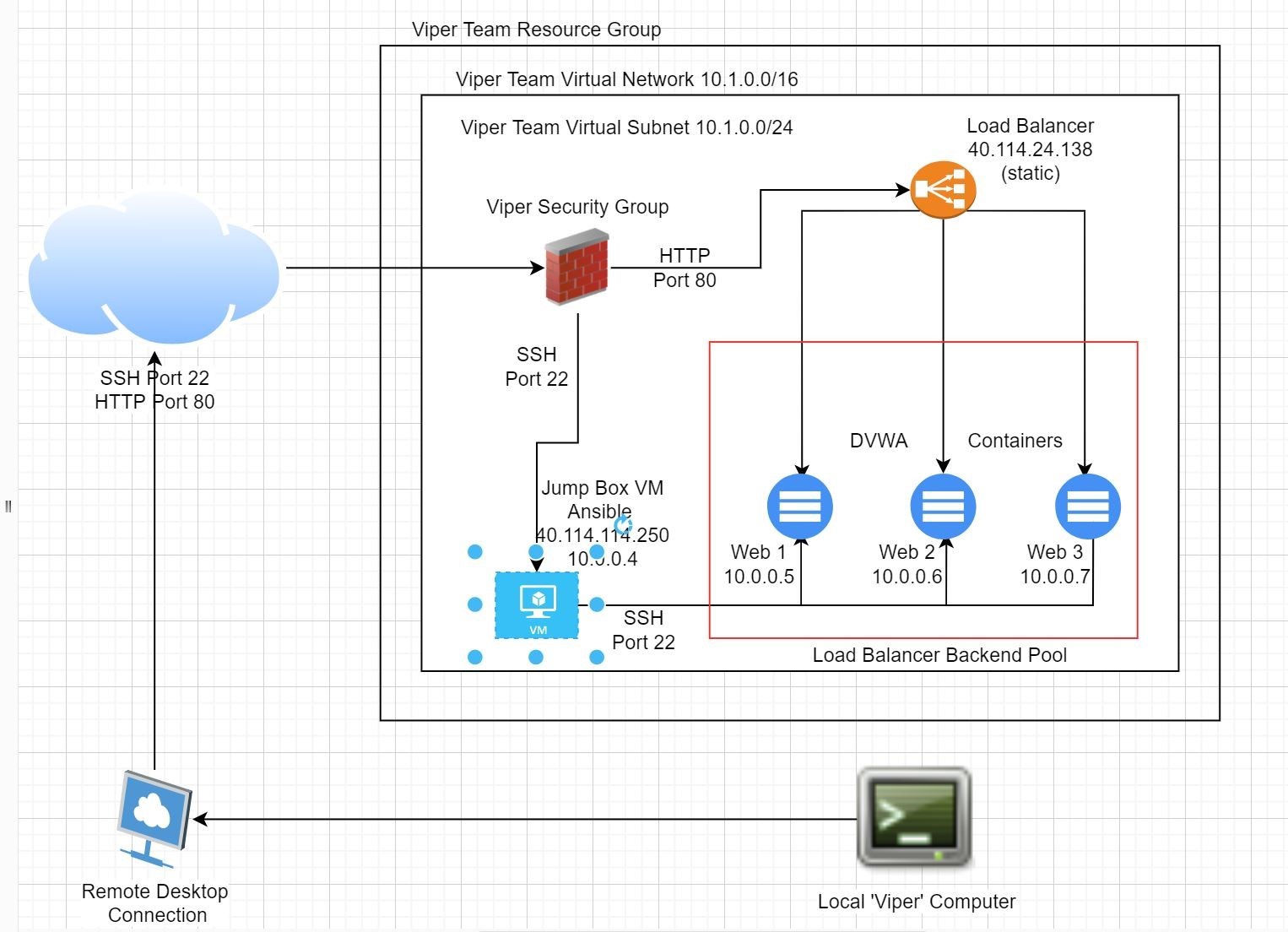
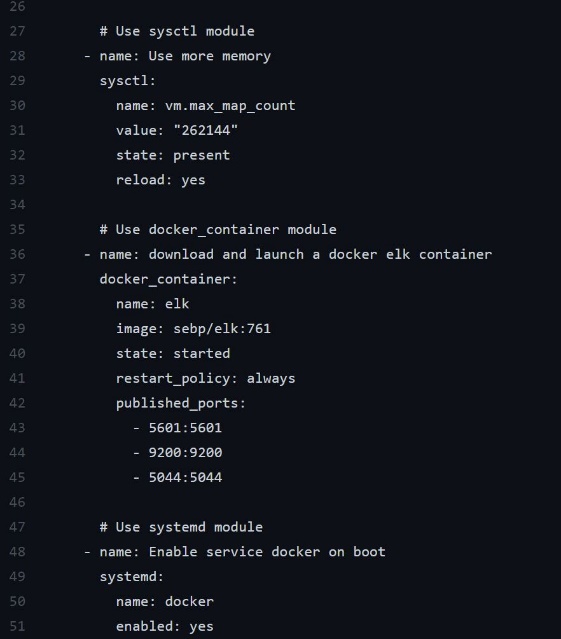
Week 13 homework assignments The files in this repository were used to configure the network depicted



