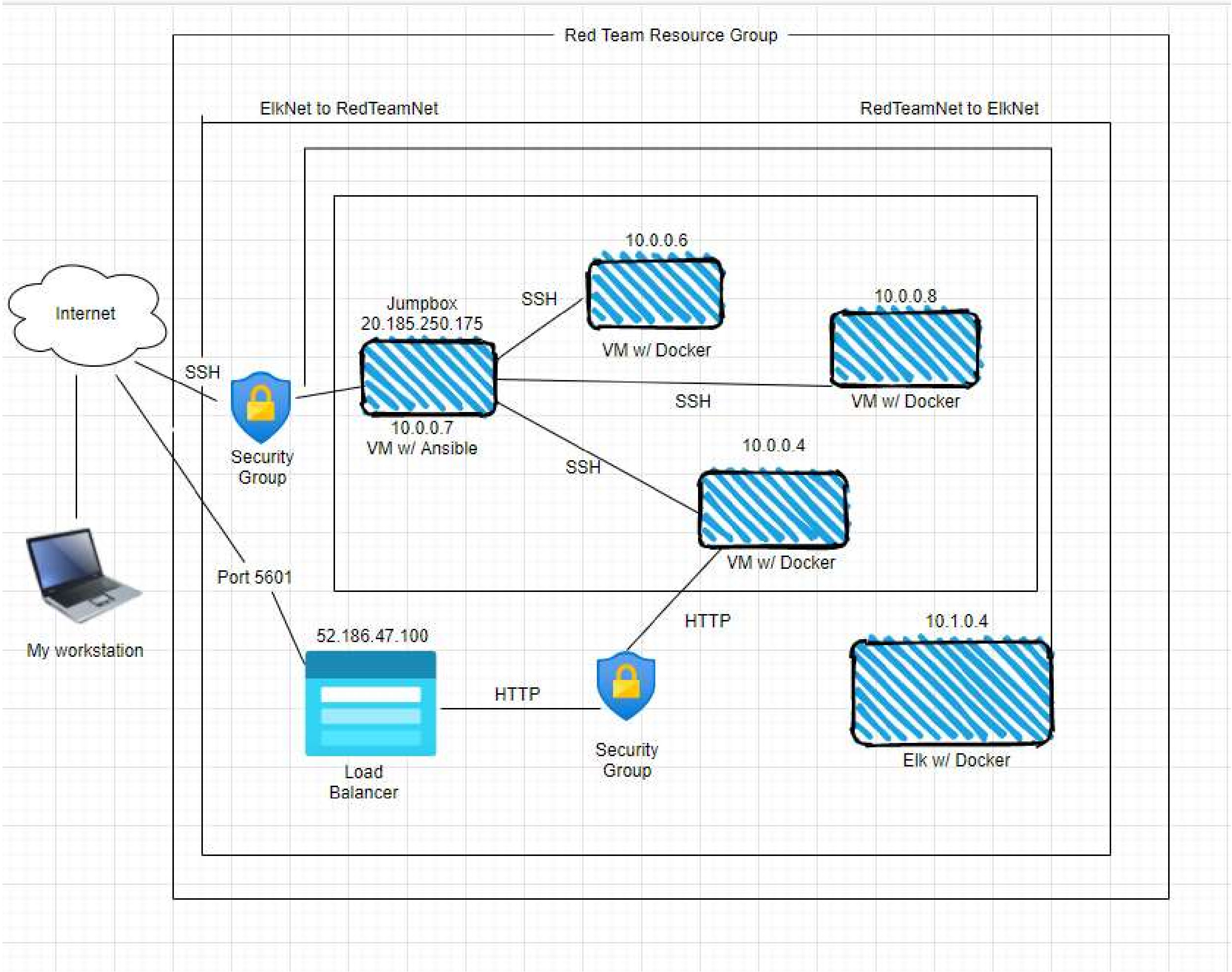
Automated Elk Stack Deployment

The files in this repository were used to configure the network depicted below. (Images/elkstack.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.

