Name: Hermitano, Johnny C.	**Date Performed: 10/26/2022**
Course/Section: CPE31S23	**Date Submitted: 10/26/2022**
Instructor: Engr. Jonathan Taylar	**Semester and SY: 1st sem sy 2022-2023**

Activity 10: Install, Configure, and Manage Log Monitoring tools

1. Objectives

Create and design a workflow that installs, configure and manage enterprise log monitoring tools using Ansible as an Infrastructure as Code (IaC) tool.

2. Discussion

Log monitoring software scans and monitors log files generated by servers, applications, and networks. By detecting and alerting users to patterns in these log files, log monitoring software helps solve performance and security issues. System administrators use log monitoring software to detect common important events indicated by log files.

Log monitoring software helps maintain IT infrastructure performance and pinpoints issues to prevent downtime and mitigate risks. These tools will often integrate with IT alerting software, log analysis software, and other IT issue resolution products to more aptly flesh out the IT infrastructure maintenance ecosystem.

To qualify for inclusion in the Log Monitoring category, a product must:

- Monitor the log files generated by servers, applications, or networks
- Alert users when important events are detected
- Provide reporting capabilities for log files

Elastic Stack

ELK suite stands for Elasticsearch, Kibana, Beats, and Logstash (also known as the ELK Stack). Source: https://www.elastic.co/elastic-stack

The Elastic Stack is a group of open source products from Elastic designed to help users take data from any type of source and in any format, and search, analyze and visualize that data in real time. The product group was formerly known as the ELK Stack for the core products in the group -- Elasticsearch, Logstash and Kibana -- but has been rebranded as the Elastic Stack. A fourth product, Beats, was subsequently added to the stack. The Elastic Stack can be deployed on premises or made available as software as a service (SaaS). Elasticsearch supports Amazon Web Services (AWS), Google Cloud Platform and Microsoft Azure.

GrayLog

Graylog is a powerful platform that allows for easy log management of both structured and unstructured data along with debugging applications.

It is based on Elasticsearch, MongoDB, and Scala. Graylog has a main server, which receives data from its clients installed on different servers, and a web interface, which visualizes the data and allows to work with logs aggregated by the main server.

We use Graylog primarily as the stash for the logs of the web applications we build. However, it is also effective when working with raw strings (i.e. syslog): the tool parses it into the structured data we need. It also allows advanced custom search in the logs using structured queries. In other words, when integrated properly with a web app, Graylog helps engineers to analyze the system behavior on almost per code line basis.

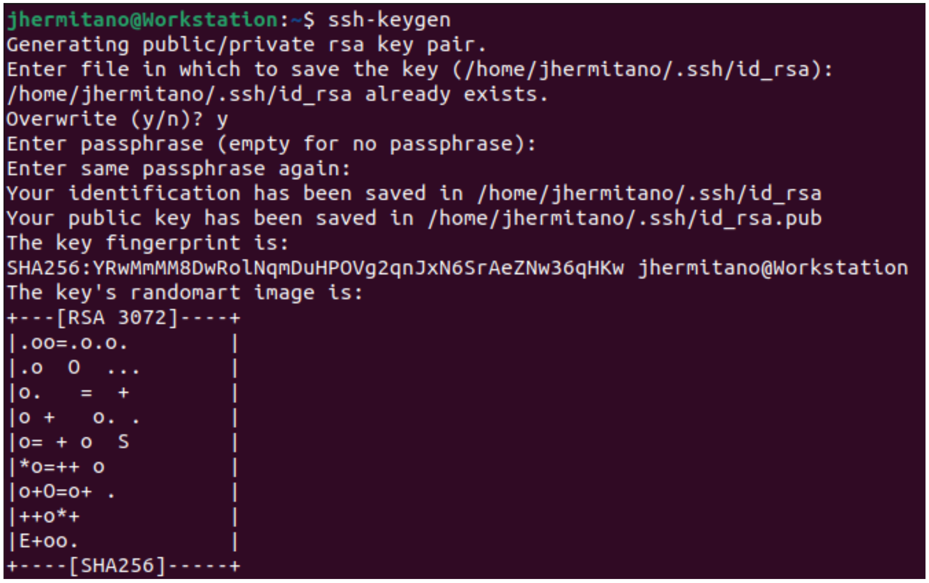
Source: https://www.graylog.org/products/open-source