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 yml and config file may beused to install only certain pieces of it, such as Filebeat

Text

Description automatically generated

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 versatile, in addition to restricting high traffic to the network.

Load balancers are useful when there is a DoDOs attack or if there's a lot of traffic another server can substitute or alleviate it. Integrating an ELK server allows users to easily monitor the vulnerable VMs for changes to the data system logs.

Filebeat watches and monitors the log files or locations that users specify, collects log events, and forwards them either to Elasticsearch or Logstash for indexing. Metricbeat Collect metrics from your systems and services. From CPU to memory, Redis to NGINX, and much more, It is a lightweight way to send system and service statistics.

The configuration details of each machine may be found below.

Name Function IP Address Operating System

Jump Box Gateway 10.0.0.15 Linux

Web-1 DVWA 10.0.0.16 Linux

Web-2 DVWA 10.0.0.17 Linux

Web-3 DVMA 10.1.0.5 Linux

The machines on the internal network are not exposed to the public Internet.

Only the ELK Server machine can accept connections from the Internet. Access to this machine is only allowed from the following IP addresses:

Name Publicly Accessible Allowed IP Address

JumpBox Yes 20.102.79.41

Elk-Server Yes 13.64.110.13

Web-1 No 10.0.0.15

Web-2 No 10.0.0.15

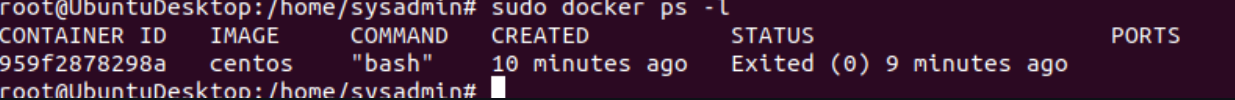
Web-3 No 10.1.0.4

Elk Configuration Ansible was used to automate configuration of the ELK machine. No configuration was performed manually, which is advantageous because there aren’t any custom codes needed. You can just list all the required tasks using a playbook, ansible will do the rest.

The advantage of automated configuration with Ansible is that it increases efficiency when configuring multiple machines.

* install Docker
* download image etc.
* Install Docker.io
* Install Docker module
* Increase Virtual Memory
* Download and launch elk container
* Enable elk ports
* Downloads and launches the sebp/elk container over ports 5601, 9200, and 5044

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: 10.0.0.15, 10.0.0.16, 10.0.0.17 We have installed the following Beats on these machines:

Winlogbeat collects Windows logs, which we use to track user logon events, etc. Filebeat collects log events, and forwards the information to Elasticsearch or Logstash for indexing. Metricbeat collects system metrics from the operating system as well as system services.

In order to use the playbooks, you will need to have an Ansible control node VM already configured, along with VMs for the webservers Web-1 & Web-2 and ELK. Assuming you have such a control node provisioned.

SSH into the control node and follow the steps below:

Copy the ansible.cfg, hosts, WebDVWA-playbook.yml, and install-elk.yml files to /etc/ansible.

Update the ansible.cfg file to set the remote user, update the hosts section name in the hosts file for [elk] and [webservers] with the target IP addresses and the location of Python on the ELK and Web servers that you want to run Beats on, and update install-elk.yml with the the remote user to match ansible.cfg.

Run the WebDVWA-playbook.yml with "ansible-playbook WebDVWA-playbook.yml"

Run the install-elk.yml playbook with "ansible-playbook install-elk.yml" and navigate to the Kibana web page (http://ElkIPaddress:5601/app/kibana) to check that the installation worked as expected.

Copy the filebeat-config.yml and metricbeat-config.yml to /etc/ansible and update the Elasticserch output hosts field and Kibana host field to the IP address of the ELK server. Copy the filebeat-playbook.yml and metricbeat-playbook.yml to /etc/ansible and from there run "ansible-playbook filebeat-playbook.yml" and "ansible-playbook metricbeat-playbook.yml" so that ELK can get logs and metrics from the web servers Web-1 and Web-2.