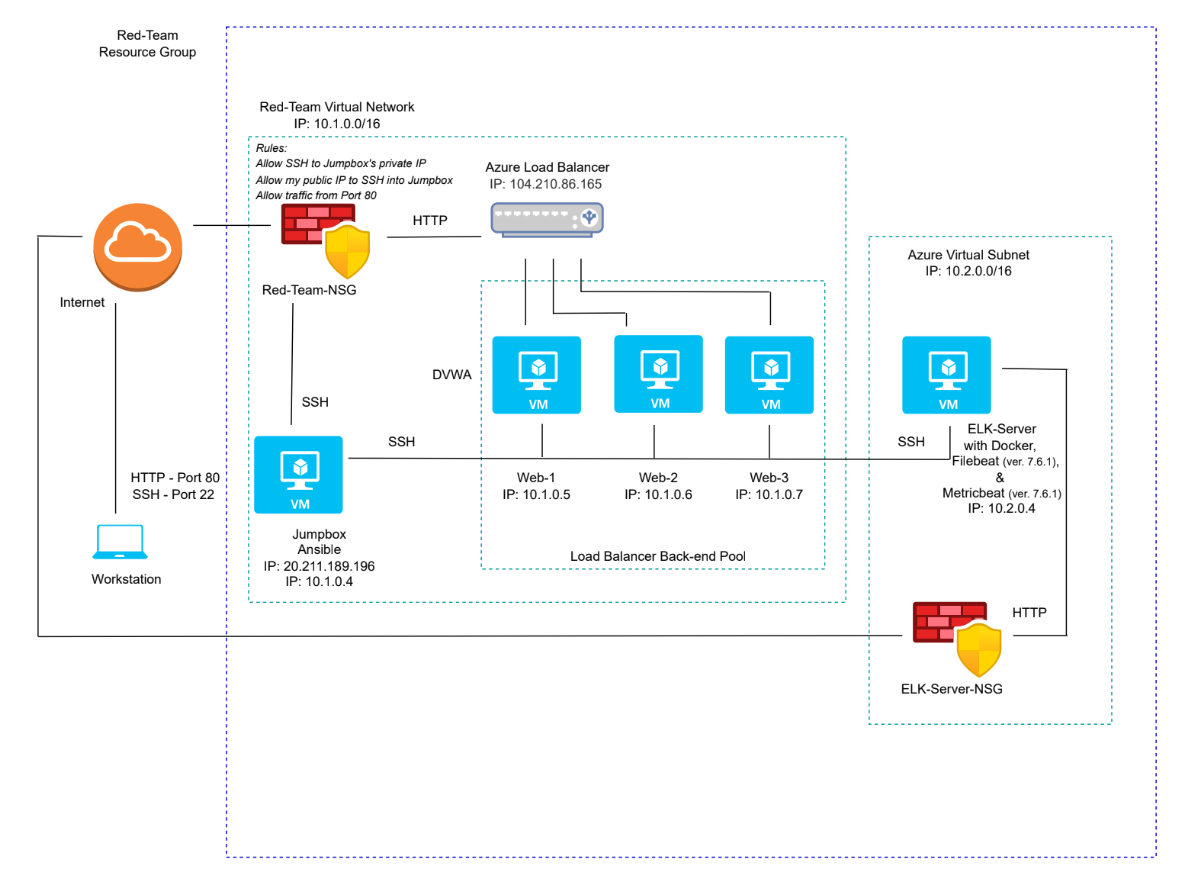
## **Automated ELK Stack Deployment**

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 Ansible-Playbook.yml file may be used to install only certain pieces of it, such as Filebeat.

* ELK Playbook
* Filebeat Playbook
* Metricbeat Playbook

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 inbound traffic to the network.

* Load balancers alleviate potentially overwhelming live traffic between servers and improve resiliency by distributing traffic between servers. By rerouting user requests between multiple servers, the load balancers can prevent DoS attacks. An additional method of protection to the network is the Jumpbox, since it is the only access point to the network using a specific IP address.

Integrating an ELK server allows users to easily monitor the vulnerable VMs for changes to the network files and monitor system metrics.