3. Tasks

1. Create a playbook that:
    a. Install and configure Elastic Stack in separate hosts (Elastic Search, Kibana, Logstash)
2. Apply the concept of creating roles.
3. Describe how you did step 1. (Provide screenshots and explanations in your report. Make your report detailed such that it will look like a manual.)
4. Show an output of the installed Elastic Stack for both Ubuntu and CentOS.
5. Make sure to create a new repository in GitHub for this activity.

4. **Output** (screenshots and explanations)

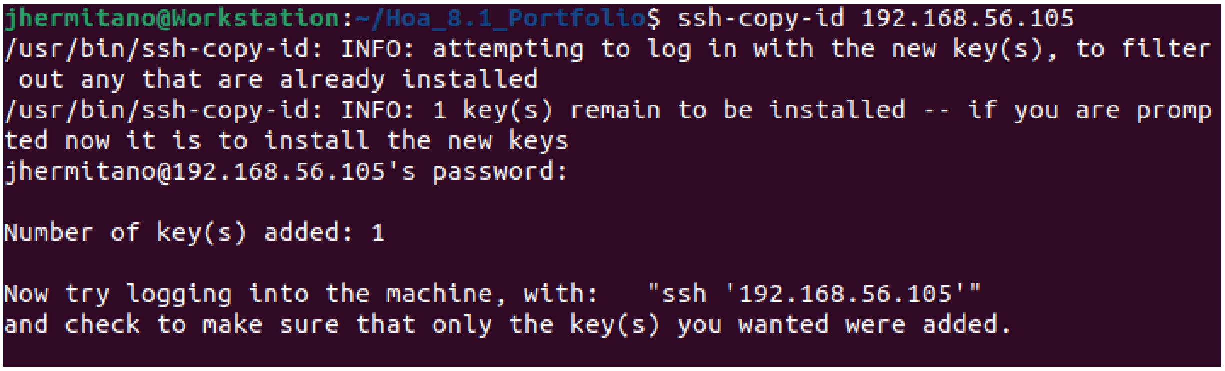
Step 1. Enter the command **ssh-keygen** to generate an rsa key.



```
jhermitano@Workstation:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/jhermitano/.ssh/id_rsa):
/home/jhermitano/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/jhermitano/.ssh/id_rsa
Your public key has been saved in /home/jhermitano/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:YRwMmMM8DwRolNqmDuHPOVg2qnJxN6SrAeZNw36qHKw jhermitano@Workstation
The key's randomart image is:
+---[RSA 3072]----+
|.oo=.o.o.        |
|.o  O  ...       |
|o.   =  +        |
|o +   o. .       |
|o= + o  S        |
|*o=++ o          |
|o+O=o+ .         |
|++o*+            |
|E+oo.            |
+----[SHA256]-----+
```

Step 2. Connect your control node to your manage node through ssh by entering the code **ssh-copy-id server@ip address** for Ubuntu and **ssh-copy-id -i ~/.ssh/id_rsa server@ip address** for CentOS.

