## Automated ELK Stack Deployment

The files in this repository were used to configure the network depicted below.

Images/NetworkDiagram.JPG

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 playbook files may be used to install only certain pieces of it, such as Filebeat.

- elk-playbook.yml

- filebeat-playbook.yml

- metricbeat-playbook.yml

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 reliable, in addition to restricting traffic to the network.

Load balancers protect the availability of resources. The advantages of a jump box is that it reduces risk by establishing a secure computer where users can manage and access devices.

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

Filebeat monitors and collects log files and forwards them to Elasticsearch for indexing.

Metricbeat records metrics and statistics from the operating system as well as services running on the server.

The configuration details of each machine may be found below.

| Name | Function | IP Address | Operating System |

|----------|----------|------------|------------------|

| Jump Box | Gateway | 10.0.0.4 | Linux (Ubuntu 18.04)

| Web-1 | DVWA | 10.0.0.5 | Linux (Ubuntu 18.04)

| Web-2 |DVWA | 10.0.0.6 |Linux (Ubuntu 18.04)

| Web-3 |DVWA | 10.0.0.7 |Linux (Ubuntu 18.04)

|Elk-Server| Monitor |10.1.0.4 |Linux (Ubuntu 18.04)

### 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 address:

- 38.94.247.28

Machines within the network can only be accessed by the Ansible container on the Jump Box virtual machine.

- The Jump Box VM is able to access the Elk VM from the Ansible container. Jump Box IP:10.0.0.4 Ansible container IP: 172.17.0.2

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

| Name | Publicly Accessible | Allowed IP Addresses |

|-----------------|---------------------|----------------------|

| Jump Box SSH | Yes | 38.94.247.28 |

| SSH-from-JumpBox| No | 10.0.0.4 | | HTTP Traffic | Yes | 38.94.247.28

### Elk Configuration

Ansible was used to automate configuration of the ELK machine. No configuration was performed manually, which is advantageous because it eliminates potential human errors when configuring manually and assures the configuration is the same as the other VMs.

The main advantage of automating configuration with Ansible is that it allows you to quickly configure multiple machines with the same configuration.

The playbook implements the following tasks:

In 3-5 bullets, explain the steps of the ELK installation play. E.g., install Docker; download image; etc.

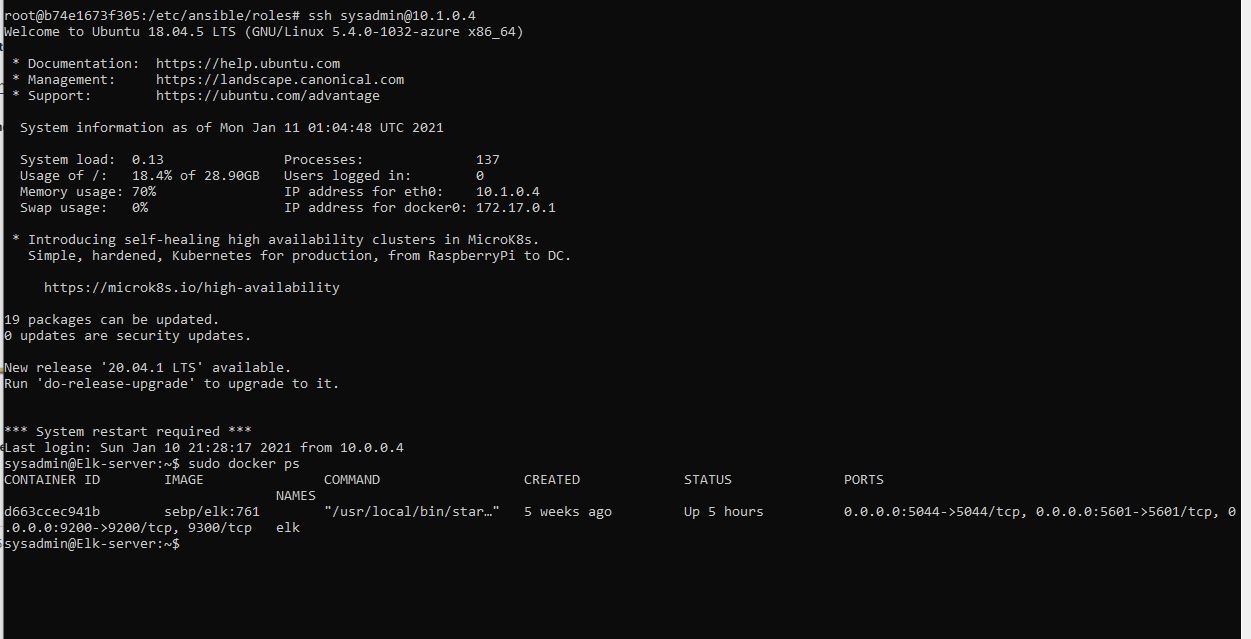
- Installs docker.io and python3-pip

- Downloads docker container called sebp/elk:761

- Increases virtual memory

- Configures the container to start with specified port mappings

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



Images/sebp\_elk761container

### Target Machines & Beats

This ELK server is configured to monitor the following machines:

- Web-1 10.0.0.5

- Web-2 10.0.0.6

- Web-3 10.0.0.7

We have installed the following Beats on these machines:

- filebeat-7.4.0-amd64.deb

- metricbeat-7.4.0-amd64

These Beats allow us to collect the following information from each machine: Web-1, Web-2, Web-3.

* Filebeat sends event logs to the elk-server using a harvester, which reads the content of a single file, and an input, which manages the harvesters and finds all sources to read from.
* Metricbeat collects metrics from the operating system and services running on the server such as: CPU usage, memory, file system, disk IO, and network IO statistics.

### 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 Filebeat configuration file to /etc/ansible/files/filebeat-config.yml.

- Update the filebeat-config.yml file to include your machine’s IP address.

- Run the playbook, and navigate to the Filebeat installation page on the machine’s GUI to check that the installation worked as expected.

- elk-playbook.yml is the playbook to install the ELK server found in /etc/ansible/

- Update the hosts file to include the machine’s IP so it can be specified in a playbook. For the elk-playbook.yml file, update the hosts line to the group that was added in the hosts file (ELK). In the filebeat-playbook.yml update the hosts line to the group that the webservers are in.

- Go to [http://](about:blank)20.51.124.128:5601/app/kibana to check that the ELK server is running.

To run the playbook enter: ansible-playbook (name of yml playbook)

To update the files enter: nano (name of file to update)