Usage Instructions for Automated ELK Stack Deployment

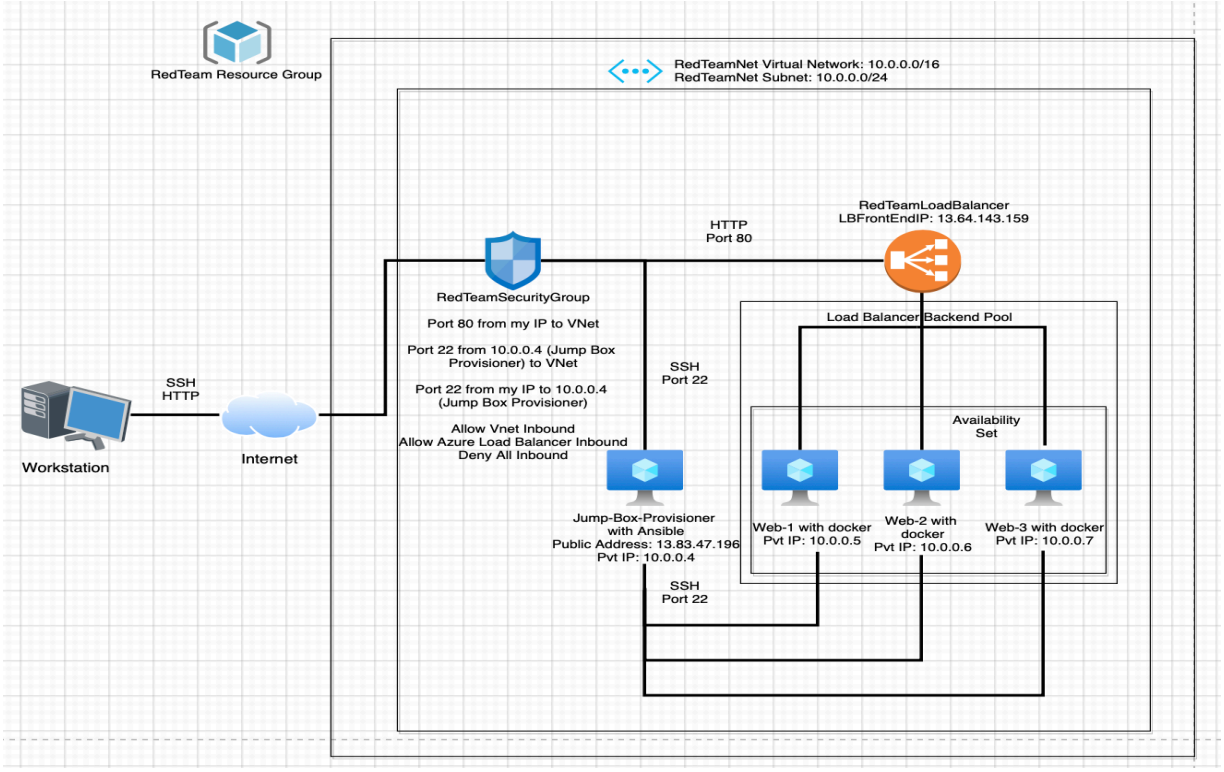
Pre-requisites

- Azure Portal account
- Resource Group: RedTeam
- Virtual Network: RedTeamNet (10.0.0.0/16; Subnet – 10.0.0.0/24)
- Network Security Group: RedTeamSecurityGroup

  Rules:
  - Port 80 from workstation IP to VNet
  - Port 22 from 10.0.0.4 (Jump Box Provisioner) to VNet
  - Port 22 from workstation IP to 10.0.0.4 (Jump Box Provisioner)
  - Allow Vnet Inbound
  - Allow Azure Load Balancer Inbound
  - Deny All Inbound

- Jumpbox: Jump-Box-Provisioner with Ansible (Public IP: 13.83.47.196 Private IP: 10.0.0.4)
- Web Servers: Web-1 (Private IP: 10.0.0.5), Web-2 (Private IP: 10.0.0.6) and Web-3 (Private IP: 10.0.0.7)
- Load Balancer: RedTeamLoadBalancer (Front End IP: 13.64.143.159)

The diagram below shows the network diagram representing the pre-requisites.

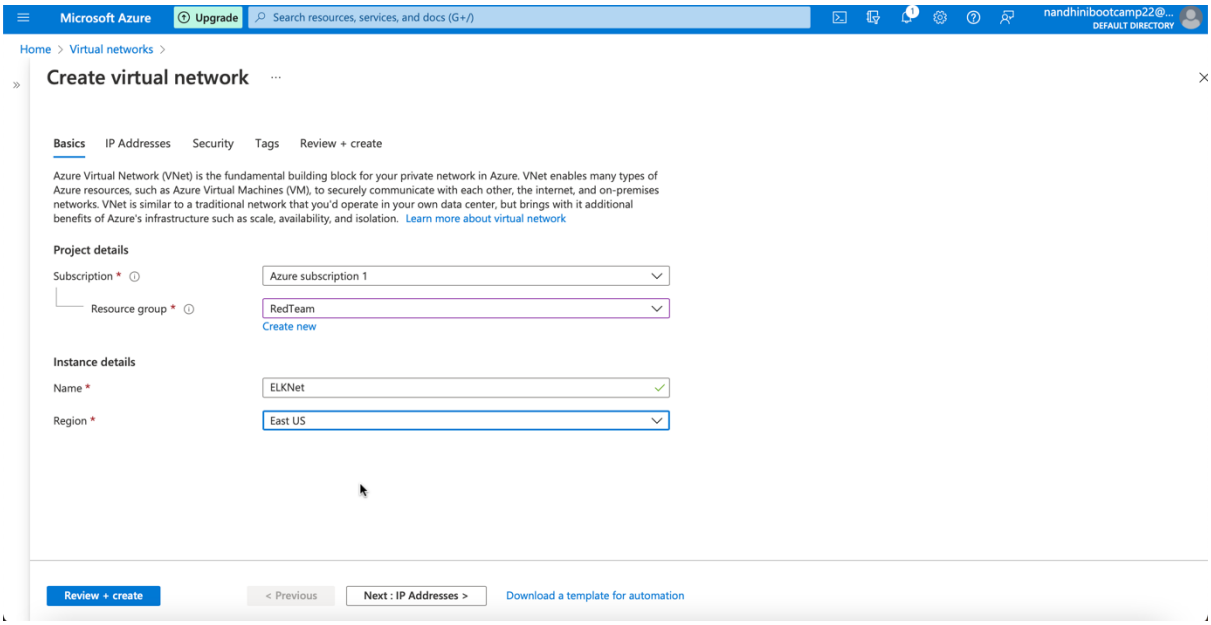


Instructions:

Chapter 1: Create a new Virtual Network

1. On the Azure portal, create a new virtual network located in the same resource group where the pre-requisite VMs (web servers) are located and in a new region that is different from that of the other VMs.

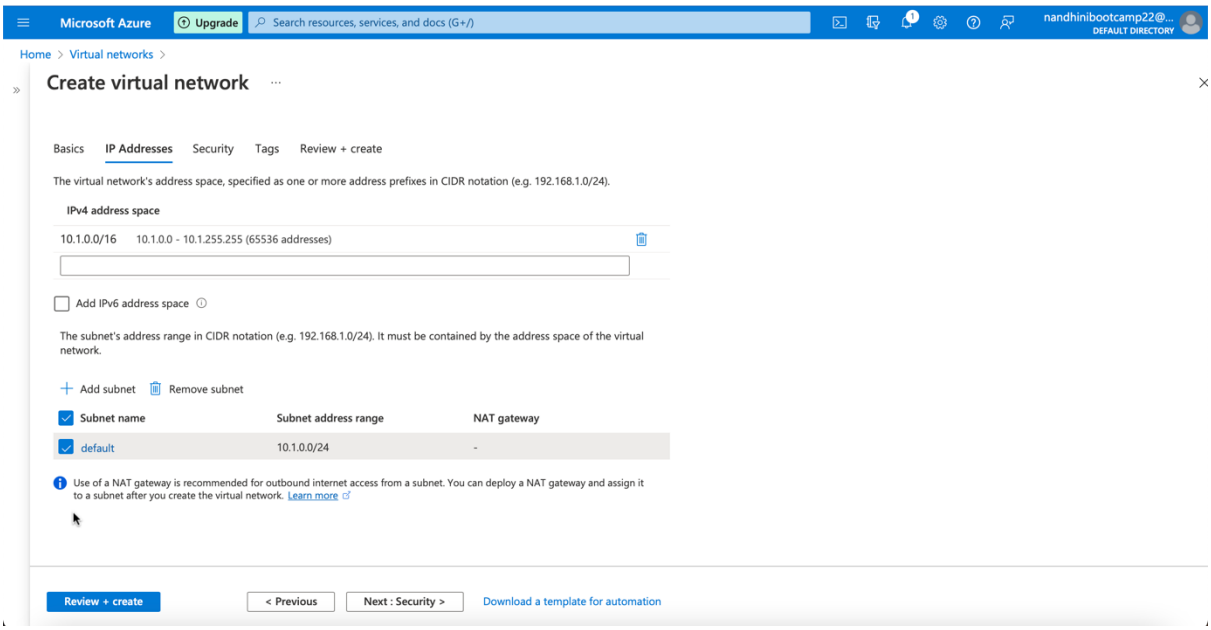
   1.1 In my example, I had created **RedTeamNet** in US West zone and created **ELKNet** in US East zone.



   1.2 Click Next.

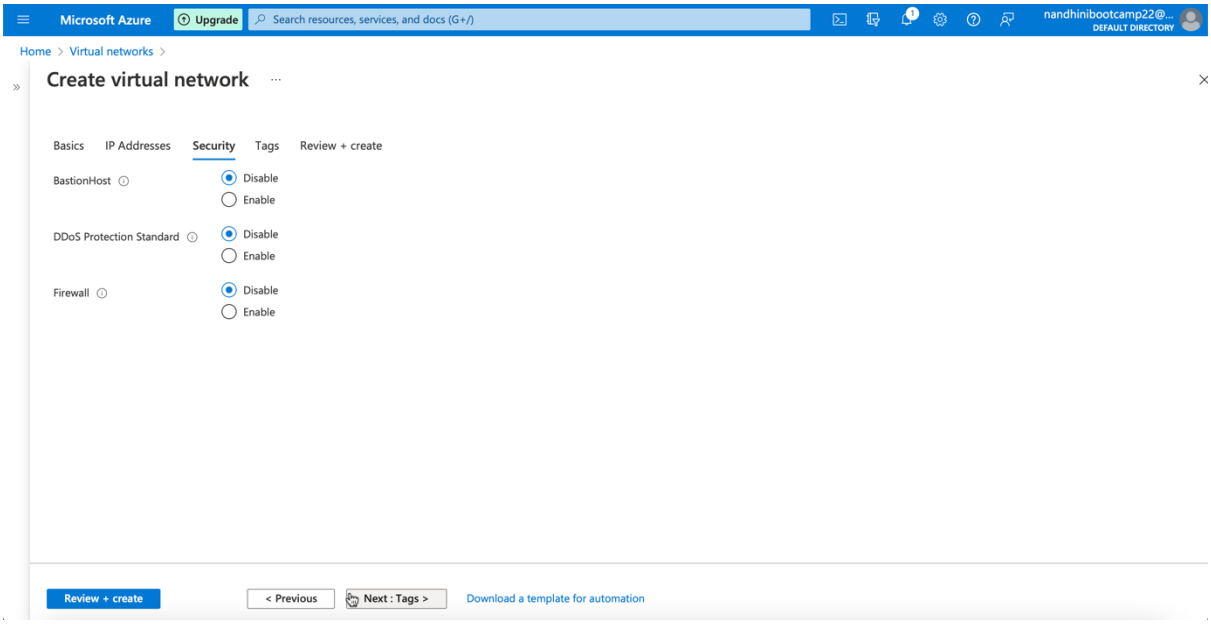
   The IP addresses space will be automatically created.

   Select the default subnet.



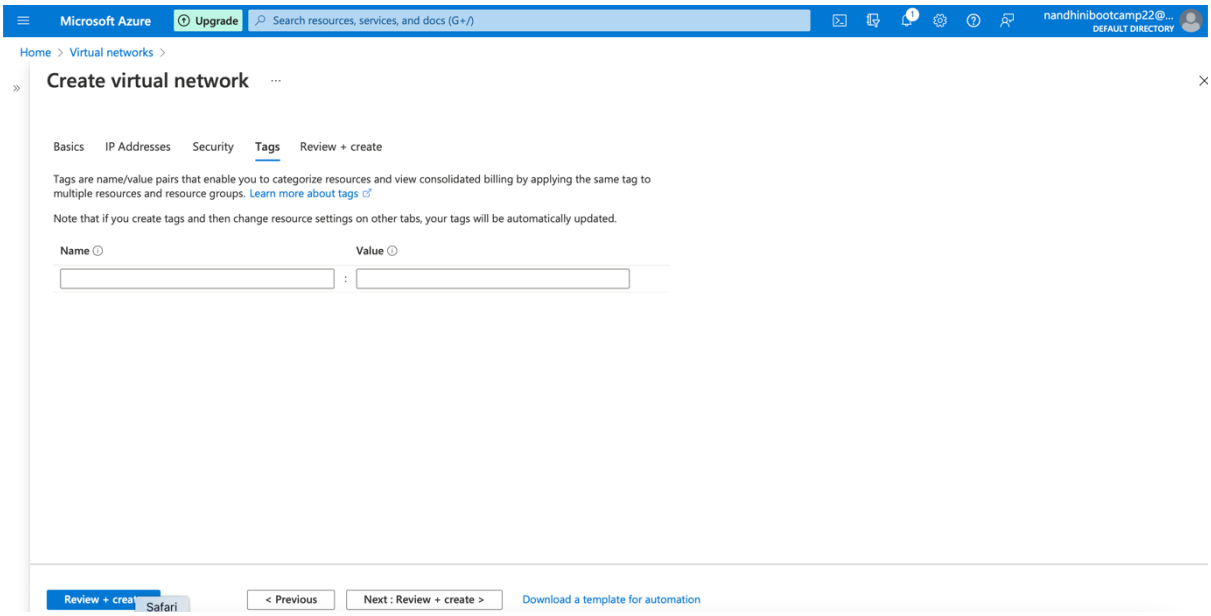
1.3 Click Next.

Leave the settings at default.



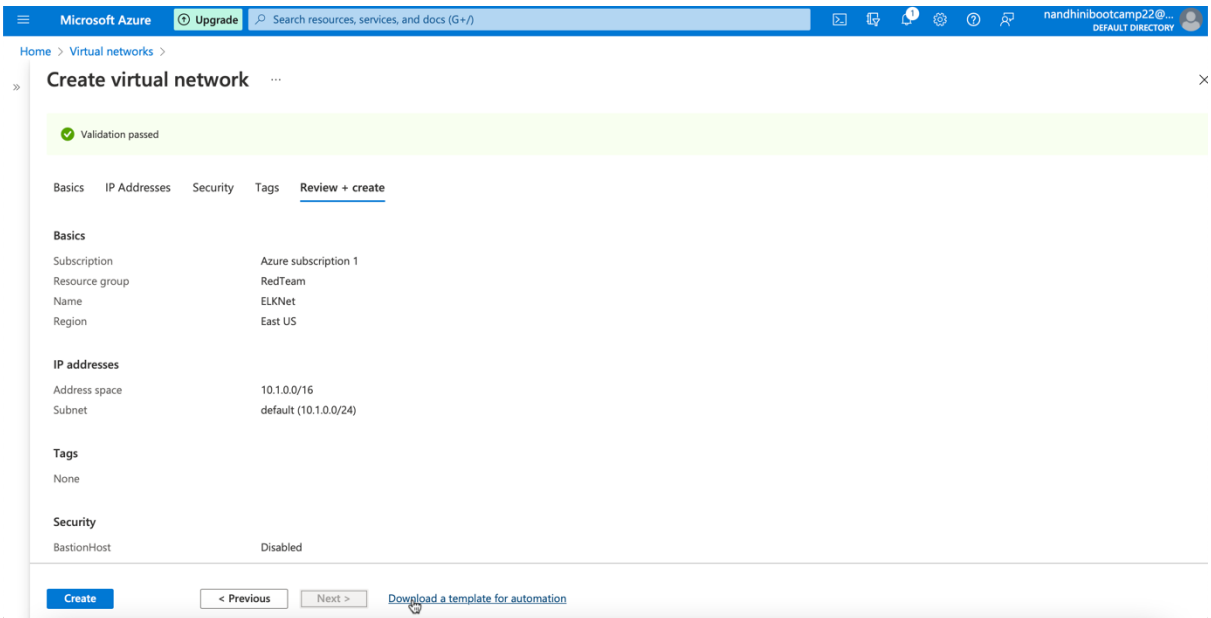
1.4 Click Next.