Ubuntu

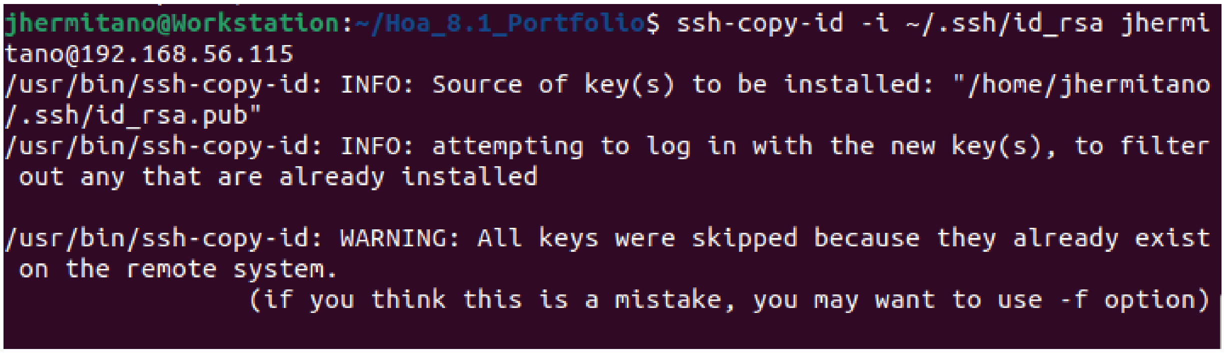


```
jhermitano@Workstation:~/Hoa_8.1_Portfolio$ ssh-copy-id 192.168.56.105
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
 out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are promp
ted now it is to install the new keys
jhermitano@192.168.56.105's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh '192.168.56.105'"
and check to make sure that only the key(s) you wanted were added.
```

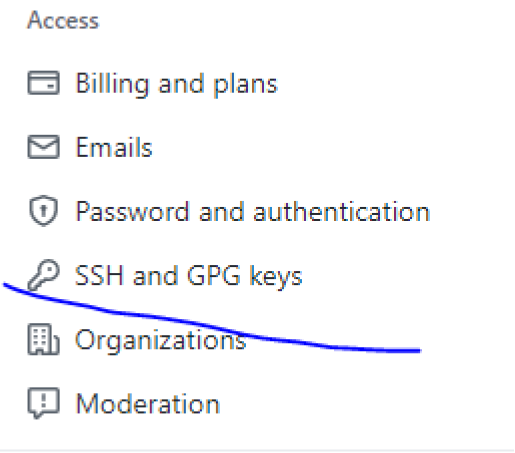
CentOS



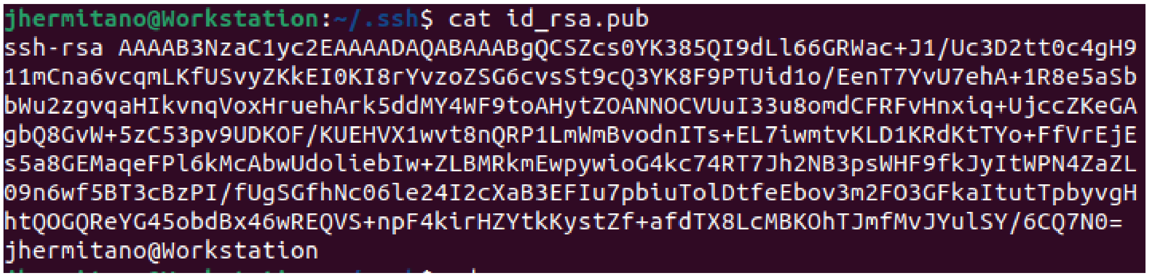
```
jhermitano@Workstation:~/Hoa_8.1_Portfolio$ ssh-copy-id -i ~/.ssh/id_rsa jhermi
tano@192.168.56.115
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/jhermitano
/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
 out any that are already installed

/usr/bin/ssh-copy-id: WARNING: All keys were skipped because they already exist
 on the remote system.
                (if you think this is a mistake, you may want to use -f option)
```

Step 3. Connect your control node to your github account by adding your ssh key onto your github. To connect go to settings and click the **SSH and GPG keys.**

