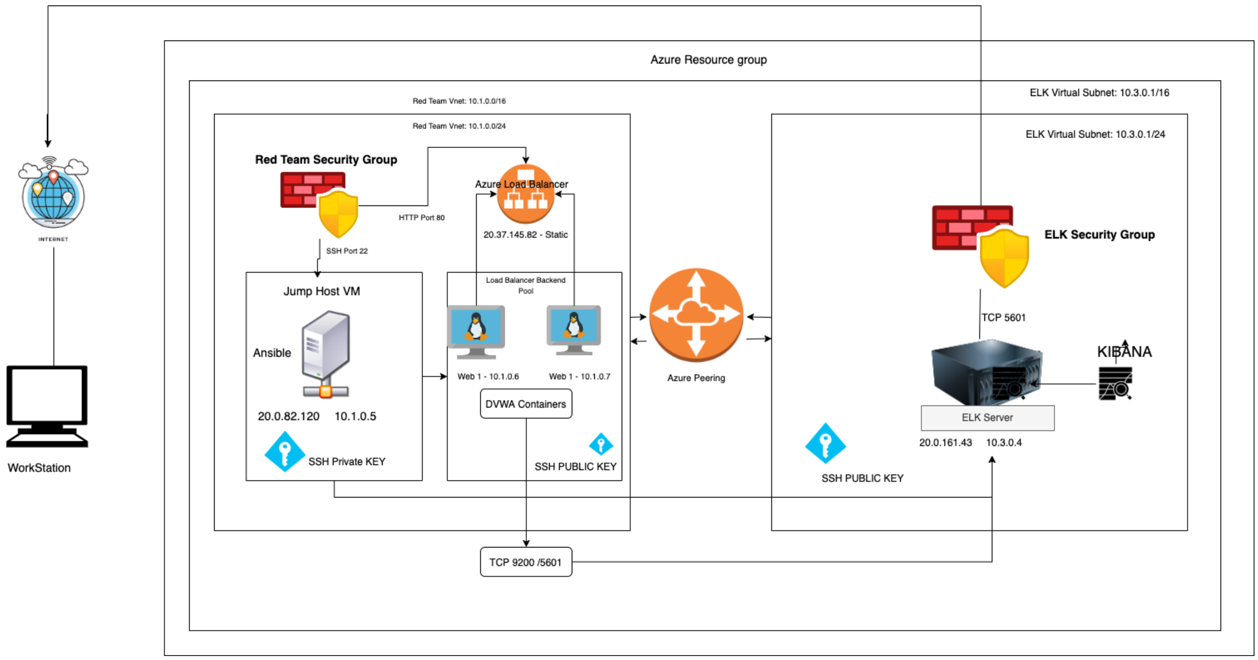
Automated ELK Stack Deployment

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

Note: The following image link needs to be updated. Replace diagram_filename.png with the name of your diagram image file.

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 ______ file may be used 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 balancing ensures that the application will be highly , in addition to restricting to the network.

> *What aspect of security do load balancers protect? What is the advantage of a jump box? They prevent unwanted or unauthorized traffic from being exposed.*

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

- *What does Filebeat watch for? Filebeat watches for log files or log events.*
- *What does Metricbeat record? Metric beat records metrics from from on going services on the server*

The configuration details of each machine may be found below. *Note: Use the* *Markdown Table Generator* *to add/remove values from the table*.

Name	Function	IP Address	Operating System
Jump Box	Gateway	10.0.0.5	Linux
WEB-1	DVWA	10.1.0.6	Linux
WEB-2	DVWA	10.0.0.7	Linux
ELK	LOG MONITORING	10.3.0.4	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 following IP addresses:

- *Add whitelisted IP addresses 20.0.82.120 , 20.37.145.82*

Machines within the network can only be accessed by Jump Box

- *Which machine did you allow to access your ELK VM? 10.0.0.5*

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

Name	Publicly Accessible	Allowed IP Addresses
Jump Box	No	20.0.82.120
WEB-1,2	Yes	Web LB 20.37.145.82
Web LB	Yes	
Elk	Yes	
Elk	Yes	