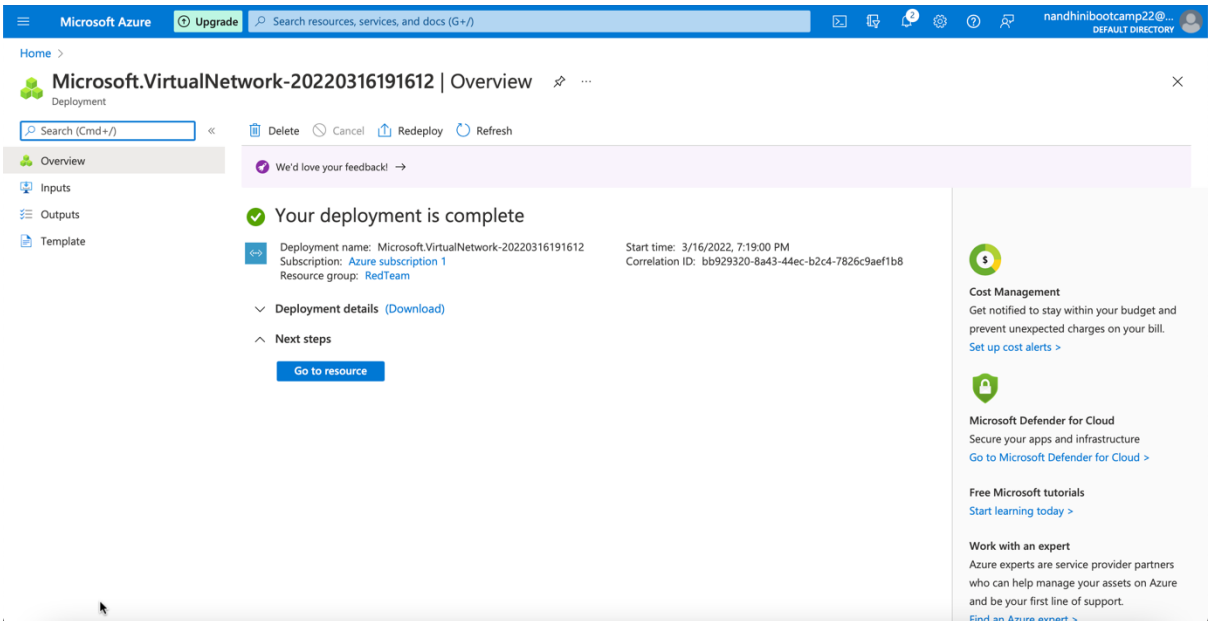
Leave the settings at default.



1.5 Click Next.

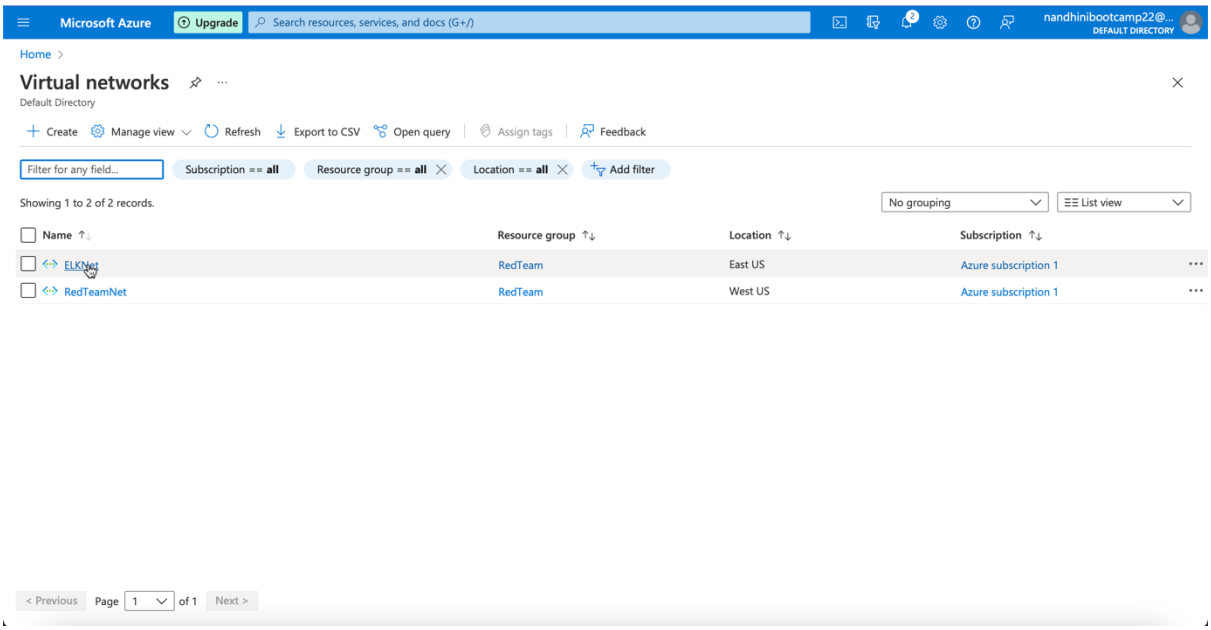
Review the settings and click on Create.

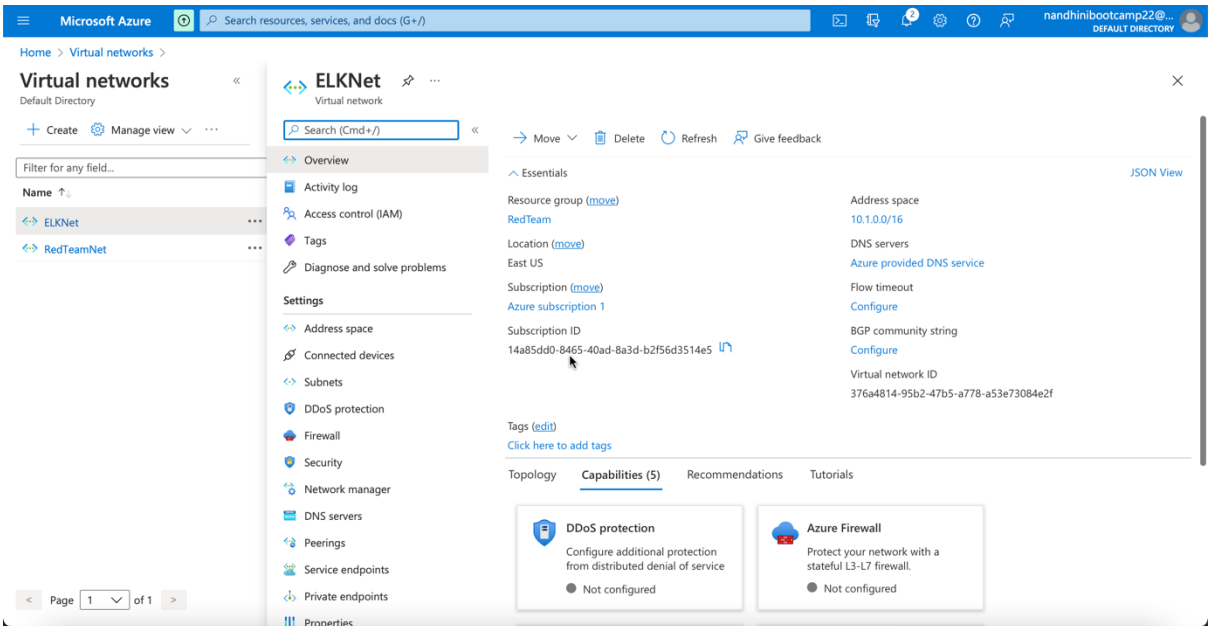




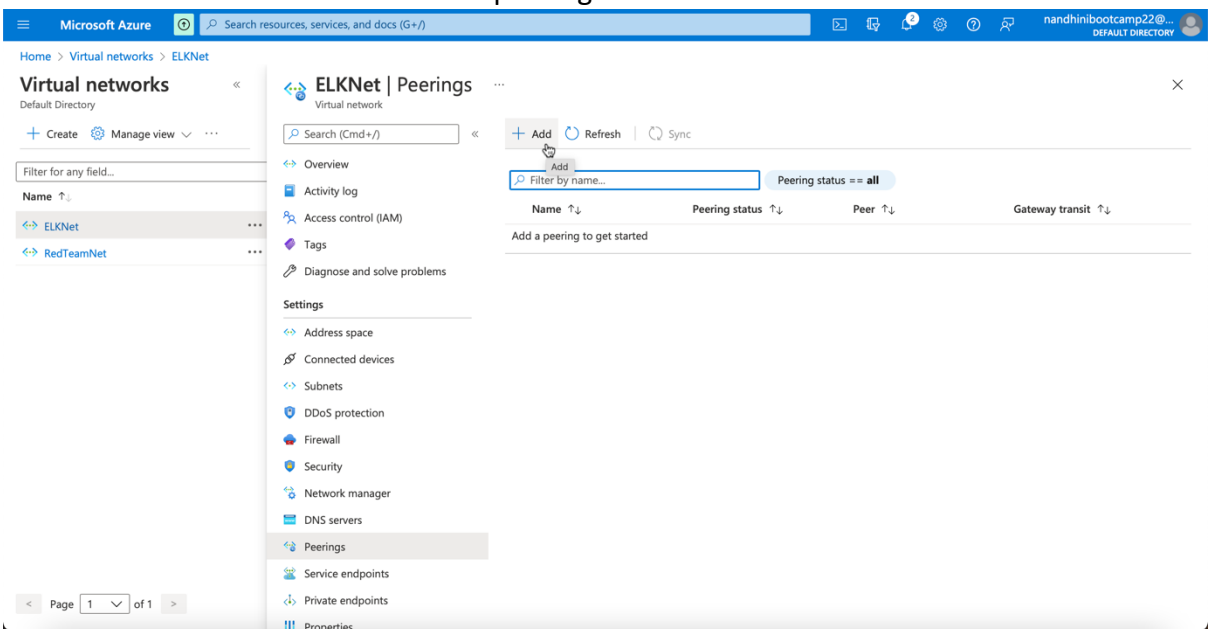
2. Create a peer connection between your virtual networks.

   2.1 Navigate to 'Virtual Network' in the Azure Portal.
   Select your new virtual network.

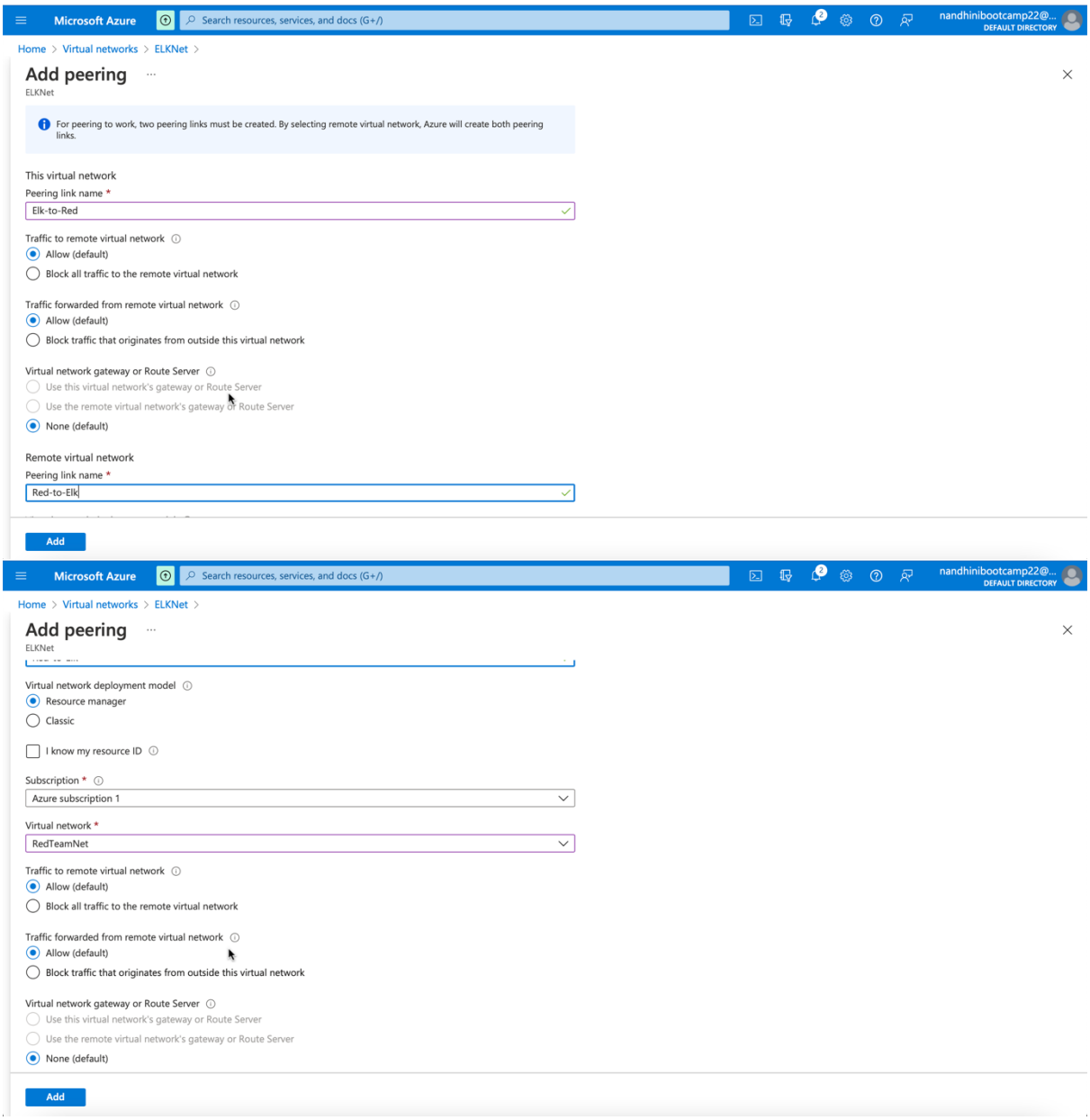




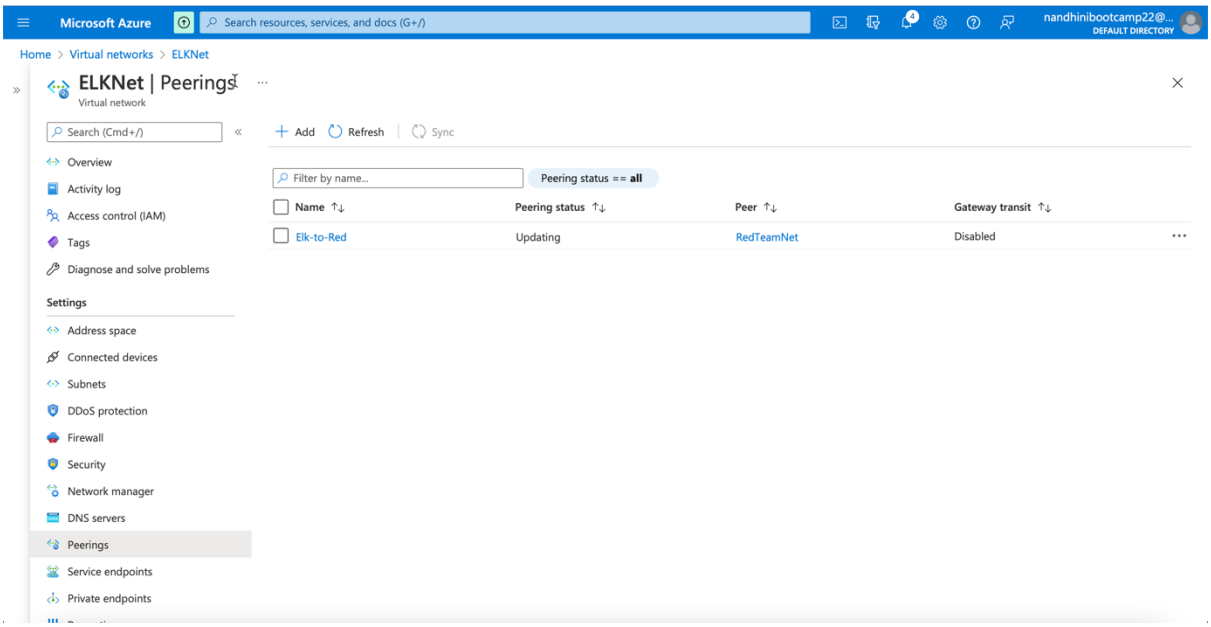
2.2 Under ‘Settings’ on the left side, select ‘Peerings’.
Click on ‘Add’ to create a new peering.



2.3 Add peering link names for the ELK virtual network and remote virtual network (RedTeamNet).
Select remote virtual network (RedTeamNet) from the ‘Virtual Network’ dropdown.
Leave the remaining settings as default.