* Filebeat collects and monitors log files (often generated by Apache, Microsoft Azure tools, MySQL, etc.), collects log events, and forwards them either to Elasticsearch or Logstash for indexing
* Metricbeat takes the metrics and statistics that it collects from the OS and services (such as Apache, MySQL, Nginx, etc.) and ships them to an output, such as Elasticsearch or Logstash.

The configuration details of each machine may be found below.

| **Name** | **Function** | **IP Address** | **Operating System** |
| --- | --- | --- | --- |
| Jumpbox | Gateway | 10.1.0.4 | Linux |
| Web-1 | Web-Server | 10.1.0.5 | Linux |
| Web-2 | Web-Server | 10.1.0.6 | Linux |
| Web-3 | Web-Server | 10.1.0.7 | Linux |
| ELK-Server | Monitoring | 10.2.0.4 | 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:

* My personal IP address

Machines within the network can only be accessed by a SSH connection from the Jumpbox VM.

* The Jumpbox can access the webservers and the ELK-Server VM via SSH with the Jumpbox’s public IP address

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

**Elk Configuration**

| **Name** | **Publically Accessible** | **Allowed IP Addresses** |
| --- | --- | --- |
| Jumpbox | Yes | Personal IP address  10.1.0.4 |
| Web-1 | No | 10.1.0.0/16 |
| Web-2 | No | 10.1.0.0/16 |
| Web-3 | No | 10.1.0.0/16 |
| ELK-Server | No | 10.2.0.0/16 |

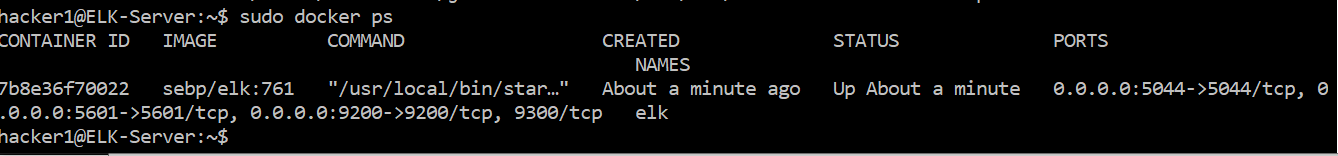
Ansible was used to automate configuration of the ELK machine. No configuration was performed manually, which is advantageous because...

* Ansible can automate configurations through YAML files
* Ansible is free and has an accessible guide for its language

The playbook implements the following tasks:

* Identifies host and remote user
* Installs docker.io
* Installs python3-pip
* Installs docker module
* Increases virtual memory
* Downloads and launches a docker container with a restart policy
* Enables service docker on boot
* Enables ports (5601, 9200, 5044) for availability

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:

* 10.1.0.5
* 10.1.0.6
* 10.1.0.7

We have installed the following Beats on these machines:

* Filebeat
* Metricbeat

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

* Filebeat collects log files and system data, such as Web Log Data
* Metricbeat collects machine metrics, such as uptime, CPU usage, successful and failed commands

### **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 configuration file from the Ansible container to the Web VMs
* Update the etc/ansible/hosts file to include the IP address(es) of the Web VMs and the ELK server
  + [webservers]

10.1.0.5

10.1.0.6

10.1.0.7

[elk]

10.2.0.4

* Run the playbook, and navigate to *http://[Elk\_VM\_Public\_IP]:5601/app/kibana* to check that the installation worked as expected
* Which file is the playbook?
  + ELK-Playbook
  + Filebeat
  + Metricbeat Playbook
* Where do you copy it?
  + copy /etc/ansible/filebeat-config.yml to /etc/filebeat/filebeat.yml
  + copy /etc/ansible/metricbeat-config.yml to /etc/metric/metricbeat.yml
* Which file do you update to make Ansible run the playbook on a specific machine?
  + filebeat-config.yml
  + metricbeat-config.yml
* How do I specify which machine to install the ELK server on versus which to install Filebeat on?
  + The etc/ansible/hosts file lists two groups: webservers and elk
* Which URL do you navigate to in order to check that the ELK server is running?
  + *http://[Elk\_VM\_Public\_IP]:5601/app/kibana*