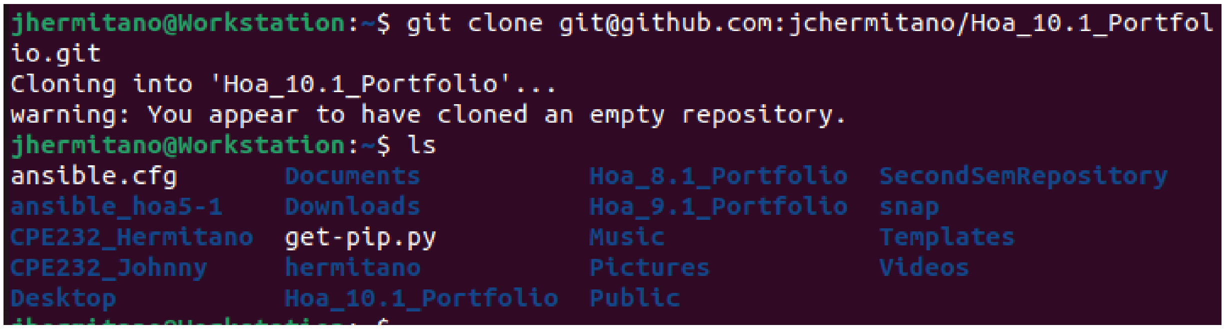


Just click the **New SSH keys** and then you may copy paste your rsa key from your control node. To see the rsa keys, enter **cat id_rsa.pub** and you will see your rsa keys.



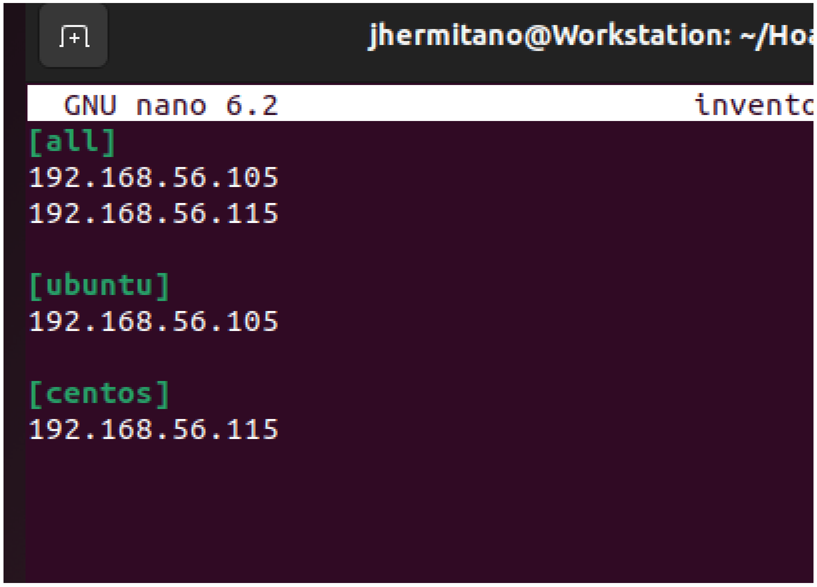
```
jhermitano@Workstation:~/.ssh$ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQCSZcs0YK385QI9dLl66GRWac+J1/Uc3D2tt0c4gH9
11mCna6vcqmLKfUSvyZKkEI0KI8rYvzoZSG6cvsSt9cQ3YK8F9PTUid1o/EenT7YvU7ehA+1R8e5aSb
bWu2zgvqaHIkvnqVoxHruehArk5ddMY4WF9toAHytZOANNOCVUuI33u8omdCFRFvHnxiq+UjccZKeGA
gbQ8GvW+5zC53pv9UDKOF/KUEHVX1wvt8nQRP1LmWmBvodnITs+EL7iwmtvKLD1KRdKtTYo+FfVrEjE
s5a8GEMaqeFPl6kMcAbwUdoliebIw+ZLBMRkmEwpywioG4kc74RT7Jh2NB3psWHF9fkJyItWPN4ZaZL
09n6wf5BT3cBzPI/fUgSGfhNc06le24I2cXaB3EFIu7pbiuTolDtfeEbov3m2FO3GFkaItutTpbyvgH
htQOGQReYG45obdBx46wREQVS+npF4kirHZYtkKystZf+afdTX8LcMBKOhTJmfMvJYulSY/6CQ7N0=
jhermitano@Workstation
```

