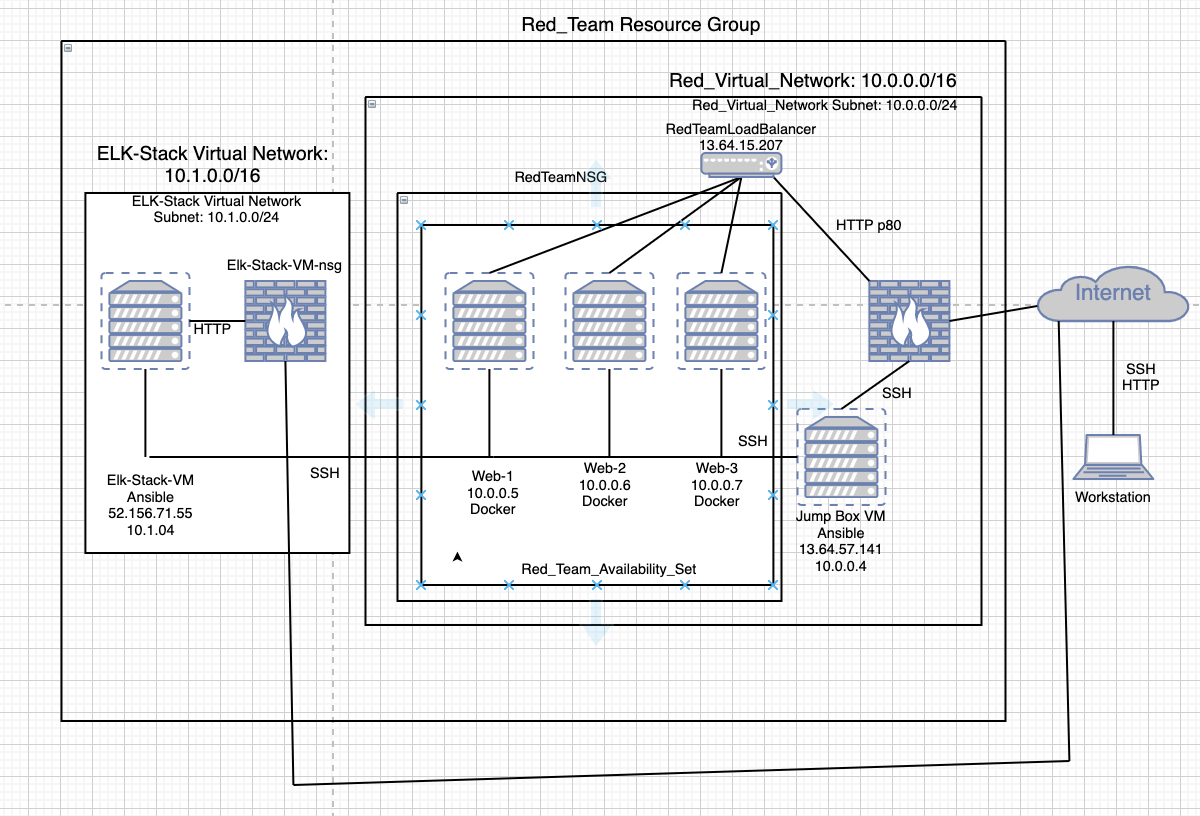
**Project 1: README**

**Automated ELK Deployment**

The following files in the repository (/etc/ansible/) within Ansible were used to configure the network depicted below.

* Elkplaybook.yml
* filebeat-playbook.yml



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, or select portions to install only certain pieces of it, such as Filebeat.

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 balancers ensure a network is protected against Denial of Service (DoS) attacks by balancing web traffic across multiple servers, preventing one server from shutting down availability to the website.

This ensures that the application will be highly redundant and fault tolerant, in addition to restricting access to the network.

As an added security precaution, I created a Jump Box virtual machine that acts as a gateway router. The Jump Box VM forces all web traffic through a single node, requiring the administrator to fan in, or secure and monitor this one node. The added benefit of the Jump Box is that it allows the administrator to focus on the few connections between a few machines rather than connections between all machines.

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

Within the ELK server, I employed Filebeat. Filebeat is a tool that collects data about the file system. It helps generate and organize log files to send to Logstash and Elasticsearch.

Since our machines are able to use up to eight different Beats suite tools, we can download others as well. For example, we could use Metricbeat, which collects machine metrics, such as uptime or CPU usage, that measure an aspect of a system and lets analysts know how “healthy it is.