2.4 Click Next


Microsoft Azure
Search resources, services, and docs (G+/)
nandhinibootcamp22@...
DEFAULT DIRECTORY
Home > Virtual networks > ELKNet
ELKNet | Peerings
Virtual network
Search (Cmd+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Settings
Address space
Connected devices
Subnets
DDoS protection
Firewall
Security
Network manager
DNS servers
Peerings
Service endpoints
Private endpoints
Add
Refresh
Sync
Filter by name...
Peering status == all
Name
Peering status
Peer
Gateway transit
Elk-to-Red
Updating
RedTeamNet
Disabled

Chapter 2: Create a new Virtual Machine

1. Create a new Ubuntu VM (ELKServer)in the virtual network with the following configurations:

   a. **RAM:** 4 GB+
   You can use either of these machines:
   - Standard D2s v3 (2 vcpus, 8GiB memory)
   - Standard B2s (2vcpus, 4GiB memory

   In case you are unable to get the required VM, try deploying the VM in a different region.

   b. **Operating System:** Ubuntu Server 18.04 LTS – Gen2 (free services eligible)

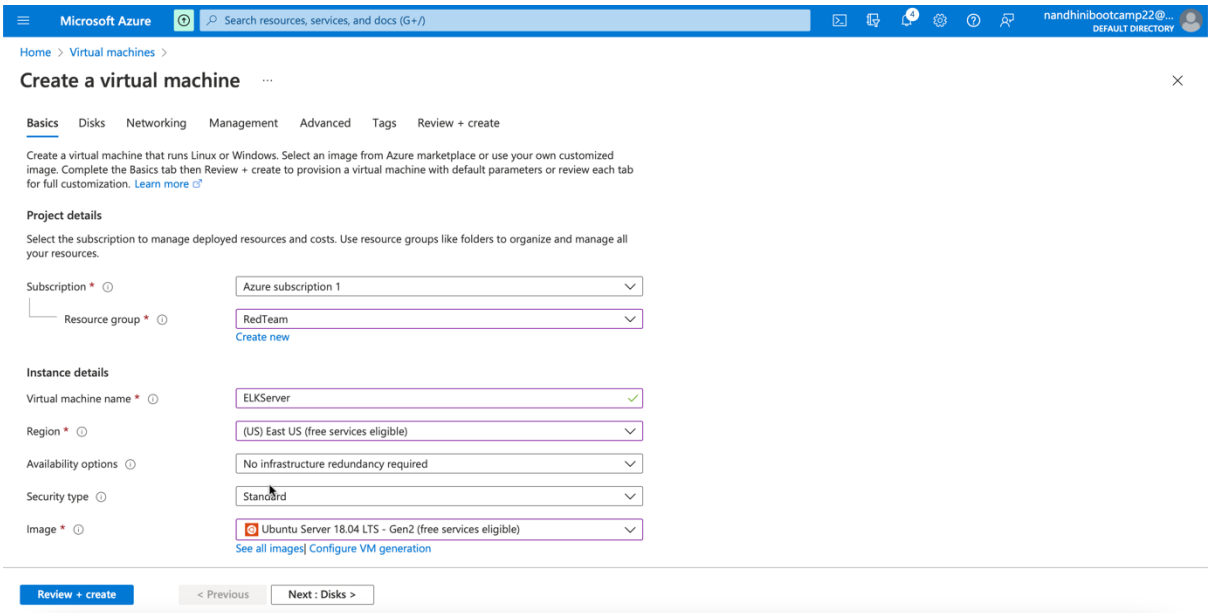
   To get the SSH key created on the Ansible container running on your jump box, follow the steps below:

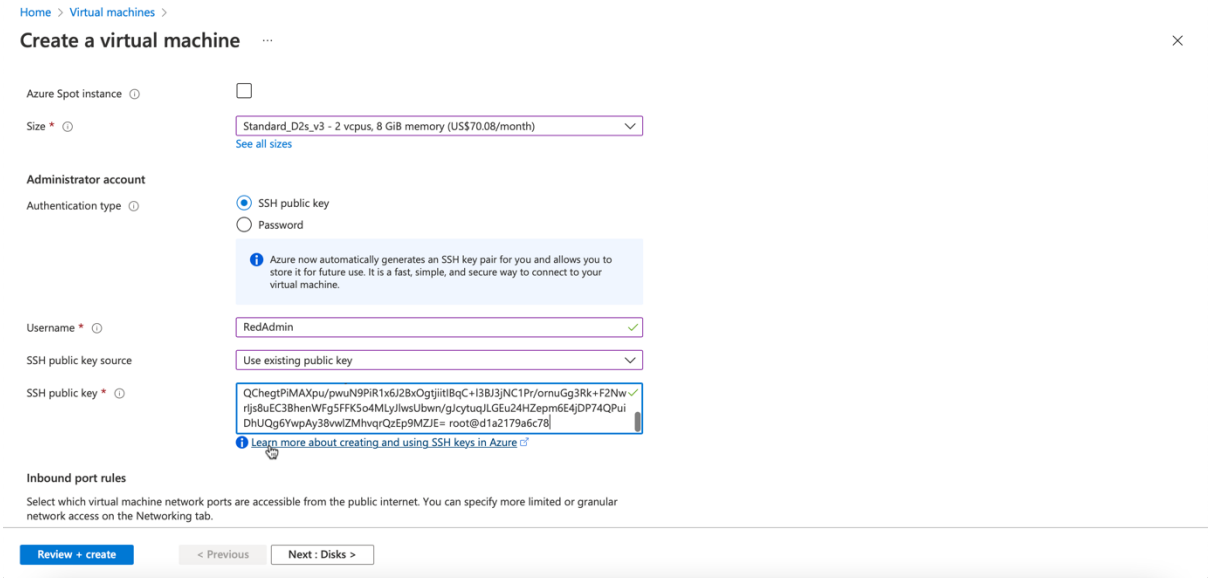
   1. SSH into the jump box through the terminal on your workstation.
   2. Run the docker commands to start and attach to your Ansible container.

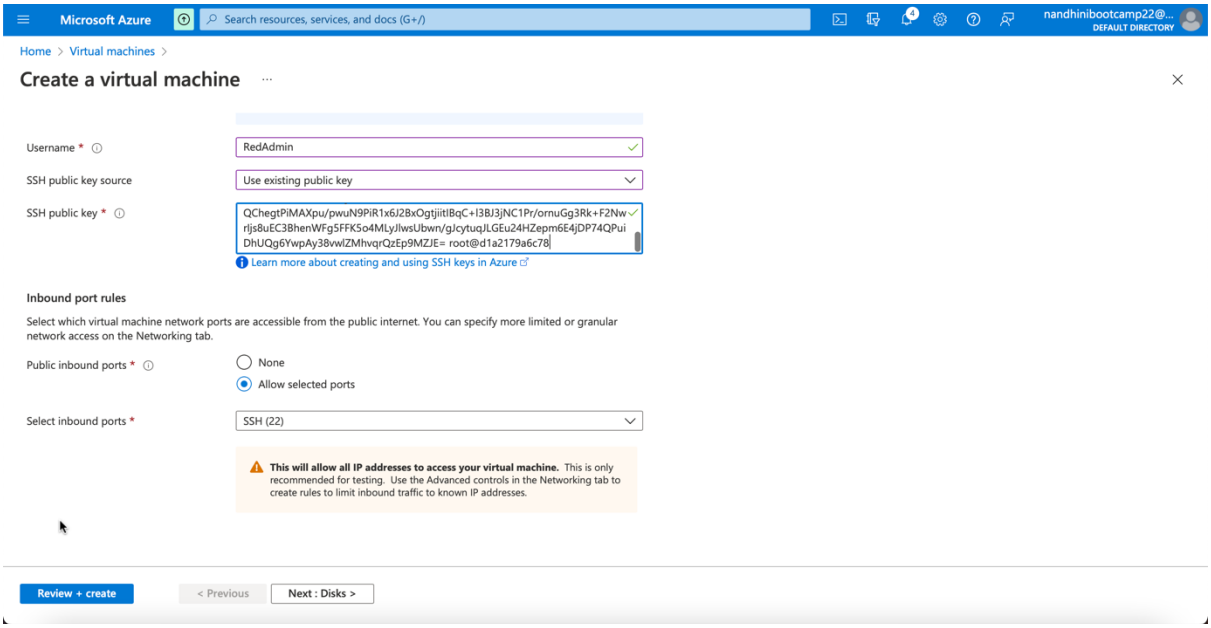
      sudo docker start (container_name)
      sudo docker attach (container_name)

   3. Retrieve your public ssh key.

      cat ~/.ssh/id_rsa.pub

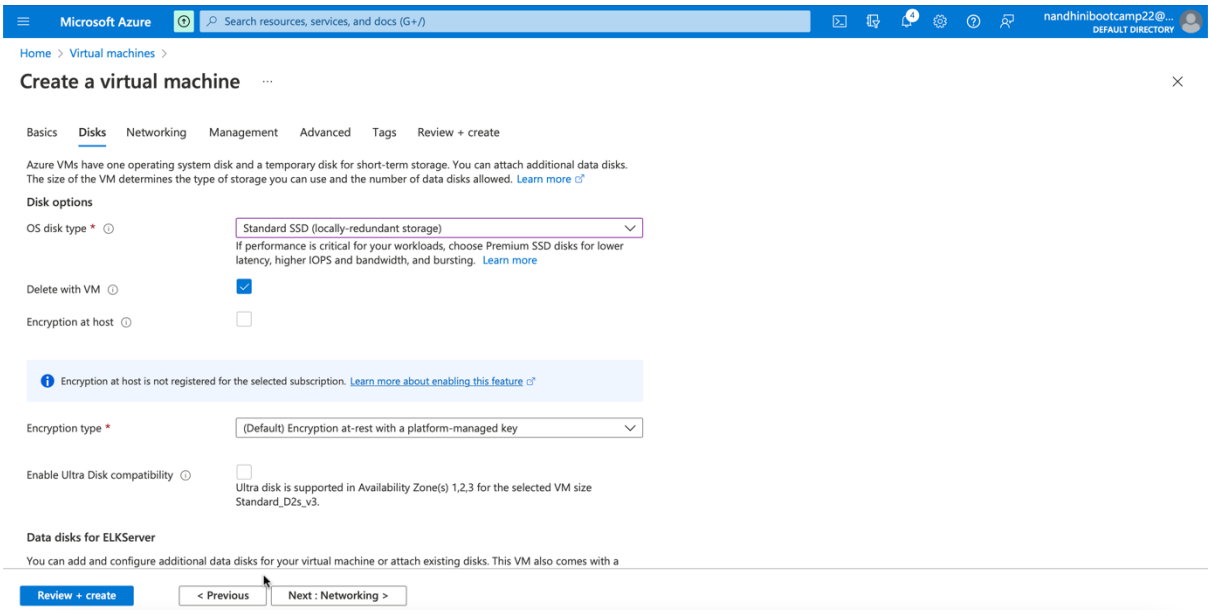


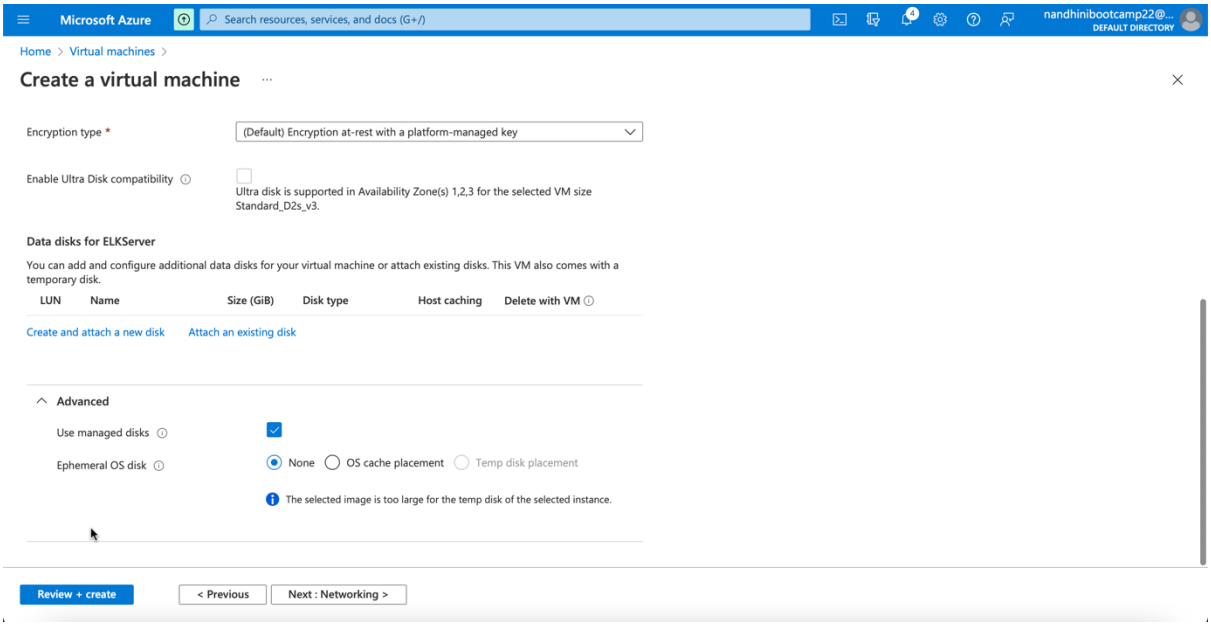




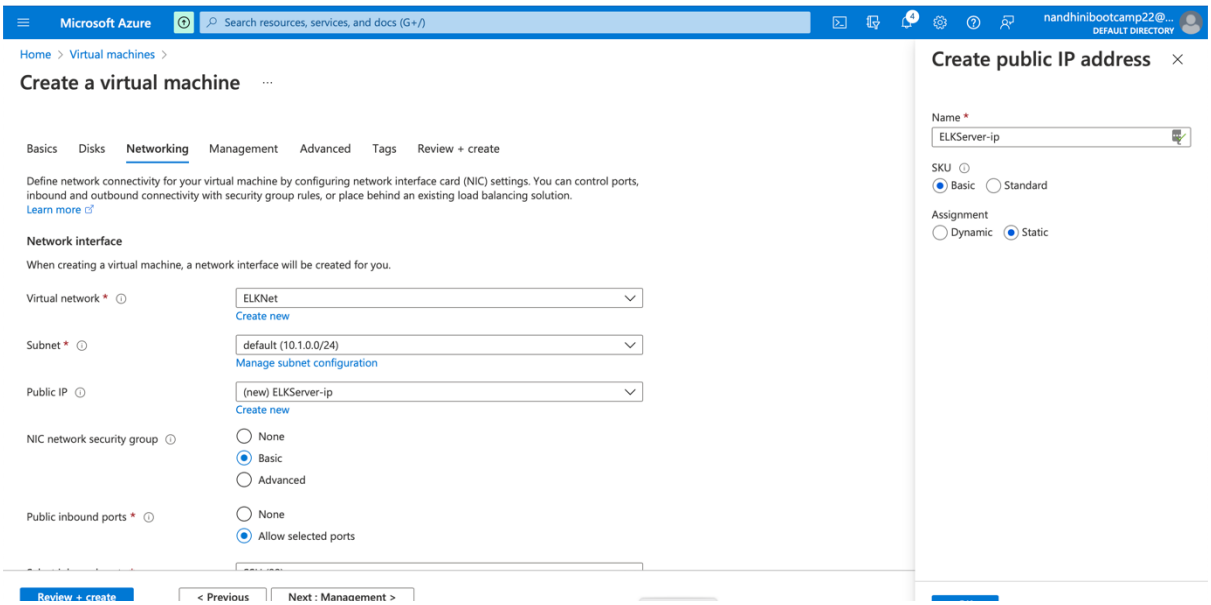
Click Next.

c. **Disks:** Standard SSD for OS Disk Type. Leave the remaining settings as default.