Elk Configuration

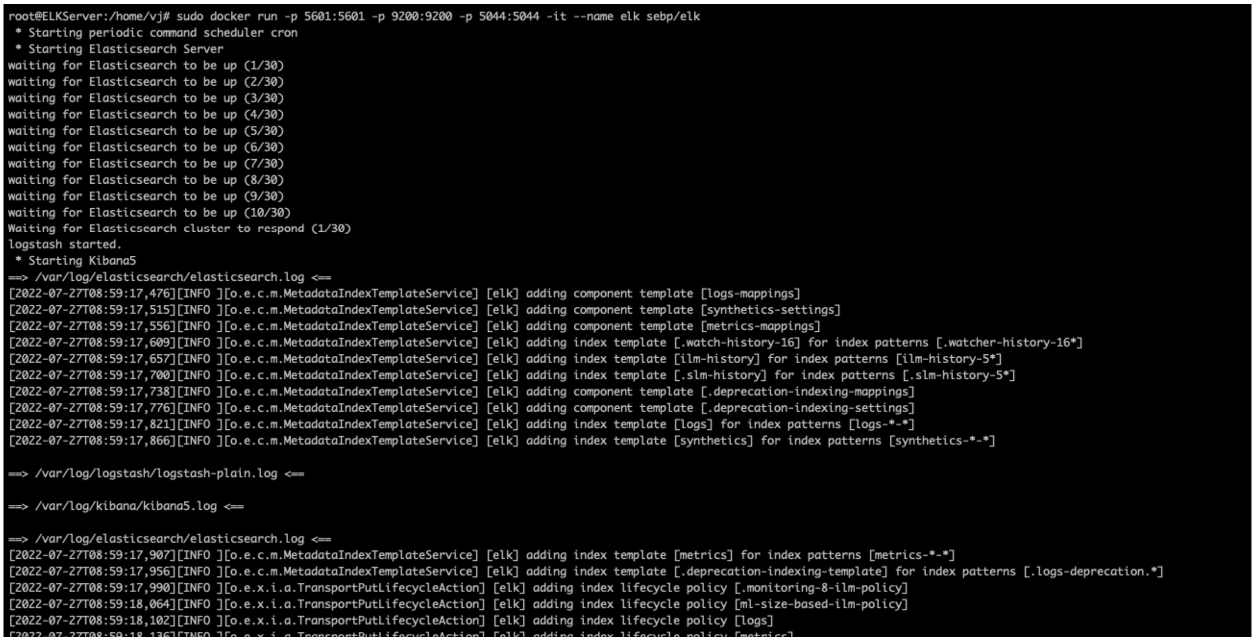
Ansible was used to automate configuration of the ELK machine. No configuration was performed manually, which is advantageous because...

- *What is the main advantage of automating configuration with Ansible? It is flexible, it allows changes to be made within any of the VM's*

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.*
- ...1. Install Docker.io
- …2.Install python3-pip
- 3.Install Docker Python Module

