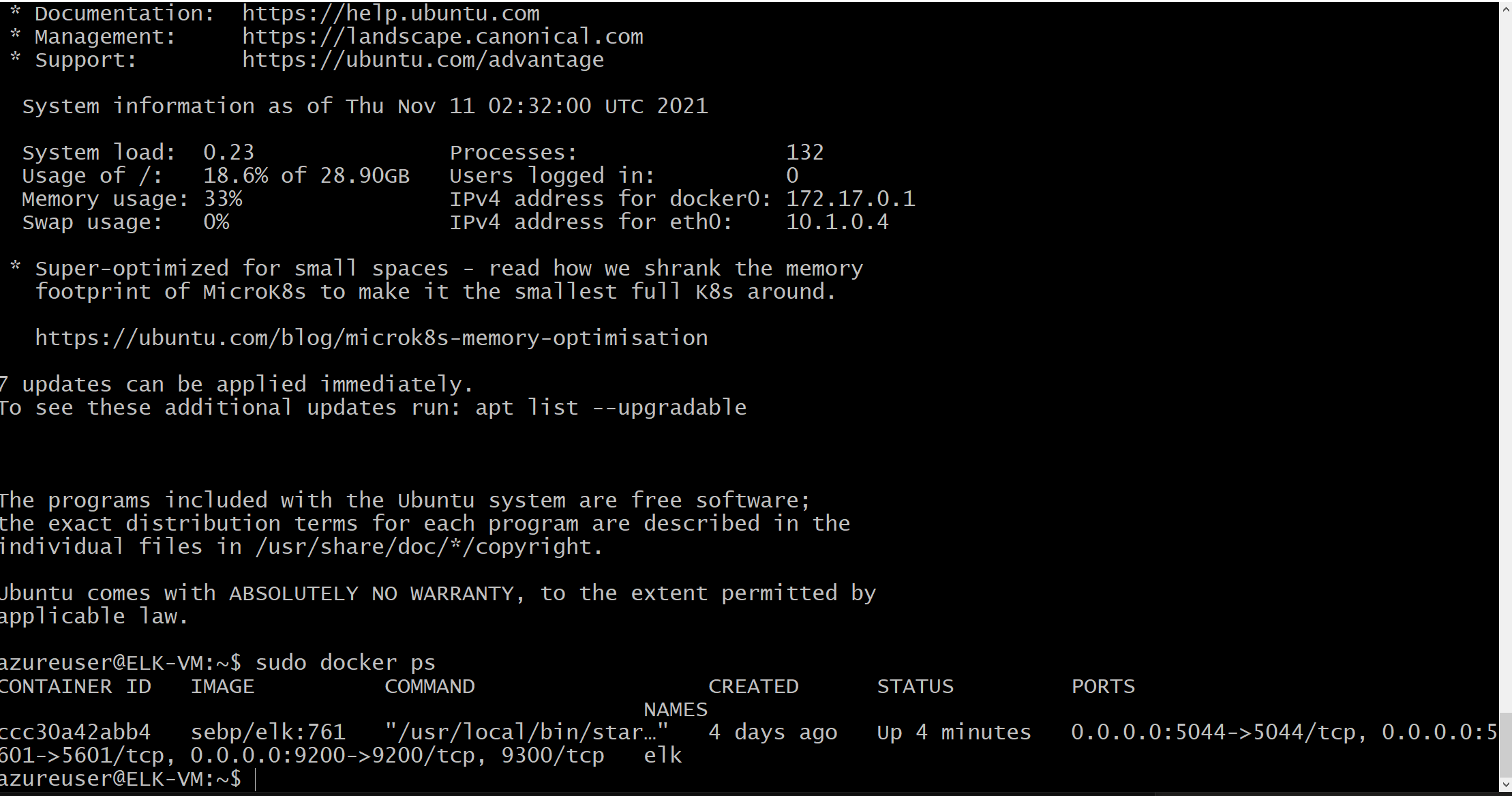
-… Write Ansible Playbook to pull and install ELK Software to deploy on ELK VM

-… Installs docker and runs an ELK Container

- ... Installs Elasticsearch, Logstache, & Kibana on the ELK Container

-… Launch Public ip of ELK VM in browser with Port:5601

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 10.0.0.9*

*Web-2 10.0.0.10*

*Web-3 10.0.0.11*

We have installed the following Beats on these machines:

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

*Both 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.\_*

### **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 \_\_play\_book\_ file to \_/etc/ansible/\_\_.

- Update the \_hosts\_ file to include the ip addreasses of server to configure

- Run the playbook, and ssh into \_ELK Server\_\_ to check that the installation worked as expected by running sudo docker ps and navigate to <ip of ELK VM>:5601/app/kibana#/home

*\_TODO: Answer the following questions to fill in the blanks:\_*

- *\_Which file is the playbook? Where do you copy it?\_*

/etc/ansible/playbooks/ELK.yaml

/etc/ansible/roles/filebeat.yaml

/etc/ansible/roles/metricbeat.yaml

- *\_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?\_*

*The ansible hosts file*

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

Ip addreass of ELK Server with port 5601 EX.

23.101.207.93:5601/app/kibana#/home

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