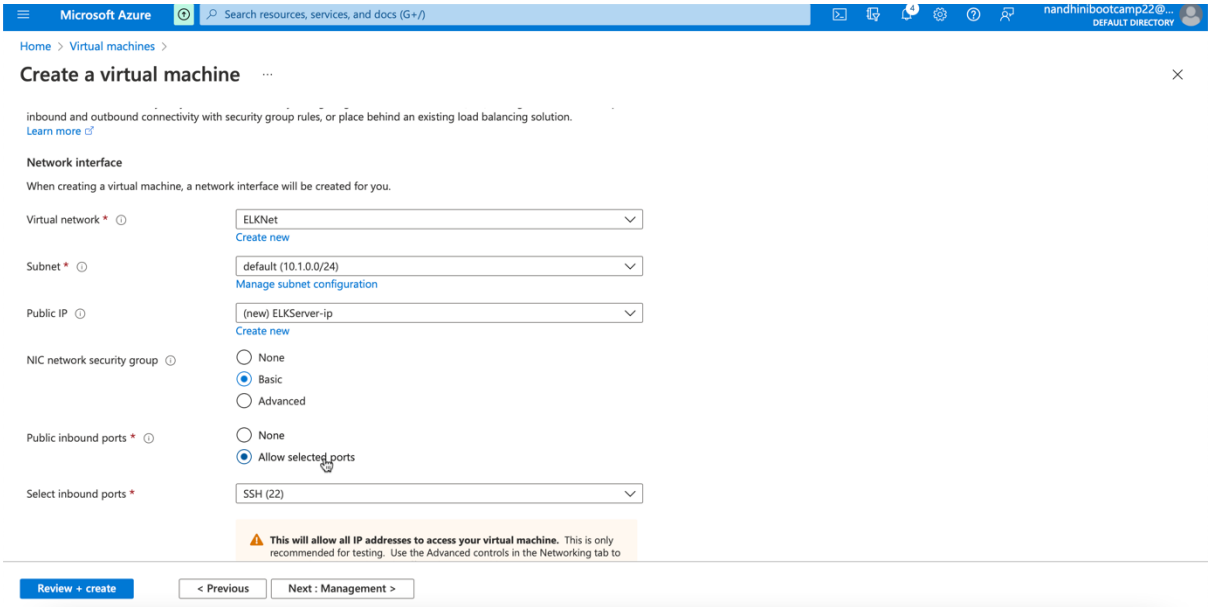


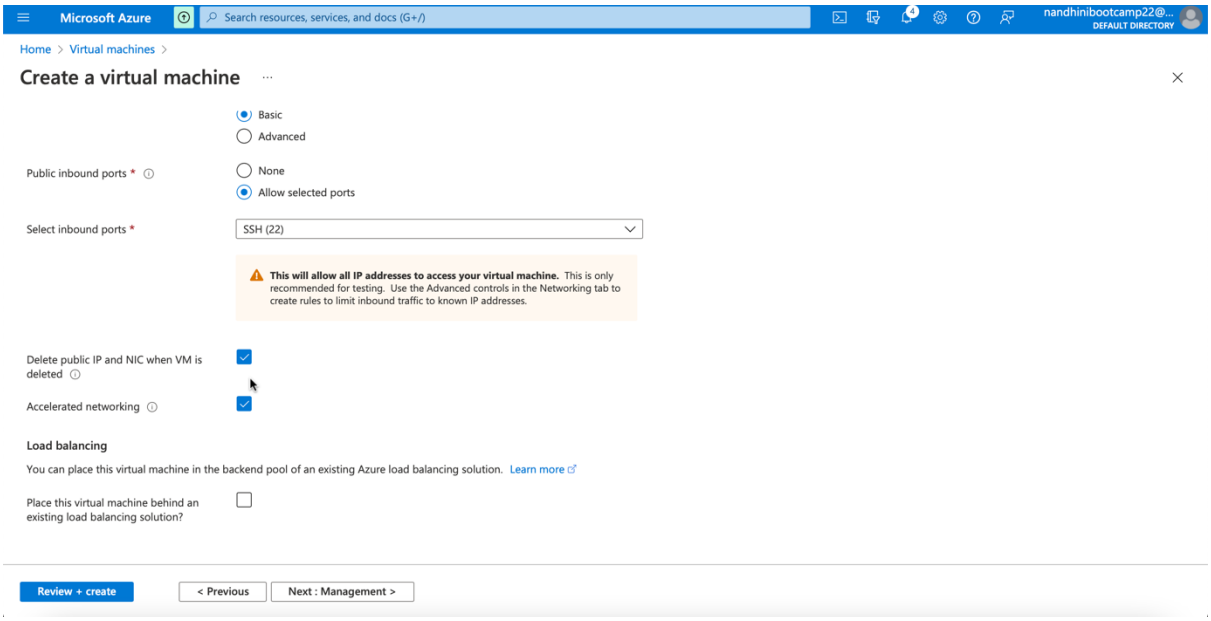


d. **IP Address:** Create a new static public IP Address.



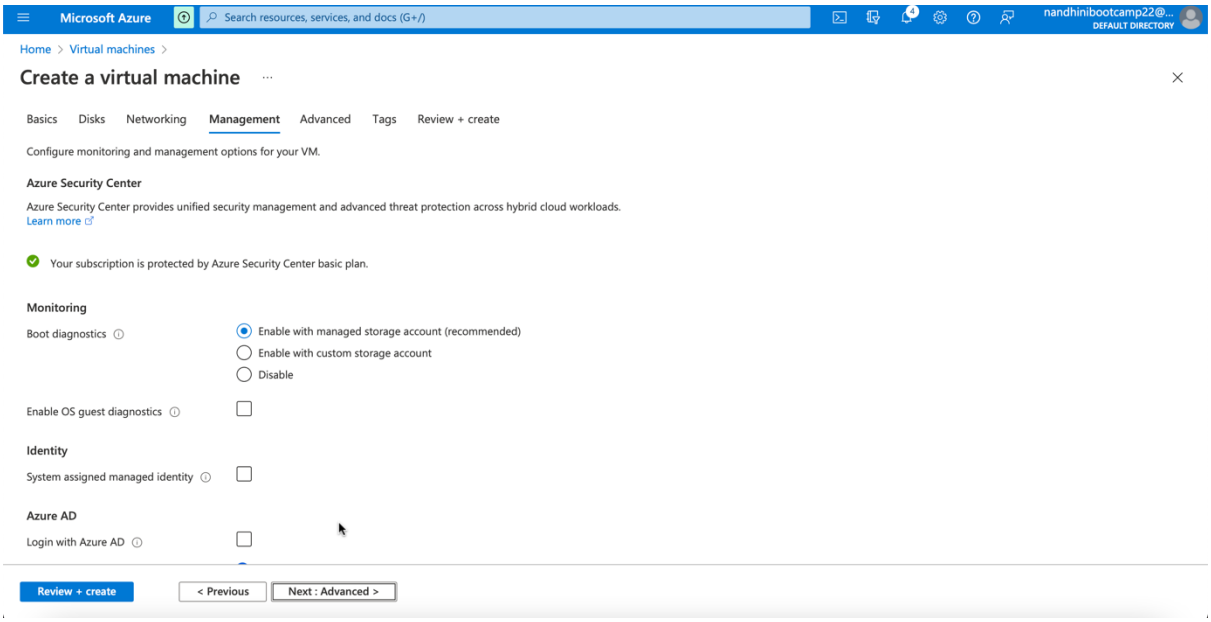
e. **Networking:** The VM should be added to the new region where you have created the virtual network. A new basic Security Group to be created for this VM.

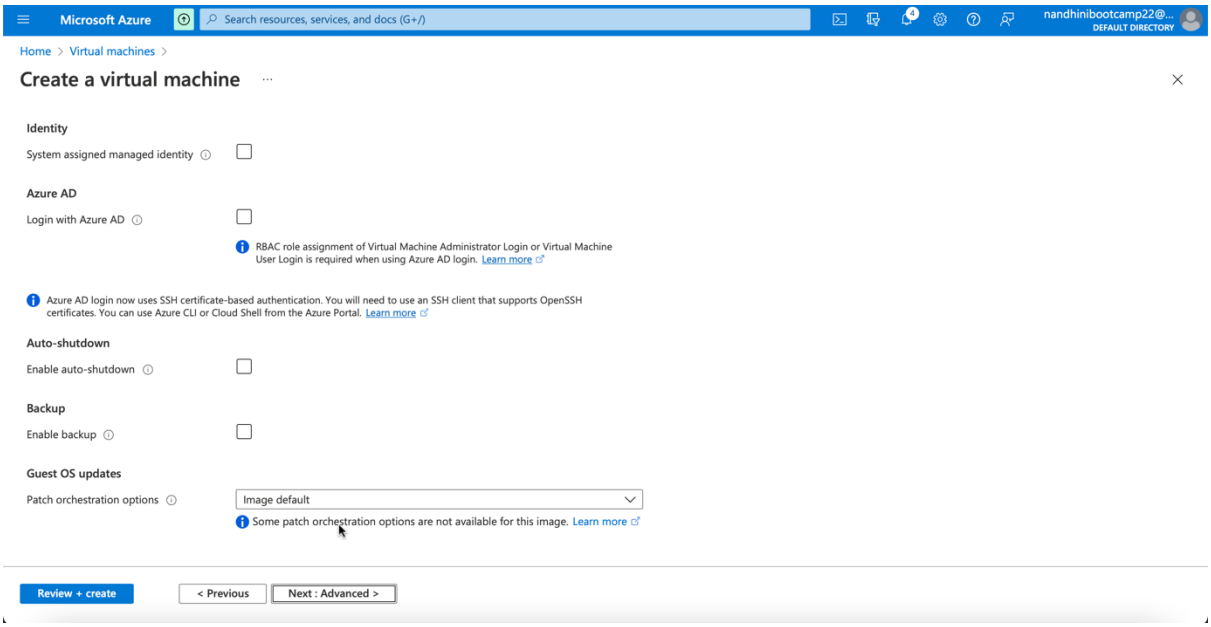




Click Next.

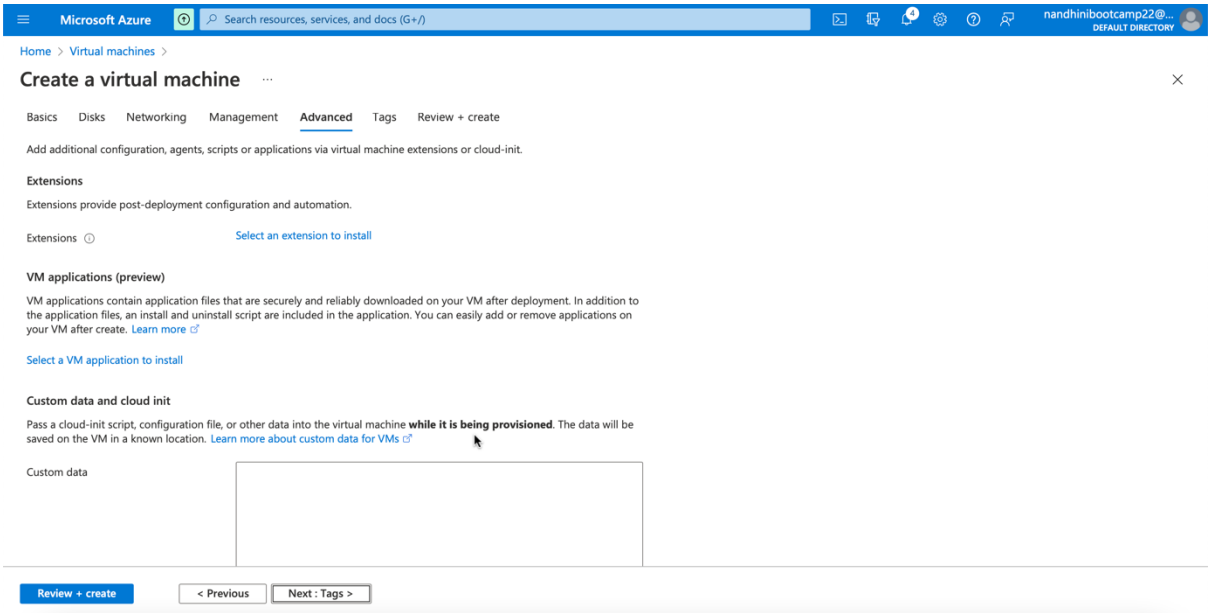
f. **Management:** Leave the settings as default.

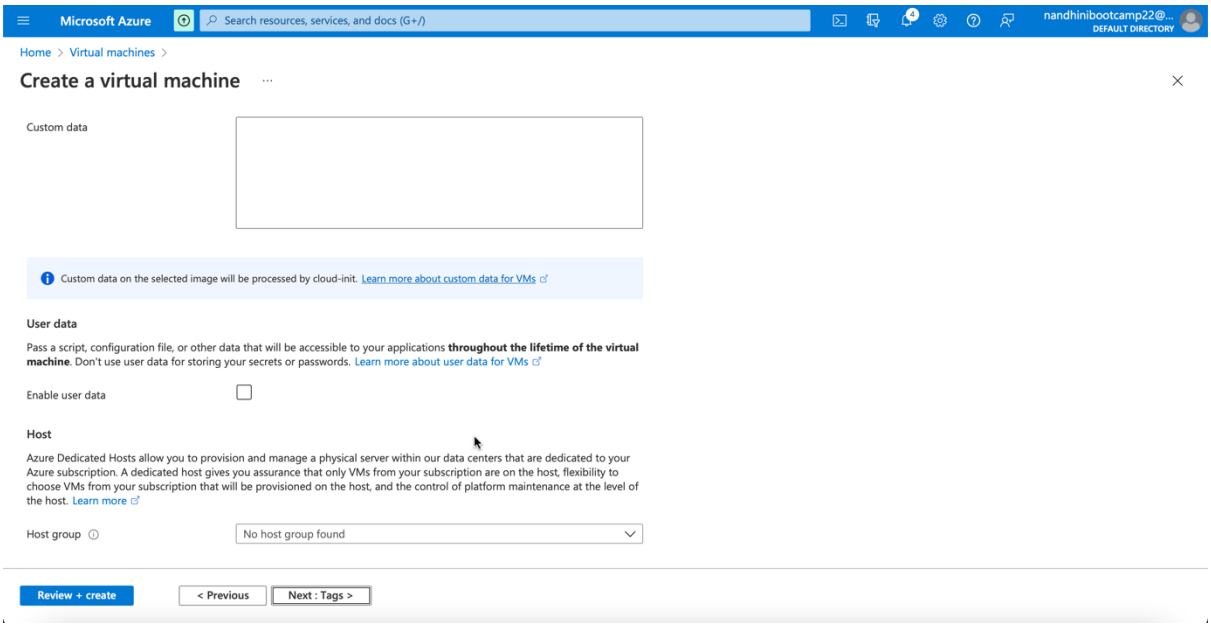


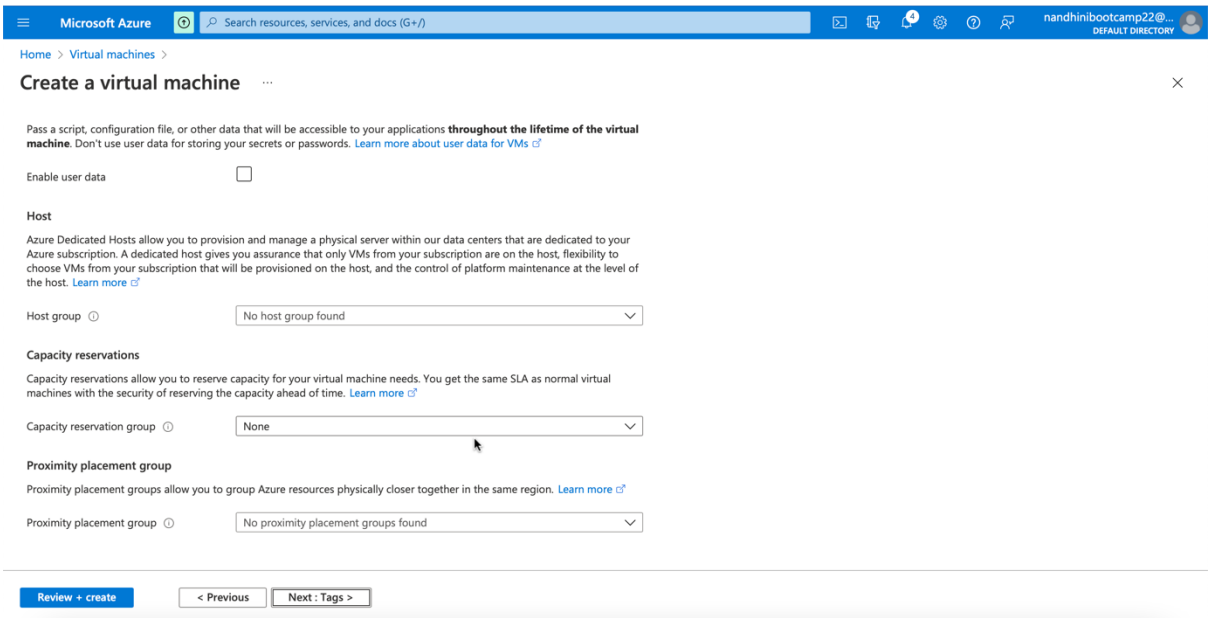


Click Next.

g. **Advanced:** Leave the settings as default.

