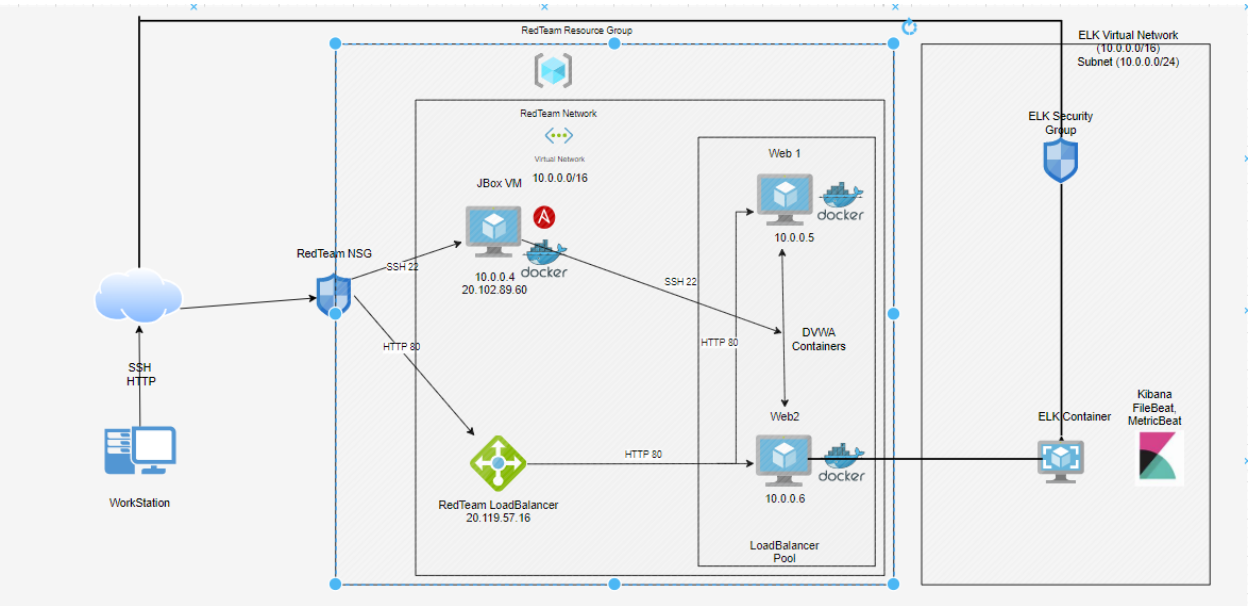
Git Fundamentals

Automated ELK Stack Deployment
The files in this repository were used to configure the network depicted below.



They can be used to either recreate the entire deployment picture. Alternatively, select portions of the yml and config file may be used to install only certain pieces of it, such as Filebeat.
Ansible folder has the below details:

- Hosts
- Ansible Configuration
- Ansible ELK Installation and VM Configuration
- Filebeat Config
- Metric beat Config
- Metricbeat Playbook

This document contains the following details:

- Description of the Topology
- Access Policies
- ELK Configuration
- Target Machines and Beats
- Playbook

Description of the Topology
The main purpose of this network is to expose a load-balanced and monitored instance of DVWA, Load balancing ensures that the application will be highly functional and available, in addition to restricting traffic to the network.

The Load balancers add resiliency by rerouting live traffic from one server to another if a server falls prey to a DDoS attack or otherwise becomes unavailable.

A Jump Box Provisioner prevents Azure VMs from being exposed via a public IP Address. This allows us to do monitoring and logging on a single box. We can also restrict the IP addresses able to communicate with the Jump Box. Integrating an ELK server allows users to easily monitor the vulnerable VMs for changes to the network and system logs.

Filebeat monitors the log files or locations that we specify, collects log events, and forwards them either to Elasticsearch or Logstash for indexing. Metric beat takes the metrics and statistics that it collects and ships them to the output that you specify, such as Elasticsearch or Logstash.

The configuration details of each machine may be found below.

Name	Function	IP Address	Operating System
Jump Box	Gateway	10.0.0.4/ 20.102.89.60	Linux
Web 1	Ubuntu Server	10.0.0.5/ 52.249.182.156	Linux
Web 2	Ubuntu Server	10.0.0.6/ 52.249.182.156	Linux
ELKVM	Ubuntu Server	10.1.0.4/ 20.62.163.231	Linux

Access Policies

The machines on the internal network are not exposed to the public Internet.
Only the Jump-Box-Provisioner machine can accept connections from the Internet. Access to this machine is only allowed from the below IP addresses. Machines within the network can only be accessed by Workstation and Jump-Box-Provisioner through SSH Jump-Box.

