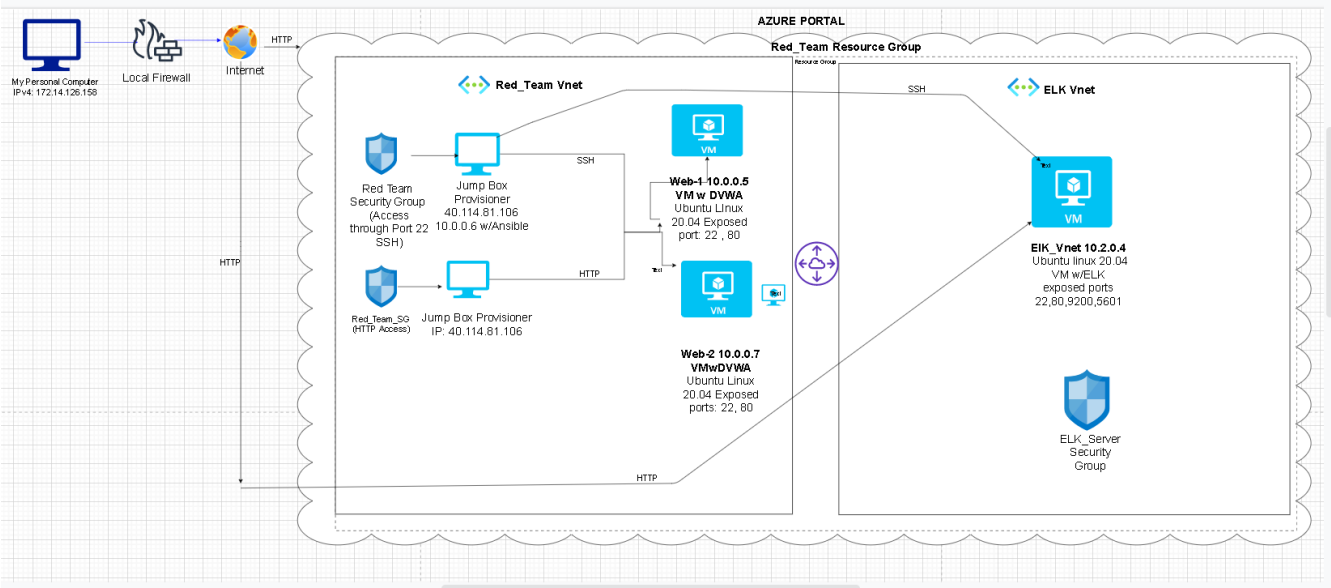
Automated ELK Stack Deployment

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

```
root@6b2eb65858a0:/etc/ansible# cat my_playbook.yml
---

  - name: My first playbook
    hosts: webservers
    become: true
    tasks:

    - name: docker.io
      apt:
        update_cache: yes
        name: docker.io
        state: present

    - name: Install pip3
      apt:
        name: python3-pip
        state: present

    - name: Install Python Docker module
      pip:
        name: docker
        state: present

    - name: download and launch a docker web container
      docker_container:
        name: dvwa
        image: cyberxsecurity/dvwa
        state: started
        published_ports: 80:80

    - name: Enable docker service
      systemd:
        name: docker
        enabled: yes


root@6b2eb65858a0:/etc/ansible#
```

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 access to the network.

-Load balancers protect against distributed denial-of-service attacks by rerouting traffic to other available servers when a specific server becomes overloaded.

-Jump box provide a single point of connection to the network so other machines on the network will not be exposed to public internet. It will limit the security settings to a single machine instead of multiple machines such as

- Implementing a firewall host on jump box
- Implementing log monitoring
- Implementing two-factor authentication for SSH login to the jump box.

Integrating an ELK server allows users to easily monitor the vulnerable VMs for changes to the file system and system metrics.
- Filebeat watch for and collect log files from very specific files such as Microsoft Azure tools. Nginx web server, Apache or MYSQL databases.
- Metricbeat record metrics and statistics for a system or services such as CPU

The configuration details of each machine may be found below.

Name	Function	IP Adress	Operating System
Jump Box	Gateway	10.0.0.6	Linux
Web-1	Web Server	10.0.0.5	Linux
Web-2	Web Server	10.0.0.7	Linux
ElK_Server	ELK Server	10.2.0.4	Linux
Red_Team_LB	Load-balancer	52.188.121.141	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:
- My personal IP Address

Machines within the network can only be accessed by SSH.
- The ELK.server is only accessible by SSH from the Jump Box at IP 10.0.0.6 and via web access from my personal 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	10.0.0.6/Personal IP
Web-1	No	10.0.0.5
Web-2	No	10.0.0.7
ELK.Server	Yes	10.0.0.6/Personal IP

Elk Configuration

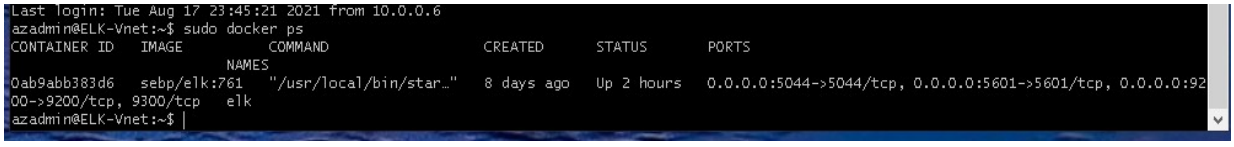
Ansible was used to automate configuration of the ELK machine. No configuration was performed manually, which is advantageous because...

- Allows for a consistent configuration. You can deploy multiple servers easily and quickly

The playbook implements the following tasks:

- Install 'docker.io' and 'python3-pip' packages with 'apt'module
- Install docker 'python' package with 'pip'
- Increase memory with 'sysctl' module
- Enable 'systemd'docker service
- Run ELK docker container

The following screenshot displays the result of running `docker ps` after successfully configuring the ELK instance.



Target Machines & Beats

This ELK server is configured to monitor the following machines:

- Web-1: 10.0.0.5
- Web-2: 10.0.0.7

We have installed the following Beats on these machines:
- Filebeat
- Metricbeat

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

- Filebeat monitors the log files or locations that you specify, collects log events, and forwards them either to Elasticsearch or Logstash for indexing.
- Metricbeat helps you monitor your servers by collecting metrics from the system and services running on the server, such as: Apache. HAProxy.

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 install-elk.yml playbook file to /etc/ansible/roles/ directory inside the ansible container.
- Update the configuration files to include the Private IP of the ELK-Server to the ElasticSearch and Kibana Sections of the Configuration File
- Run the playbook, and navigate to Eld-Server-Public:5601/app/kibana to check that the installation worked as expected.

Answer the following questions to fill in the blanks:

- The playbook file is located under ansible directory. This can be located by opening gitbash on your computer ssh@40.114.81.106 then starting and attaching your docker then /etc/ansible.

- To update the Ansible file for it to run on a specific machine you update the configure file with the IP address of the machine you want it to run on.

- The URL to see if the ELK server is working is the HTTP://20.75.19.168:5601/app/kibana