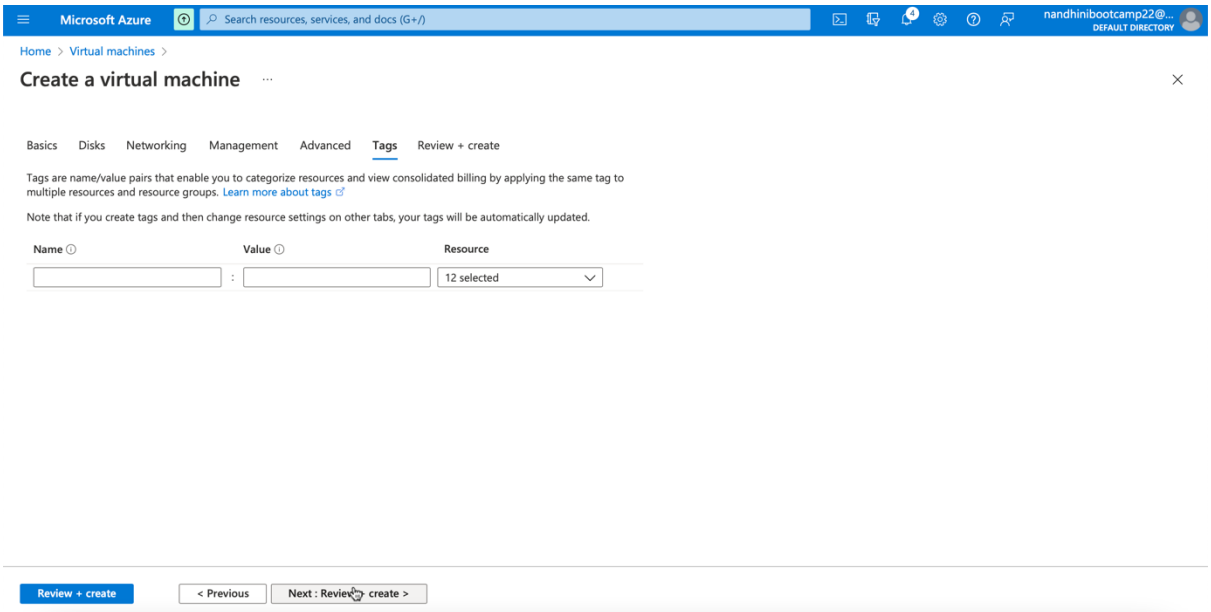




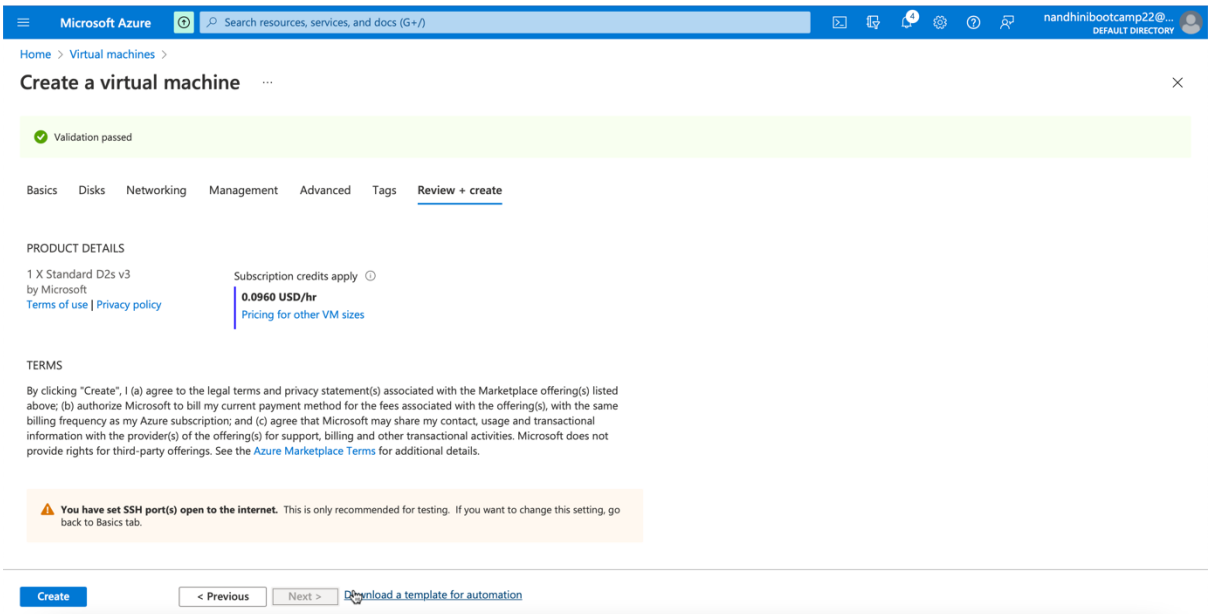


Click Next.

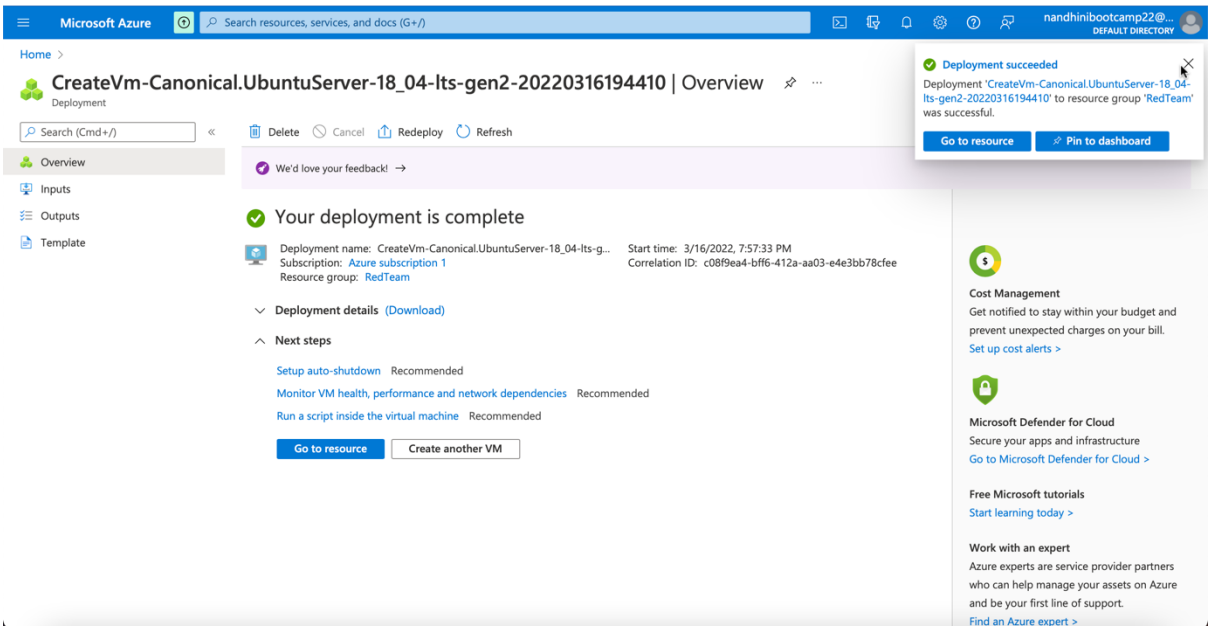
1. **Tags:** Leave the settings as default.



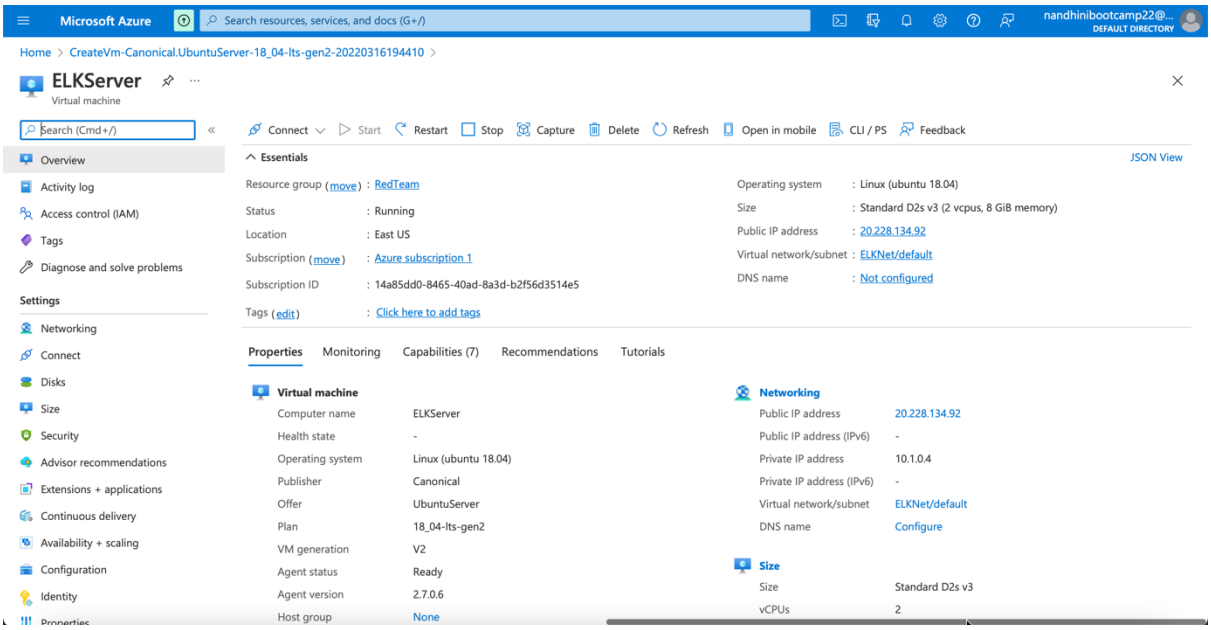
Click Review + create.



Click Create.



Click Go to resource to view the ELKServer VM.



2. To verify if the VM was created correctly, SSH into the VM using the private IP address from the Ansible container on jump box.

Chapter 3: Downloading and Configuring the Container

Using Ansible, we can configure the new VM to function as an ELK Server.

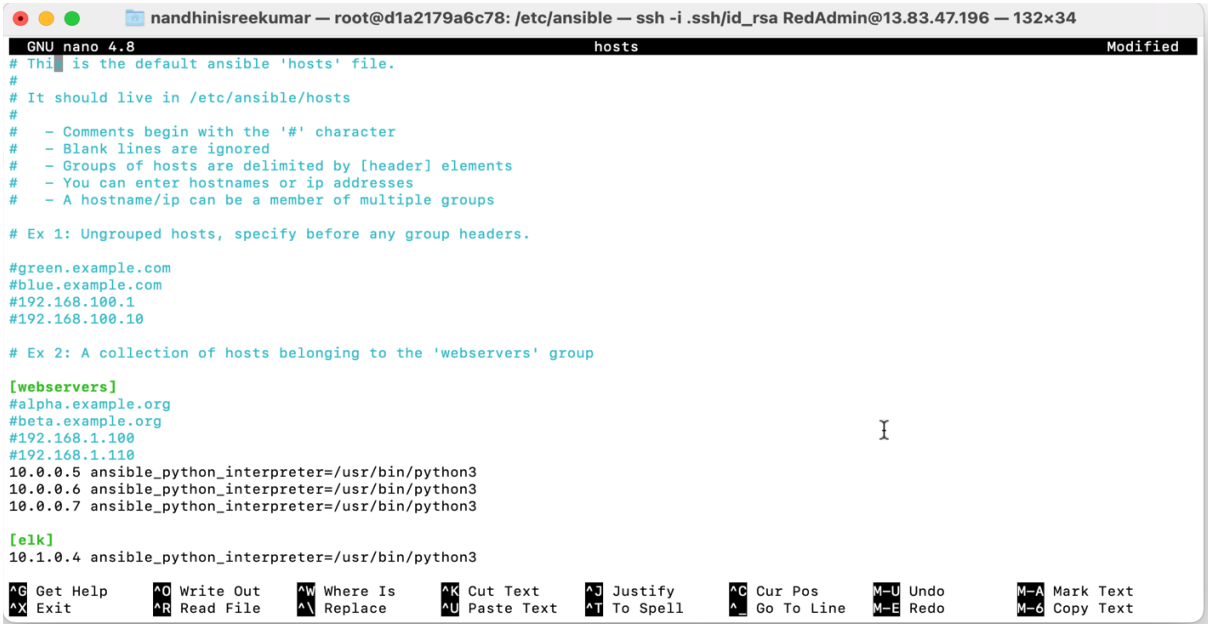
1. From the Ansible VM, change to /etc/ansible/ directory.

   cd /etc/ansible/

2. Add the new VM to Ansible's ***hosts*** file for Ansible to understand the list of machines it can discover and connect to.

   nano hosts

3. Add **[elk]** group and IP address of the ELKServer VM.
   We can specify the groups on which the playbooks should be run.

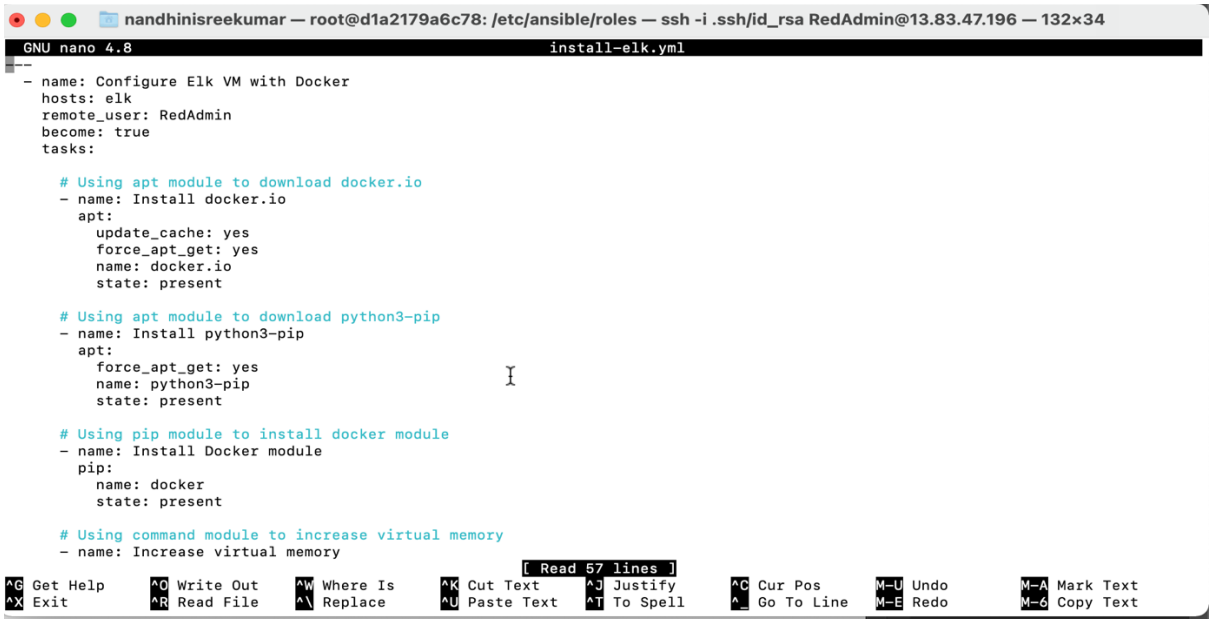


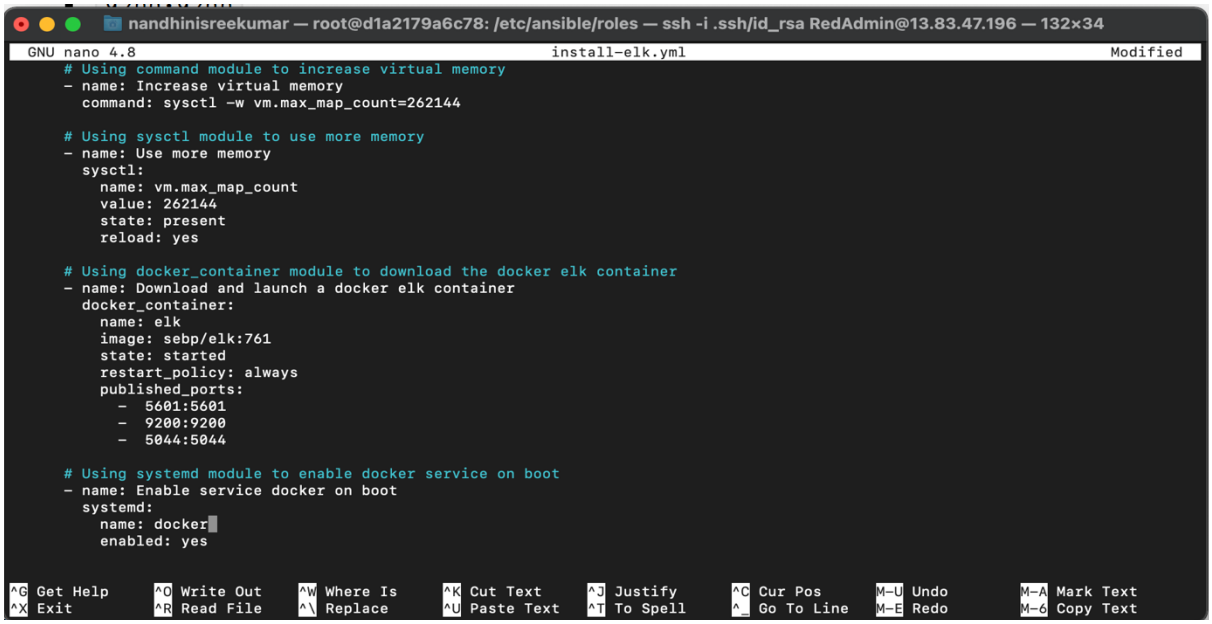
4. Create the playbook to configure the ELK Server. The playbook should run on the **[elk]** group.

