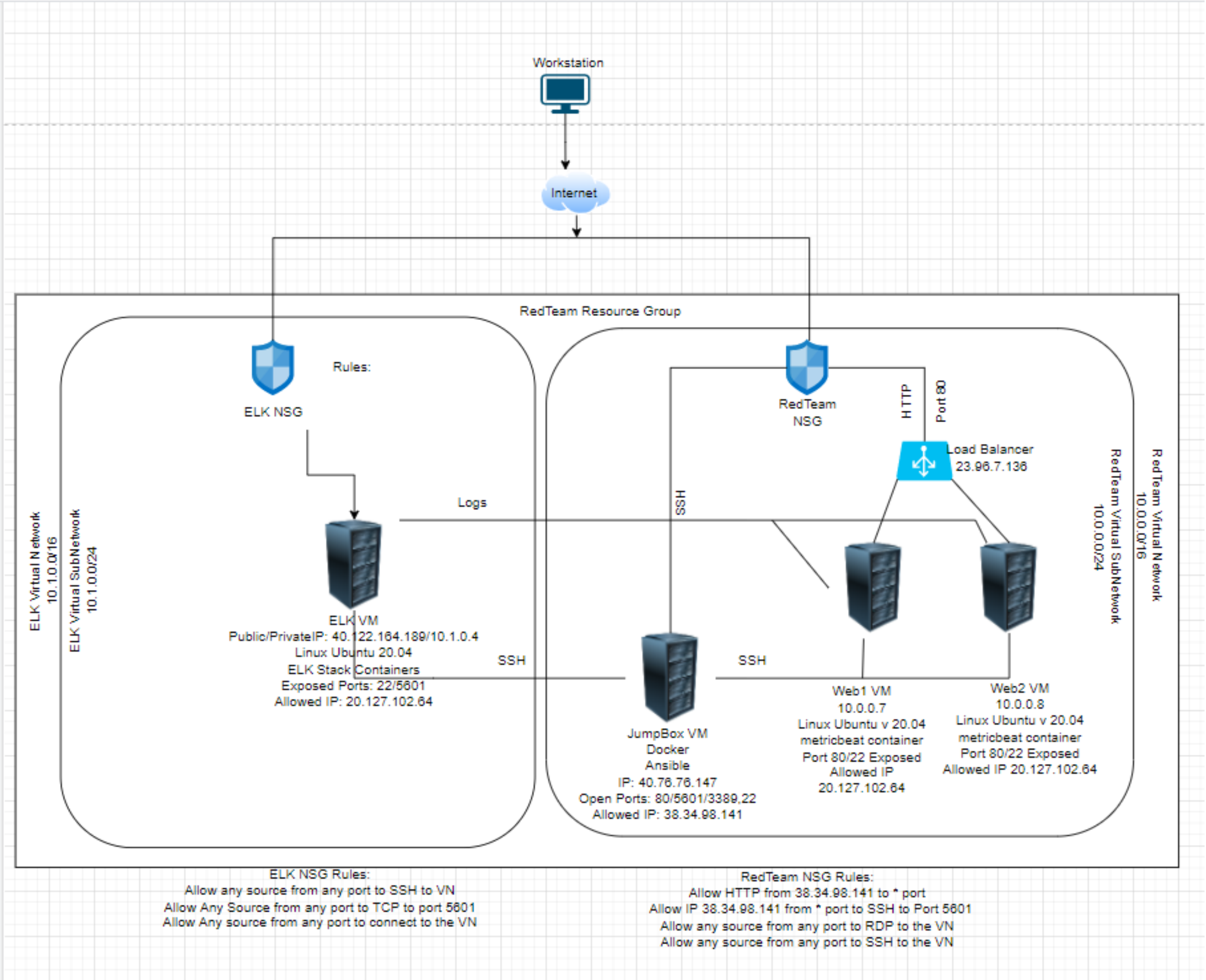
Automated ELK Stack Deployment

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

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?_
  - Load balancers protect a network against distributed denial-of-service attacks. These attacks flood servers with traffic which affects the availability of data.
  - The jump box has the only path to ssh into the class (or company) servers. This reduces the ability of threat actors to access the companies servers directly.