Step 4. Clone your github repository by entering **git clone "your github repository link"**.



```
jhermitano@Workstation:~$ git clone git@github.com:jchermitano/Hoa_10.1_Portfol
io.git
Cloning into 'Hoa_10.1_Portfolio'...
warning: You appear to have cloned an empty repository.
jhermitano@Workstation:~$ ls
ansible.cfg       Documents           Hoa_8.1_Portfolio  SecondSemRepository
ansible_hoa5-1    Downloads           Hoa_9.1_Portfolio  snap
CPE232_Hermitano  get-pip.py          Music              Templates
CPE232_Johnny     hermitano           Pictures           Videos
Desktop           Hoa_10.1_Portfolio  Public
```

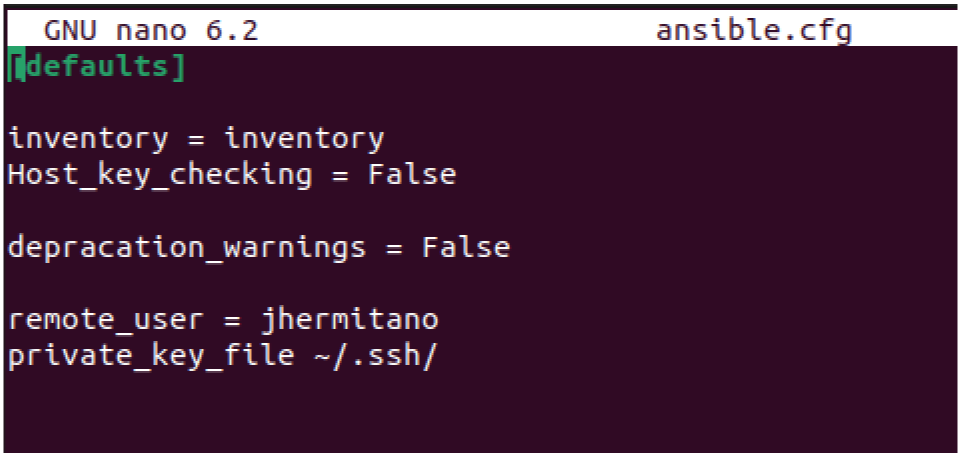
Step 5. Inside your repository, create your nano inventory and ansible.cfg for your playbook.



```
  GNU nano 6.2                    invento
[all]
192.168.56.105
192.168.56.115

[ubuntu]
192.168.56.105

[centos]
192.168.56.115
```