   The playbook should:
   - Install docker.io (Docker engine) and python3-pip (Python software installer) packages.
   - Install docker, the Python client for Docker which required by Ansible to control the state of Docker containers.
   - Download the Docker container called sebp/elk:761.
   - Configure the container to start with the following port mappings:
     - 5601:5601
     - 9200:9200
     - 5044:5044
   - Start the container

- Enables the docker service on boot

nano install-elk.yml

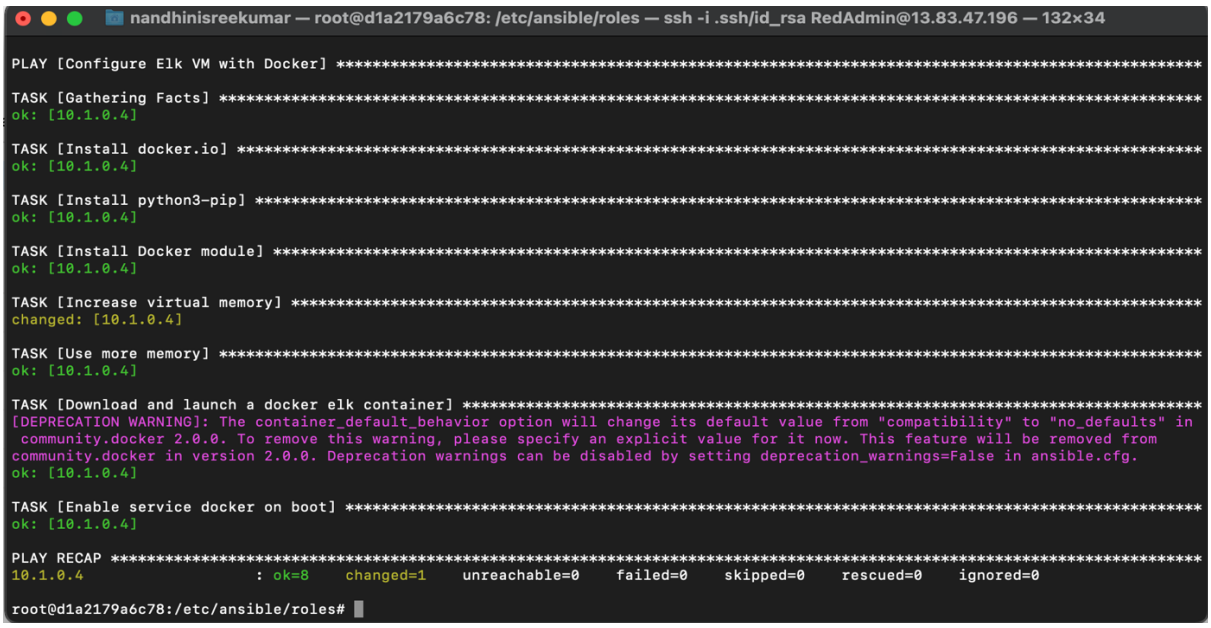




Chapter 4: Launching and Exposing the Container

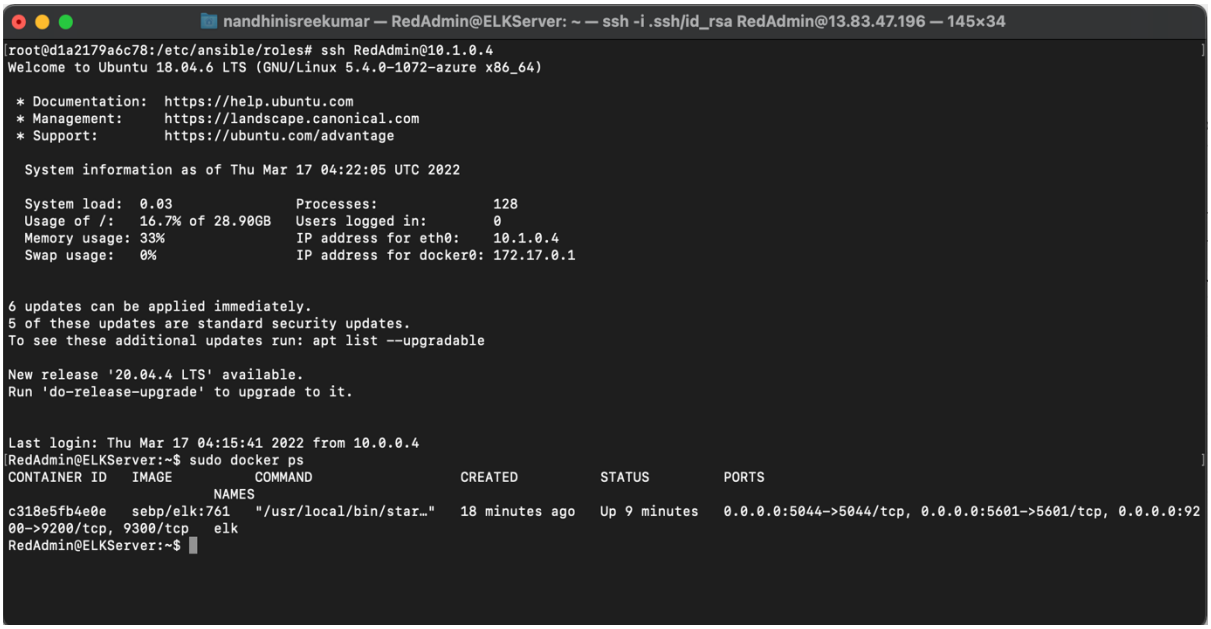
1. Run the playbook

   ansible-playbook -m install-elk.yml



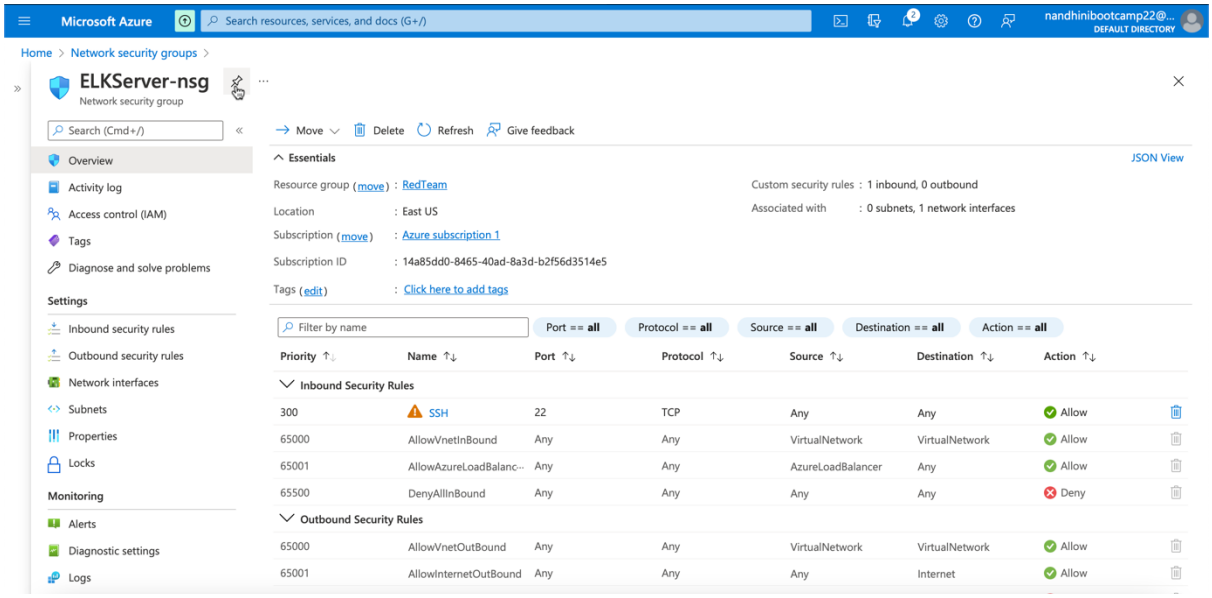
2. SSH into the ELK Server VM and ensure **sebp/elk:761** container is running.

   docker ps

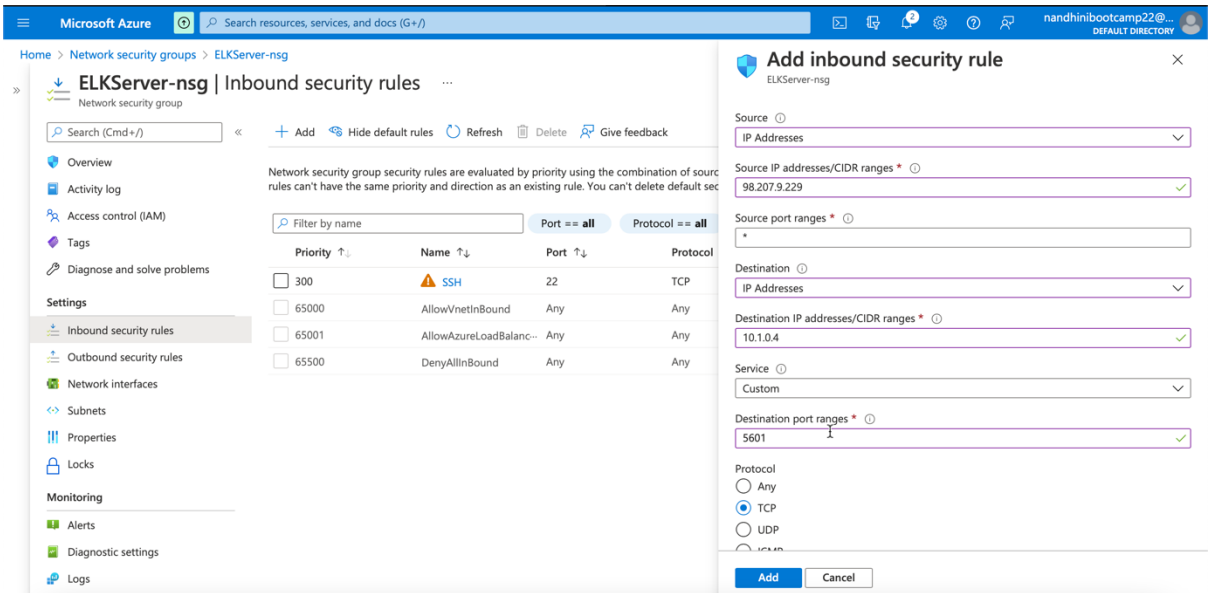


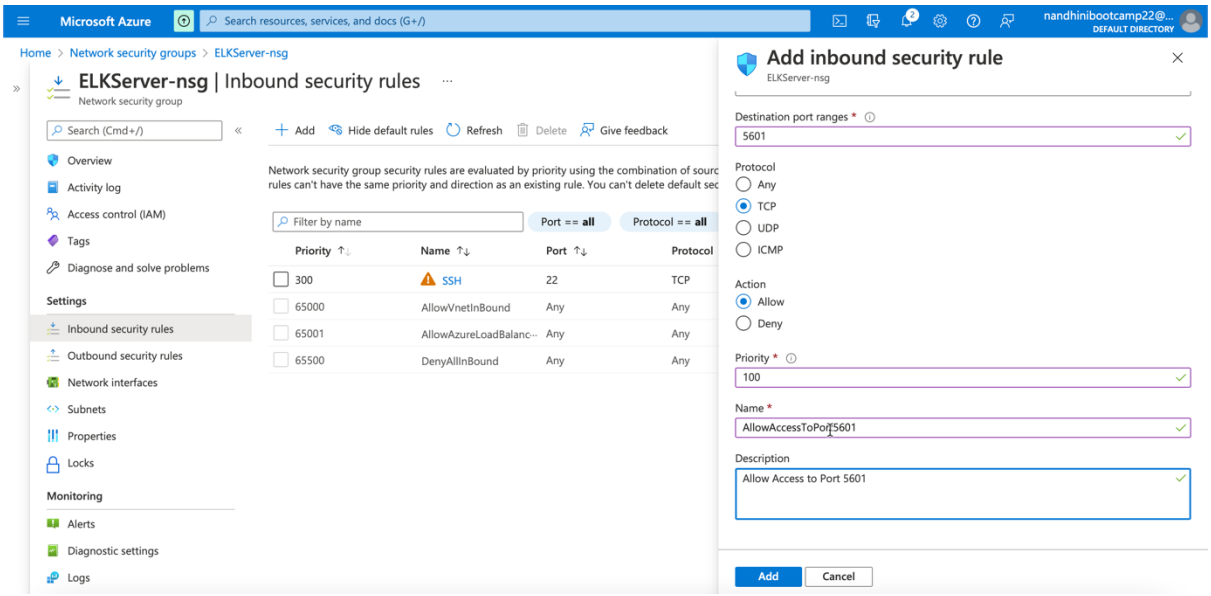
Chapter 5: Identity and Access Management

1. On the Azure portal, navigate to network security groups.

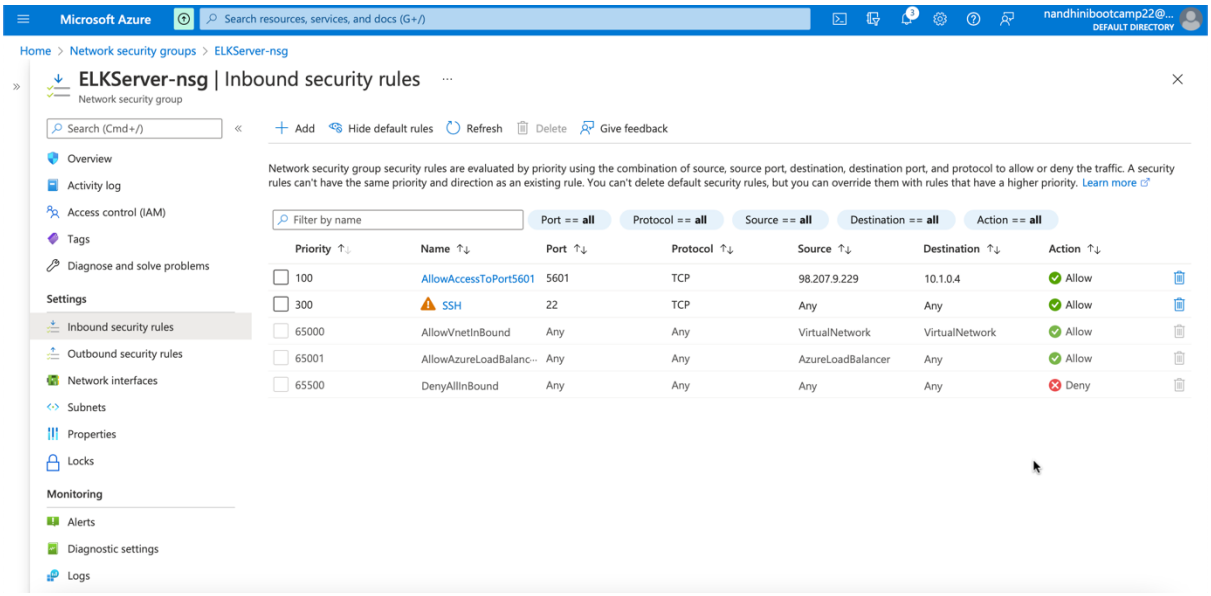


2. Create an inbound security rule in the network security group to allow traffic over TCP 5601 from the workstation public IP to the ELK Server and to restrict other access to the ELK Server.