/etc/ansible/files/filebeat-playbook.yml

This document contains the following details:
-Description of the Topology
-Access Policies
-ELK Configuration
 -Beats in Use
 -Machines in Use
-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 the **inbound access** to the network. **The load balancer ensures that processing incoming traffic will be shared by all the vulnerable web servers. The jump box ensures that only authorized users (us) will be able to connect to the web servers.**

Integrating ELK servers allows users to easily monitor the vulnerable VMs for changes to the **file systems of the VMs on the network**, as well as watch **system metrics**, such as CPU usage, attempted ssh logins, sudo escalation failures, etc. **Filebeat watches for log files, collects log events and forwards them to Elasticsearch for indexing. Metricbeat records metrics and statistics of the operating system and services running on the server and ships the output to Elasticsearch.**

The configuration details for each machine may be found below:

Name	**IP Address**	**Operating System**
Jump Box Gateway	**10.0.0.7**	**Linux Ubuntu**
Web 2 Web Server	**10.0.0.6**	**Linux Ubuntu**
DVWA 2 Web Server	**10.0.0.4**	**Linux Ubuntu**
ELK Monitoring	**10.1.0.4**	**Linux Ubuntu**

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 address: **20.185.250.175**

Machines within the network can only be accessed by **each other. DVWA 2 and Web 2 VMs** send traffic to the ELK server.

A summary of the access policies can be found below:

Name	**Publicly Accessible?**	**Allowed IP Addresses**
Jump Box	**Yes**	**20.185.250.175**
ELK	**No**	**10.0.0.1-254**
Web 2	**No**	**10.0.0.1-254**
DVWA 2	**No**	**10.0.0.1-254**

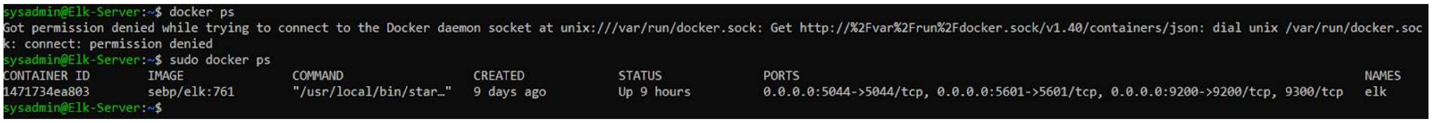
ELK Configuration

Ansible was used to automate configuration of the ELK machine. No configuration was performed manually, which is advantageous because **ansible helps administrators save time from daily, monotonous tasks and focus that time elsewhere for more important tasks for the organization.**

The playbook implements the following tasks:

-install docker
-install pip3
-install docker python module
-increase virtual memory
-use more memory
-download and launch a docker elk container

The following screenshot displays the result of running “docker ps” after successfully configuring the ELK instance:



```
sysadmin@Elk-Server:~$ docker ps
Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get http://%2Fvar%2Frun%2Fdocker.sock/v1.40/containers/json: dial unix /var/run/docker.soc
k: connect: permission denied
sysadmin@Elk-Server:~$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS                                                                              NAMES
1471734ea803        sebp/elk:761        "/usr/local/bin/star…"   9 days ago          Up 9 hours          0.0.0.0:5044->5044/tcp, 0.0.0.0:5601->5601/tcp, 0.0.0.0:9200->9200/tcp, 9300/tcp   elk
sysadmin@Elk-Server:~$
```

Target Machines and Beats

The ELK server is configured to monitor the following machines, **Web 2 and DVWA 2 at 10.0.0.6 and 10.0.0.4** respectively.

We have installed the following beas on the machines:

Filebeat - Filebeat detects changes to the file system. We specifically use it to collect Apache logs.

Using the Playbook

In order to use the playbook, you will need to have a jump box already configured. Assuming you have such a control node provisioned:

SSH into the control node and follow the steps below:
-copy the **playbooks** to the **ansible control node**
-run each **playbook** and navigate to the **appropriate targets** to check the installation has worked as expected

Which file is the playbook? Where do you copy it?
The file that ends in .yml is the playbook. It is copied from Git.

Which file do you update to make ansible run a playbook on a specific machine? How do I specify which machine to run the ELK server on versus which to install Filebeat on?
You must create a "hosts" file to specify which machines to run each playbook and then run these commands:
ansible-playbook install_elk.yml elk
ansible-playbook install_filebeat.yml webservers

Which URL do you navigate to in order to check that the ELK server is running.
curl http://10.1.0.4:5601/app/kibana