Integrating an ELK server allows users to easily monitor the vulnerable VMs for changes to the data and system logs.

- _TODO: What does Filebeat watch for?_
  - Filebeat monitors log files and locations specified by the administrator. It collects log events and forwards them to Elasticsearch and logstash.
- _TODO: What does Metricbeat record?_
  - Metricbeat records the statistics from the system and services running on your network and presents them in an easily digestible format.

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	10.0.0.1	Linux
DVWA1	Web Server	10.0.0.7	Linux
DVWA2	Web Server	10.0.0.8	Linux
ELK	Elasticsearch Stack	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: Add whitelisted IP addresses_
  - My IP: 38.34.98.141

Machines within the network can only be accessed by the jumpbox.

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

- I allowed the jumpbox to access my ELK VM.
  - Public IP of my jumpbox was 40.76.76.147.
  - Private IP of my jumpbox 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/No	10.0.0.1 10.0.0.2
Web 1&2	No	WebLB 23.96.7.136
Web LB	Yes-HTTP-80	*
ELK	Yes-Kibana-5601	*

Elk Configuration

Ansible was used to automate configuration of the ELK machine. No configuration was performed manually, which is advantageous because...

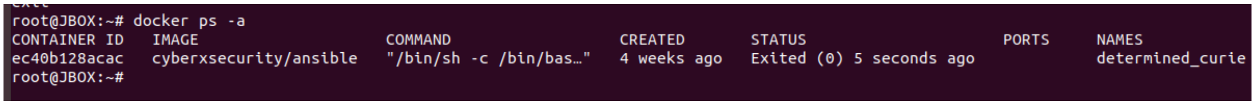
- _TODO: What is the main advantage of automating configuration with Ansible?_
  - The main advantage of ansible is that you can automate configuration. This saves time for administrators who do not have to setup virtual machines manually. The automatic nature of ansible also helps reduce mistakes made because of normal human error.

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
  - download image (install python3_pip)
  - Increase memory
  - download configuration files and create playbooks
  - run playbooks

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)



```
root@JBOX:~# docker ps -a
CONTAINER ID   IMAGE                    COMMAND                  CREATED       STATUS                     PORTS     NAMES
ec40b128acac   cyberxsecurity/ansible   "/bin/sh -c /bin/bas…"   4 weeks ago   Exited (0) 5 seconds ago             determined_curie
root@JBOX:~#
```

Target Machines & Beats

This ELK server is configured to monitor the following machines:

- _TODO: List the IP addresses of the machines you are monitoring_

IP addresses of machines being monitored:

- 10.0.0.7

- 10.0.0.8

We have installed the following Beats on these machines:

- _TODO: Specify which Beats you successfully installed_
    - We successfully installed filebeat and metricbeat.

These Beats allow us to collect the following information from each machine:

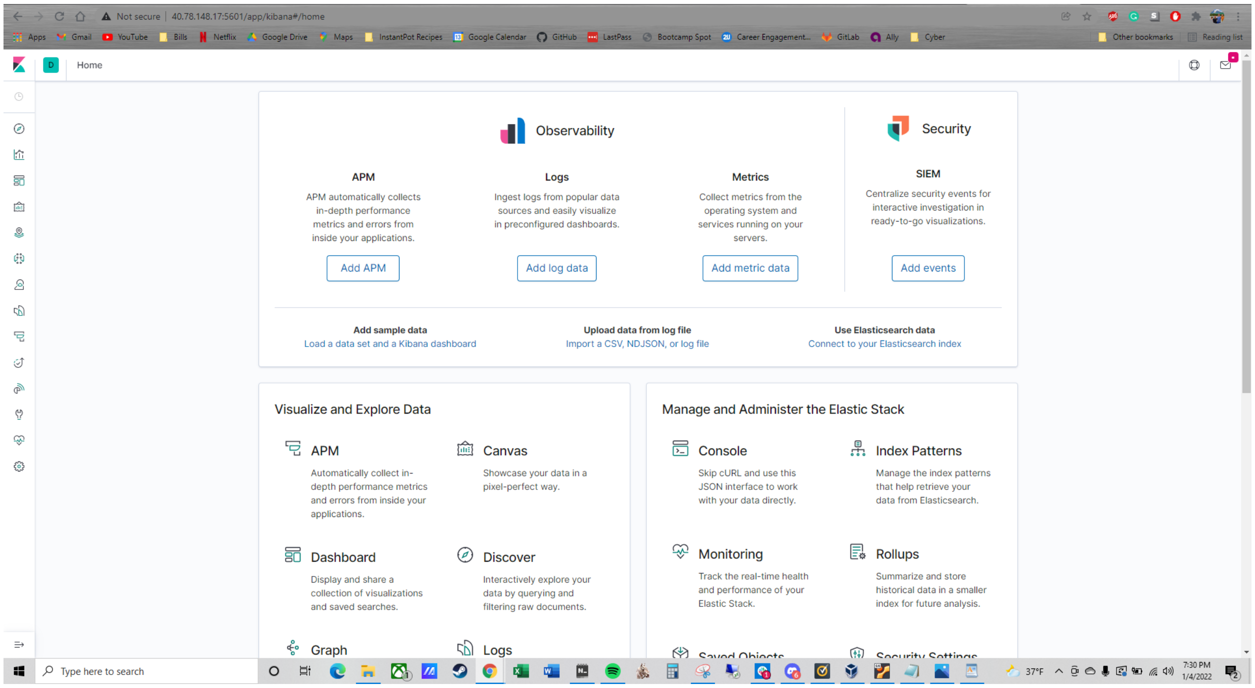
- _TODO: In 1-2 sentences, explain what kind of data each beat collects, and provide 1 example of what you expect to see. E.g., `Winlogbeat` collects Windows logs, which we use to track user logon events, etc._
    - Filebeat collects log files. We can collect logins to see who is using the system. It also collects audit logs, server logs, gc logs, and slow logs.
    - Metricbeat collects metrics and statistics from the system. We can track CPU usage and memory which can help detect malicious actors or software.

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 _____ file to include...
- Run the playbook, and navigate to ____ to check that the installation worked as expected.
    - Copy the elk_install.yml file to /etc/ansible/roles/elk_install.yml
    - Update the hosts file to include our groups and their IP’s. In this project we used the [webservers] group.
    - Run the playbook, and navigate to http://(your_elk_IP):5601/app/kibana to check if the installation worked. If it did, you should see something similar to the following:

- 

Observability

APM

APM automatically collects in-depth performance metrics and errors from inside your applications.

Logs

Ingest logs from popular data sources and easily visualize in preconfigured dashboards.

Metrics

Collect metrics from the operating system and services running on your servers.

Security

SIEM

Centralize security events for interactive investigation in ready-to-go visualizations.

Add sample data
Load a data set and a Kibana dashboard

Upload data from log file
Import a CSV, NDJSON, or log file

Use Elasticsearch data
Connect to your Elasticsearch index

Visualize and Explore Data

APM
Automatically collect in-depth performance metrics and errors from inside your applications.

Canvas
Showcase your data in a pixel-perfect way.

Dashboard
Display and share a collection of visualizations and saved searches.

Discover
Interactively explore your data by querying and filtering raw documents.

Graph

Logs

Manage and Administer the Elastic Stack

Console
Skip cURL and use this JSON interface to work with your data directly.

Index Patterns
Manage the index patterns that help retrieve your data from Elasticsearch.

Monitoring
Track the real-time health and performance of your Elastic Stack.

Rollups
Summarize and store historical data in a smaller index for future analysis.

Saved Objects

Security Settings

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?

- Copy the elk_install.yml file to /etc/ansible/roles/elk_install.yml
- Update the hosts file to include our groups and their IP’s. In this project we used the [webservers] group.
- Run the playbook, and navigate to http://(your_elk_IP):5601/app/kibana to check if the installation worked. If it did, you should see something similar to the following:

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

-