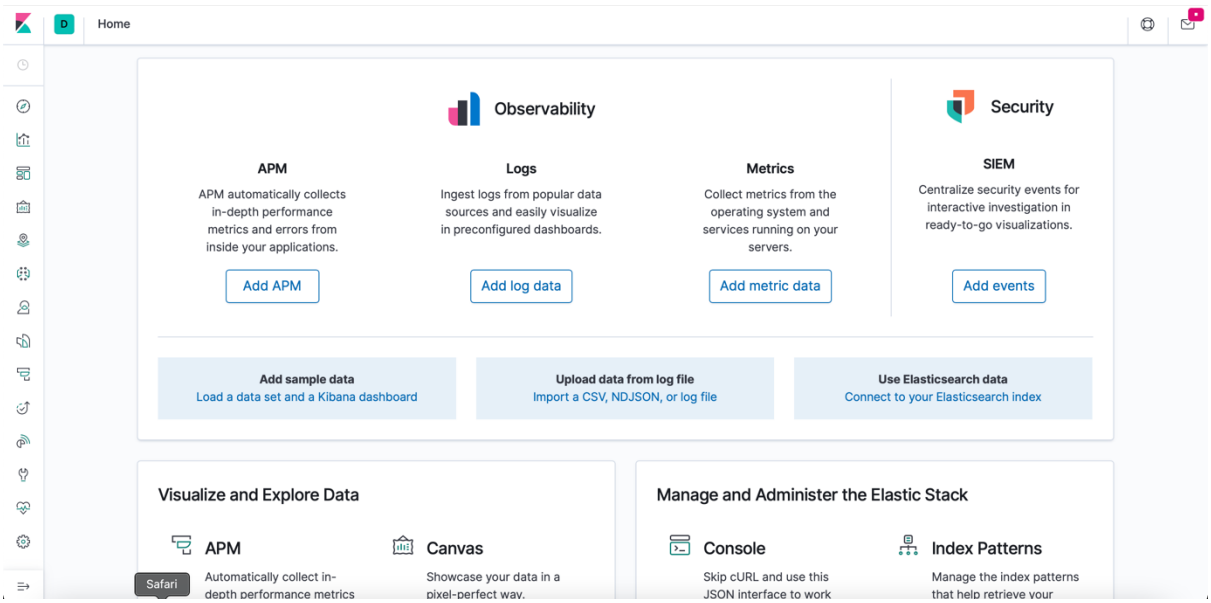




3. Click Add.

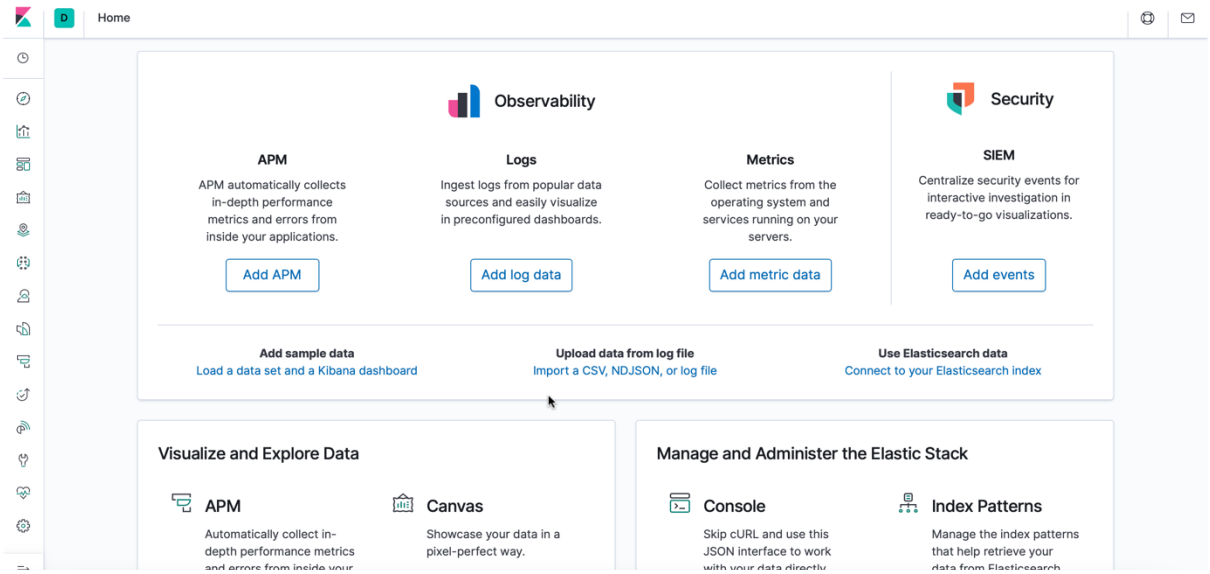


4. Navigate to [http://[ELK-VM.External.IP]:5601/app/kibana](http://[ELK-VM.External.IP]:5601/app/kibana) to verify access to ELK Server.

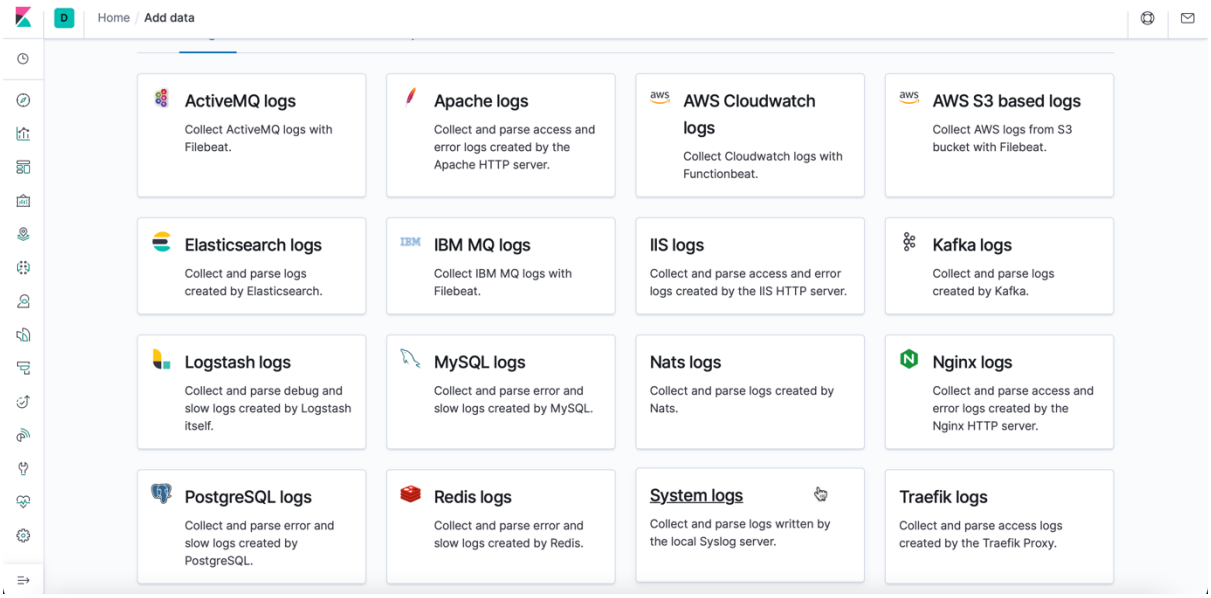

Home
Observability
APM
APM automatically collects in-depth performance metrics and errors from inside your applications.
Add APM
Logs
Ingest logs from popular data sources and easily visualize in preconfigured dashboards.
Add log data
Metrics
Collect metrics from the operating system and services running on your servers.
Add metric data
Security
SIEM
Centralize security events for interactive investigation in ready-to-go visualizations.
Add events
Add sample data
Load a data set and a Kibana dashboard
Upload data from log file
Import a CSV, NDJSON, or log file
Use Elasticsearch data
Connect to your Elasticsearch index
Visualize and Explore Data
APM
Automatically collect in-depth performance metrics
Canvas
Showcase your data in a pixel-perfect way.
Manage and Administer the Elastic Stack
Console
Skip cURL and use this JSON interface to work
Index Patterns
Manage the index patterns that help retrieve your
Safari

Chapter 6: Installing Filebeat on the DVWA Container

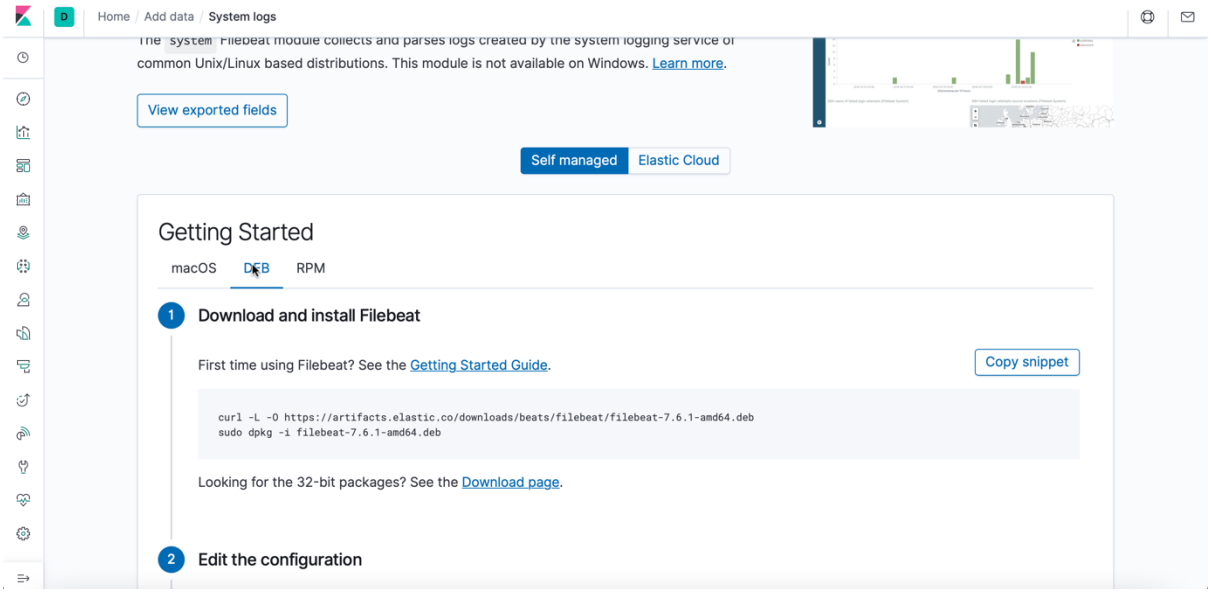
1. Click 'Explore on my Own' on the Kibana server landing page.



2. Click **Add Log Data**.

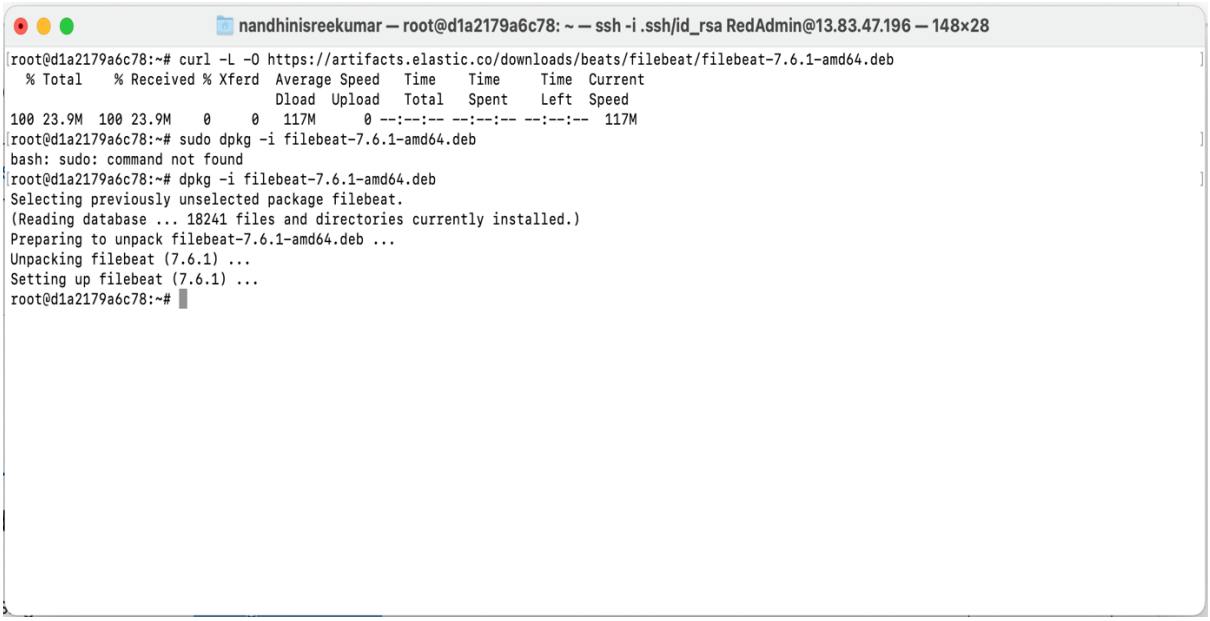


3. Choose System Logs.
   Click on the DEB tab under Getting Started.



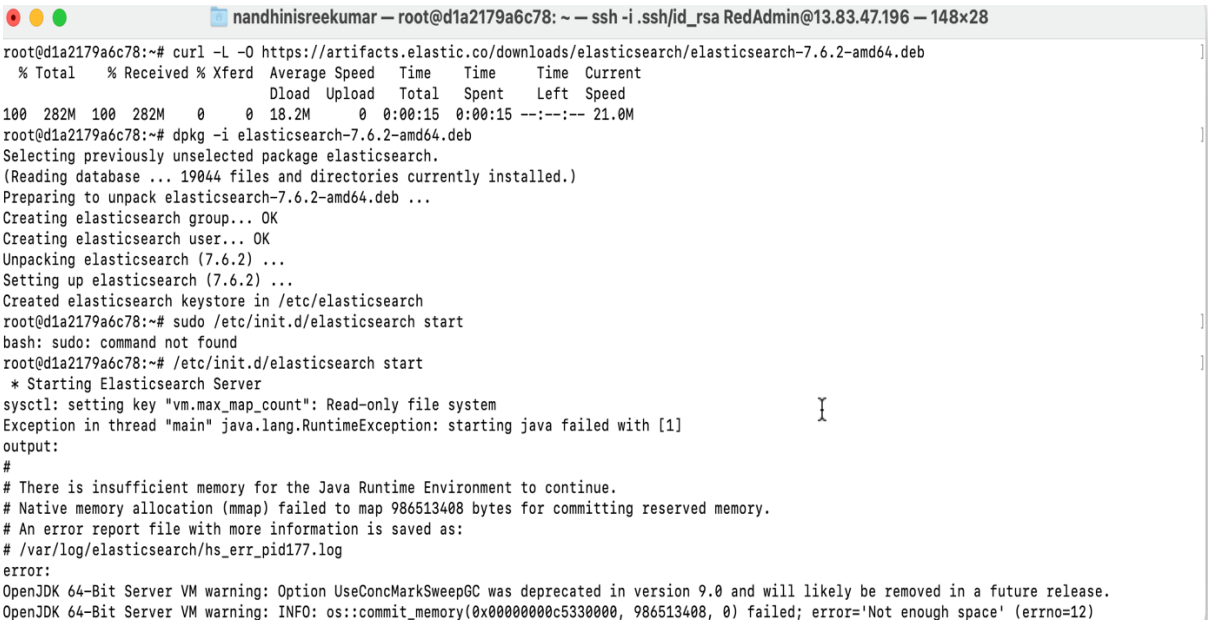
The Filebeat installation instructions can be found here which is used for installing Filebeat using ansible playbook.