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

Name	Publicly Accessible	Allowed IP address
Jump Box	Yes	20.102.89.60 / SSH 22
Web -1	No	10.0.0.4 / SSH 22
Web -2	No	10.0.0.5 / SSH 22
ELKVM	No	Public IP TCP 560

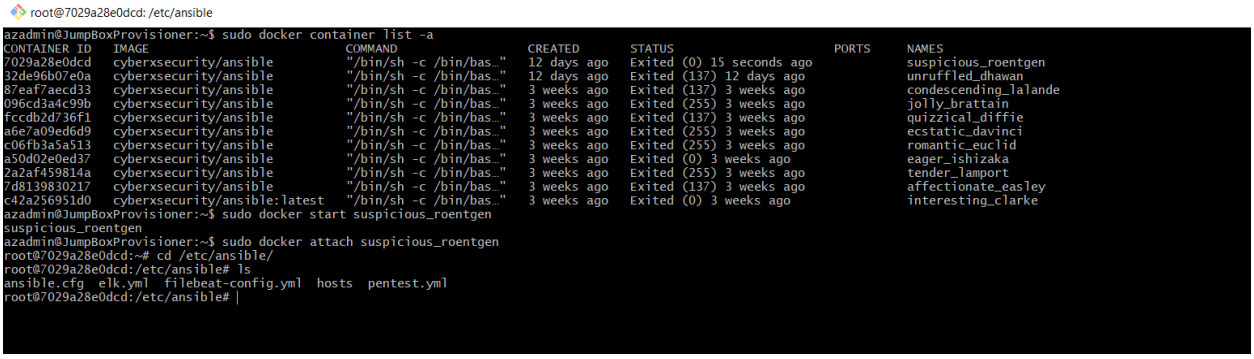
Elk Configuration

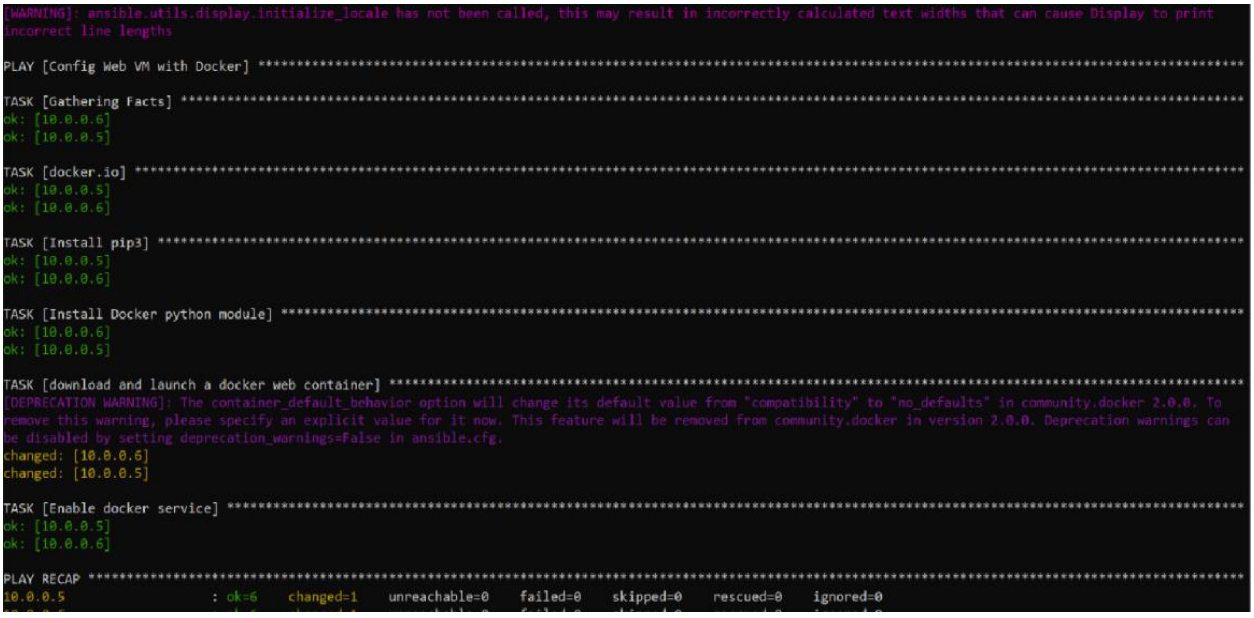
Ansible was used to automate configuration of the ELK machine. Ansible lets you quickly and easily deploy multitier applications through a YAML playbook. Ansible will also figure out how to get your systems to the state you want them to be in.

The playbook implements the following tasks:

- Config ELK VM with a Docker
- Install Docker
- Install Python with pip command
- Install Docker module
- Initialize / Increase memory
- Download and Launch ELK Container
- Published ports

The following screenshot displays the result of running `docker ps` after successfully configuring the ELK instance. Connect to jump-Box-Provisioner VM





Target Machines & Beats

This ELK server is configured to monitor the following machines:
List the IP addresses of the machines you are monitoring

- Web-1: 10.0.0.5
- Web-2: 10.0.0.6
- DVWA-VM3: 10.0.0.7

Using the Playbook

To use the playbook, you will need to have an Ansible control node already configured
SSH into the control node and follow the steps below:

- Copy the yml file to ansible folder.
- Update the config file to include remote users and ports.
- Run the playbook and navigate to check that the installation worked as expected.