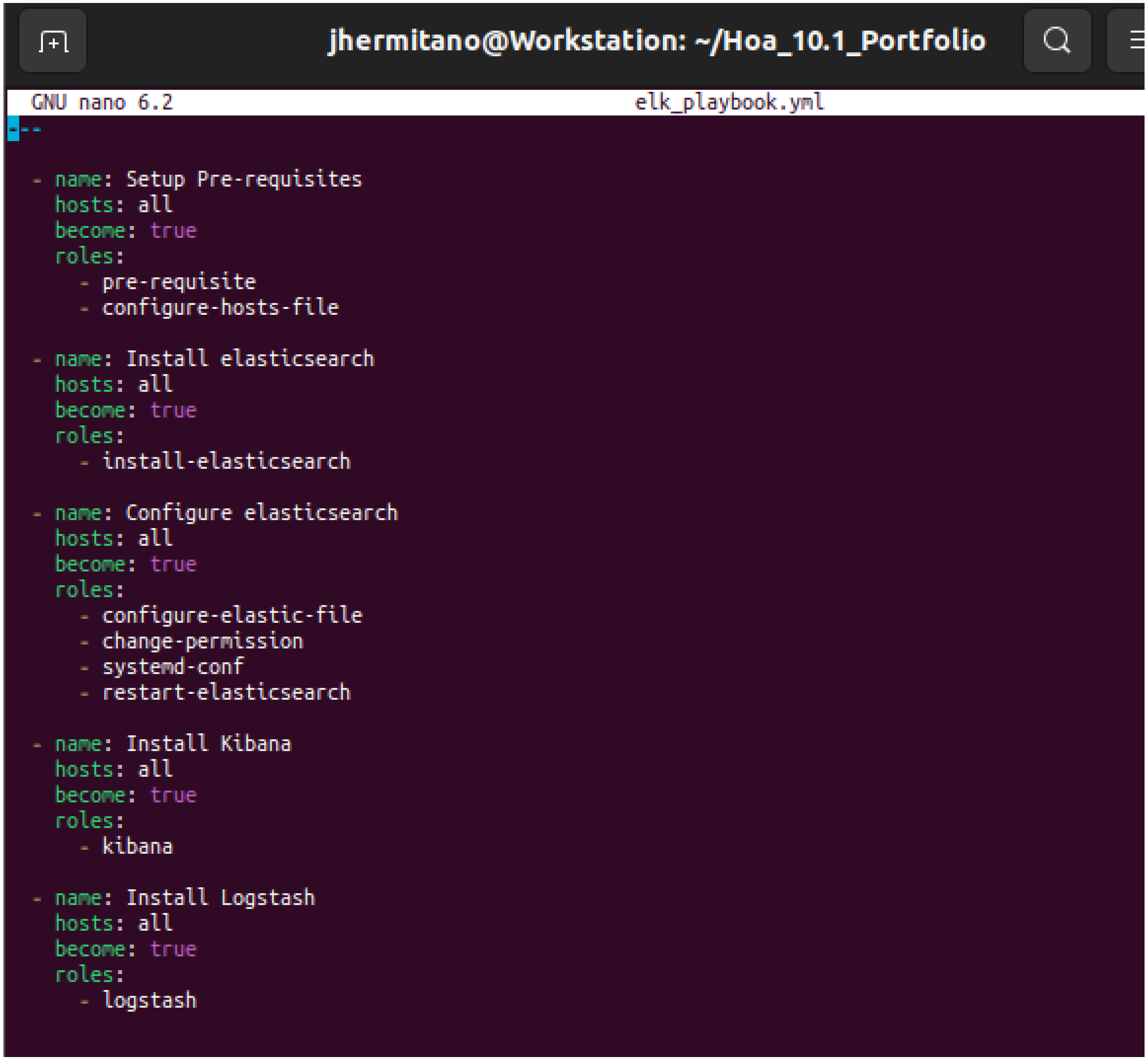
```
  GNU nano 6.2                    ansible.cfg
[defaults]

inventory = inventory
Host_key_checking = False

depracation_warnings = False

remote_user = jhermitano
private_key_file ~/.ssh/
```

Step 6. Create your playbook.



```
GNU nano 6.2                                              elk_playbook.yml
---

 - name: Setup Pre-requisites
   hosts: all
   become: true
   roles:
     - pre-requisite
     - configure-hosts-file

 - name: Install elasticsearch
   hosts: all
   become: true
   roles:
     - install-elasticsearch

 - name: Configure elasticsearch
   hosts: all
   become: true
   roles:
     - configure-elastic-file
     - change-permission
     - systemd-conf
     - restart-elasticsearch

 - name: Install Kibana
   hosts: all
   become: true
   roles:
     - kibana

 - name: Install Logstash
   hosts: all
   become: true
   roles:
     - logstash
```

Reflections:

Answer the following:

1. What are the benefits of having a log monitoring tool?

   Tech experts can more easily identify problem areas, evaluate the health of the application, enhance troubleshooting, and optimize root cause analysis of application performance issues by collecting, analyzing, and monitoring these logs.

Conclusions:

I was able to develop and improve a playbook in this project that uses ansible to install log monitoring tools in CentOS and Ubuntu. In addition, I now have a better understanding of the Ansible playbook and role definitions.

Github link: https://github.com/jchermitano/Hoa_10.1_Portfolio.git