- 4. Launch a Docker w



```
root@ELKServer:/home/vj# sudo docker run -p 5601:5601 -p 9200:9200 -p 5044:5044 -it --name elk sebp/elk
 * Starting periodic command scheduler cron
 * Starting Elasticsearch Server
waiting for Elasticsearch to be up (1/30)
waiting for Elasticsearch to be up (2/30)
waiting for Elasticsearch to be up (3/30)
waiting for Elasticsearch to be up (4/30)
waiting for Elasticsearch to be up (5/30)
waiting for Elasticsearch to be up (6/30)
waiting for Elasticsearch to be up (7/30)
waiting for Elasticsearch to be up (8/30)
waiting for Elasticsearch to be up (9/30)
waiting for Elasticsearch to be up (10/30)
Waiting for Elasticsearch cluster to respond (1/30)
logstash started.
 * Starting Kibana5
==> /var/log/elasticsearch/elasticsearch.log <==
[2022-07-27T08:59:17,476][INFO ][o.e.c.m.MetadataIndexTemplateService] [elk] adding component template [logs-mappings]
[2022-07-27T08:59:17,515][INFO ][o.e.c.m.MetadataIndexTemplateService] [elk] adding component template [synthetics-settings]
[2022-07-27T08:59:17,556][INFO ][o.e.c.m.MetadataIndexTemplateService] [elk] adding component template [metrics-mappings]
[2022-07-27T08:59:17,609][INFO ][o.e.c.m.MetadataIndexTemplateService] [elk] adding index template [.watch-history-16] for index patterns [.watcher-history-16*]
[2022-07-27T08:59:17,657][INFO ][o.e.c.m.MetadataIndexTemplateService] [elk] adding index template [ilm-history] for index patterns [ilm-history-5*]
[2022-07-27T08:59:17,700][INFO ][o.e.c.m.MetadataIndexTemplateService] [elk] adding index template [.slm-history] for index patterns [.slm-history-5*]
[2022-07-27T08:59:17,738][INFO ][o.e.c.m.MetadataIndexTemplateService] [elk] adding component template [.deprecation-indexing-mappings]
[2022-07-27T08:59:17,776][INFO ][o.e.c.m.MetadataIndexTemplateService] [elk] adding component template [.deprecation-indexing-settings]
[2022-07-27T08:59:17,821][INFO ][o.e.c.m.MetadataIndexTemplateService] [elk] adding index template [logs] for index patterns [logs-*-*]
[2022-07-27T08:59:17,866][INFO ][o.e.c.m.MetadataIndexTemplateService] [elk] adding index template [synthetics] for index patterns [synthetics-*-*]

==> /var/log/logstash/logstash-plain.log <==

==> /var/log/kibana/kibana5.log <==

==> /var/log/elasticsearch/elasticsearch.log <==
[2022-07-27T08:59:17,907][INFO ][o.e.c.m.MetadataIndexTemplateService] [elk] adding index template [metrics] for index patterns [metrics-*-*]
[2022-07-27T08:59:17,956][INFO ][o.e.c.m.MetadataIndexTemplateService] [elk] adding index template [.deprecation-indexing-template] for index patterns [.logs-deprecation.*]
[2022-07-27T08:59:17,990][INFO ][o.e.x.i.a.TransportPutLifecycleAction] [elk] adding index lifecycle policy [.monitoring-8-ilm-policy]
[2022-07-27T08:59:18,064][INFO ][o.e.x.i.a.TransportPutLifecycleAction] [elk] adding index lifecycle policy [ml-size-based-ilm-policy]
[2022-07-27T08:59:18,102][INFO ][o.e.x.i.a.TransportPutLifecycleAction] [elk] adding index lifecycle policy [logs]
[2022-07-27T08:59:18,136][INFO ][o.e.x.i.a.TransportPutLifecycleAction] [elk] adding index lifecycle policy [metrics]
```

  eb container
-

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

Note: The following image link needs to be updated. Replace docker_ps_output.png with the name of your screenshot image file.

Target Machines & Beats

This ELK server is configured to monitor the following machines:

- *: List the IP addresses of the machines you are monitoring*
- *10.1.0.6, 10.0.0.7, 10.3.0.4*

We have installed the following Beats on these machines:

- *: Filebeat and Metricbeat*

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

The filebeat collects log and system log files, while the metricbeat collects metrics on your network.

- *: In 1-2 sentences, explain what kind of data each beat collects, and provide 1 example of what you expect to see. E.g., Winlogbeat collects Windows logs, which we use to track user logon events, etc. Filebeat monitors the logs files or location that you specify, collecting them and forwards them to elasticsearch or logstash, while metricbeat does the same thing for metrics*

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 install-elk.yml file to configElk.yml
- Update the hosts file to include the Elk-VM
- Run the playbook, and navigate to 20.0.161.43:5601 to check that the installation worked as expected.

Answer the following questions to fill in the blanks:

- *Which file is the playbook? Where do you copy it? Nano /etc/ansible/host*
- *Which file do you update to make Ansible run the playbook on a specific machine? How do I specify which machine to install the ELK server on versus which to install Filebeat on? Nano /etc/ansible/hosts # add elk and 10.3.04*
- _Which URL do you navigate to in order to check that the ELK server is running?
- 20.0.161.43:5601

*As a **Bonus**, provide the specific commands the user will need to run to download the playbook, update the files, e*