4. Install Filebeat using the commands displayed on the Kibana server landing page.



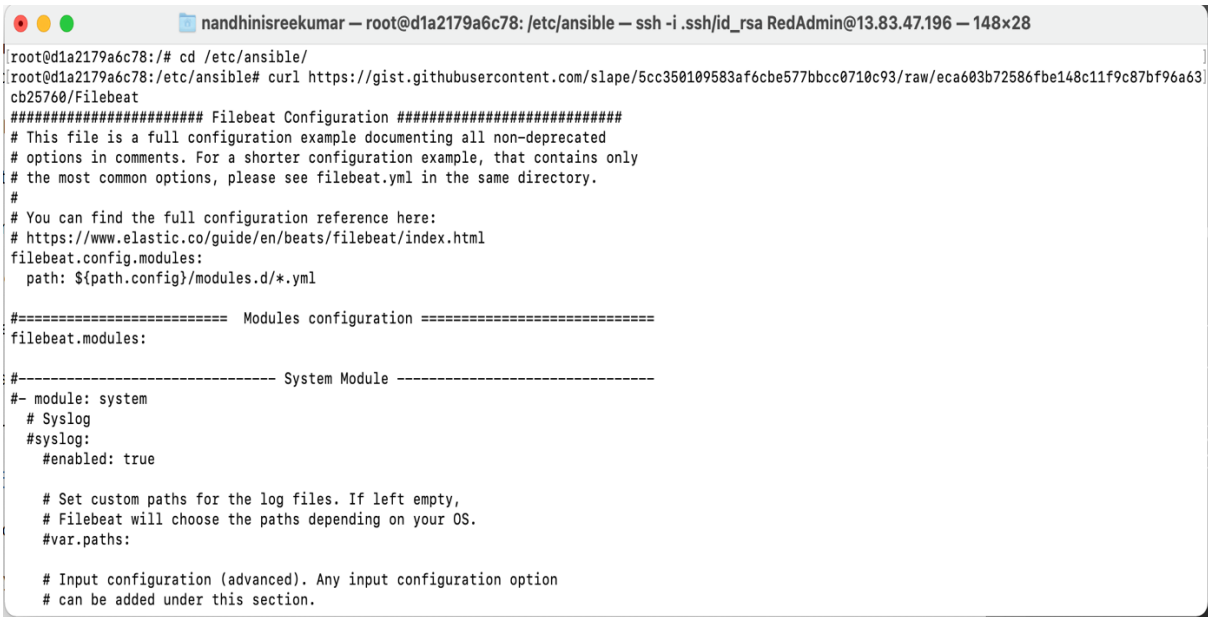
5. Install Elasticsearch using the following commands:

curl -L -O https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-7.6.2-amd64.deb
sudo dpkg -i elasticsearch-7.6.2-amd64.deb
sudo /etc/init.d/elasticsearch start



6. SSH into your Jumpbox.
7. Start and attach to the docker container.
8. Run the following curl command to get the filebeat configuration file:

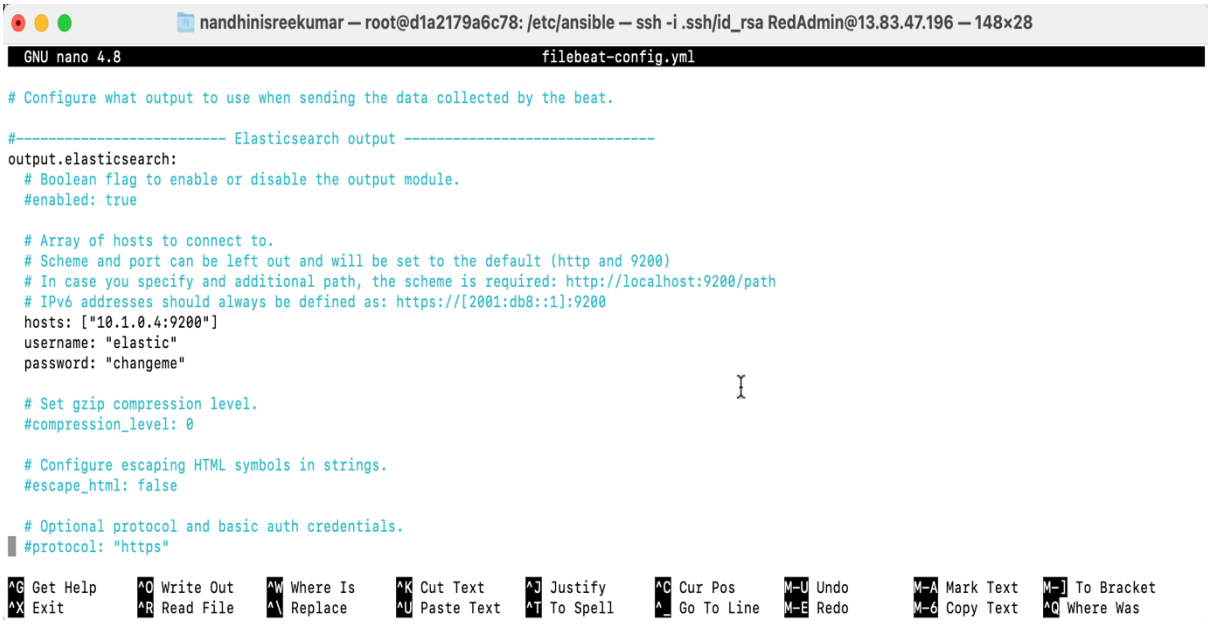
curl https://gist.githubusercontent.com/slape/5cc350109583af6cbe577bbcc0710c93/raw/eca603b72586fbe148c11f9c87bf96a63cb25760/Filebeat



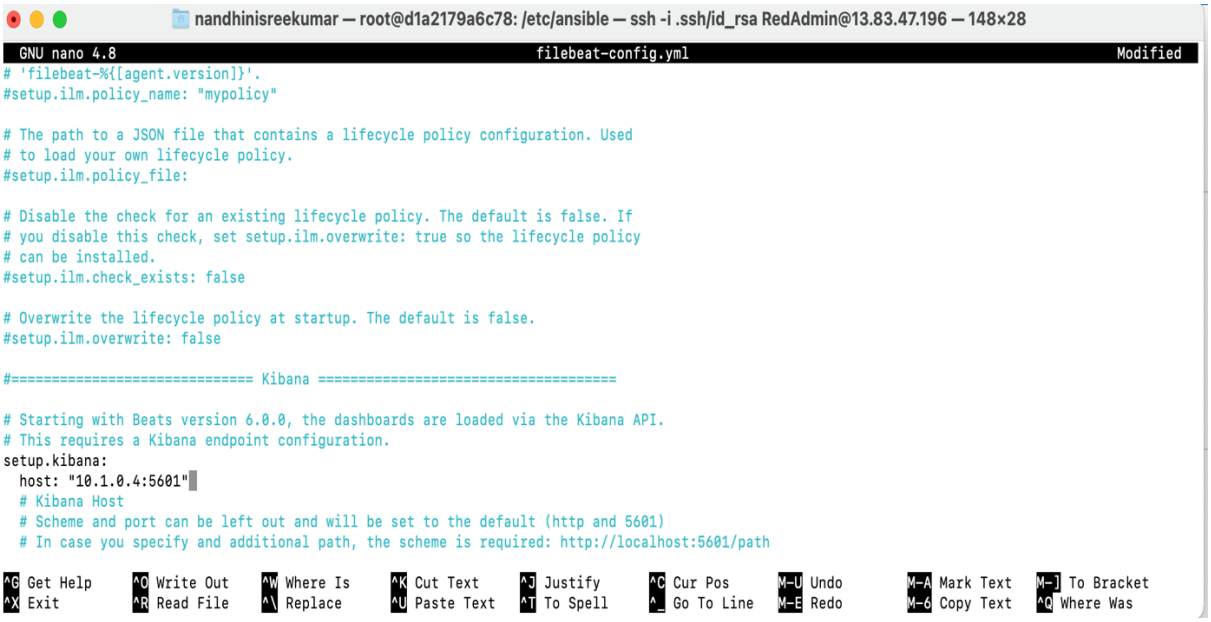
We can also use the Filebeat configuration file template provided.

9. Copy the content to a new file (**filebeat-config.yml**)

9.1 Edit the username and password to **elastic** and **changeme** respectively in the configuration file. Also, replace the IP address with the IP address of your ELK Server VM.



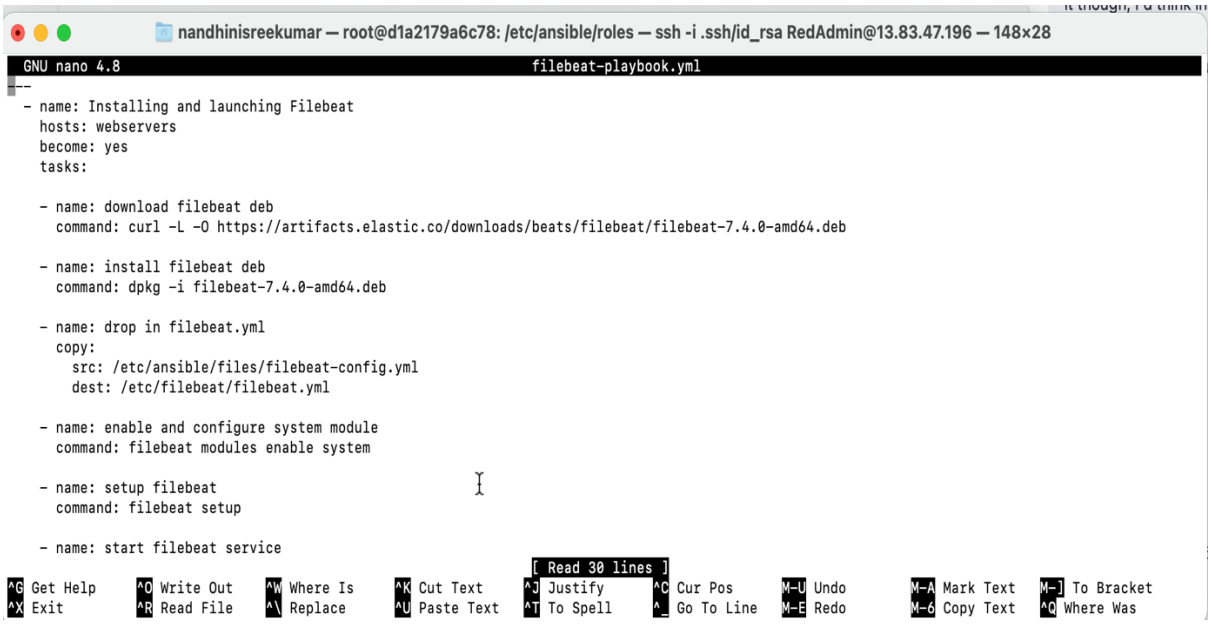
9.2 Scroll to line #1806 (setup.kibana) and replace the IP address with the IP address of your ELK machine.

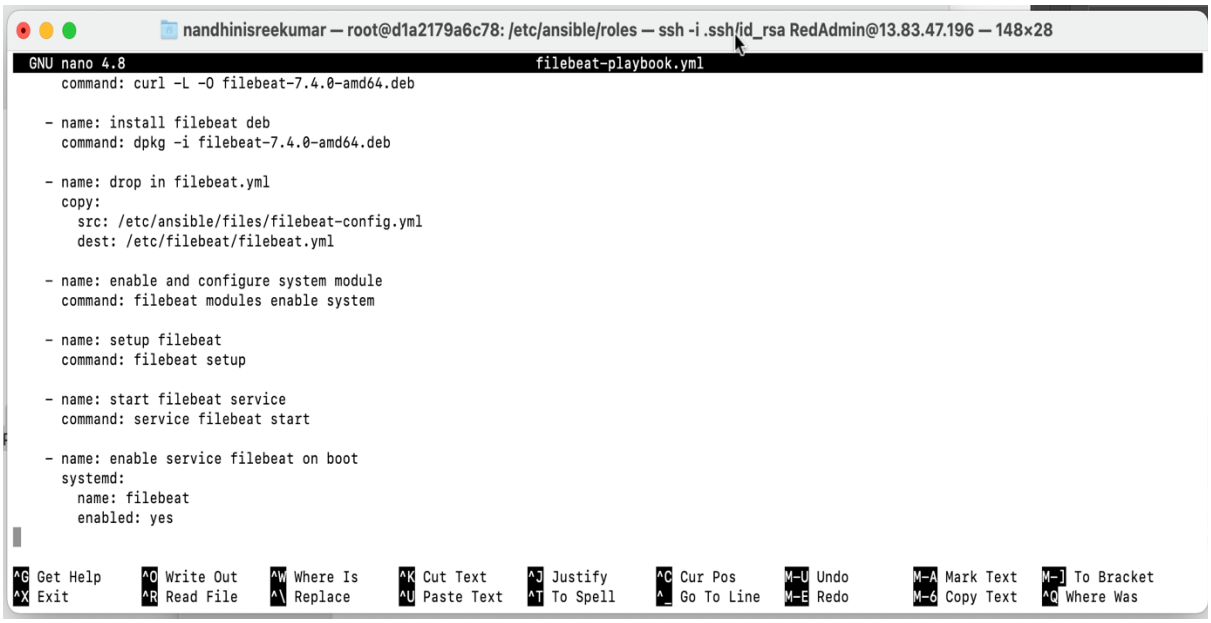


Chapter 7: Creating the Filebeat Installation Play

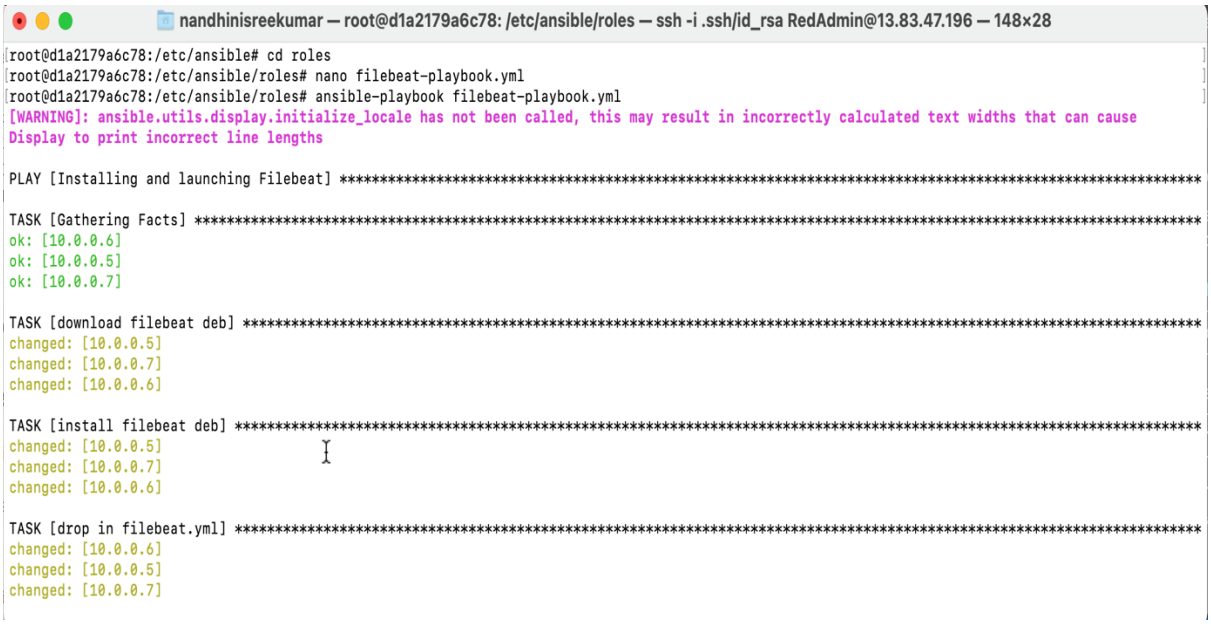
1. Create a playbook to install Filebeat in /etc/ansible/roles/ directory. (**filebeat-playbook.yml**) with the following tasks:

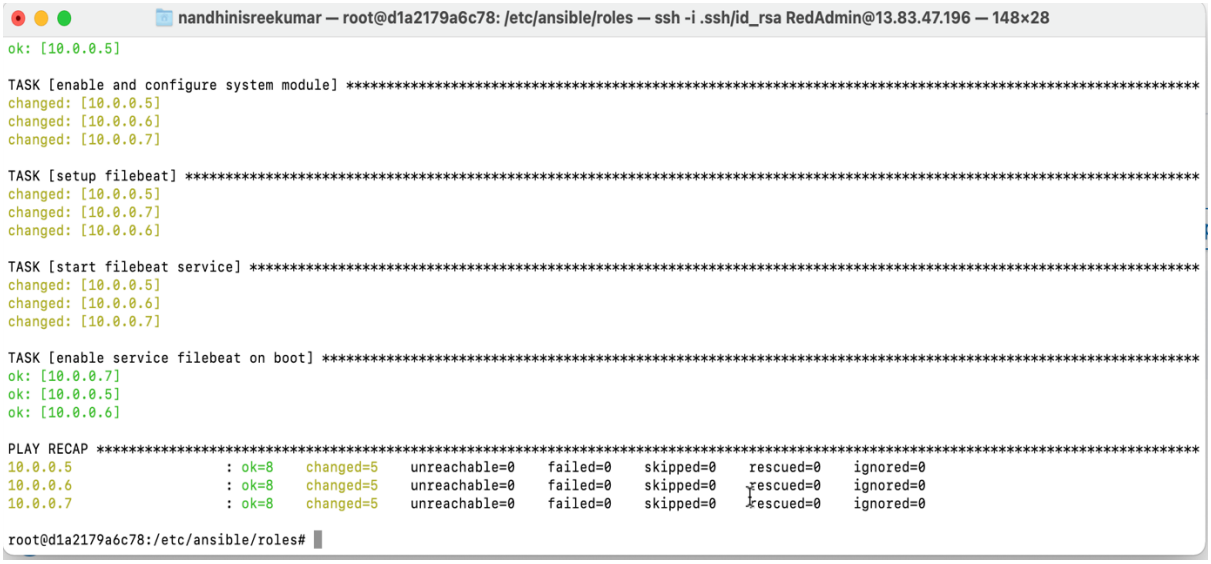
    - Download the .deb file from artifacts.elastic.co.
    - Install the .deb file.
    - Copy the Filebeat configuration file from the Ansible container to the WebVM(s) Filebeat was installed.
    - Enable and configure the system modules
    - Setup Filebeat
    - Start Filebeat service
    - Enable the Filebeat service on boot.





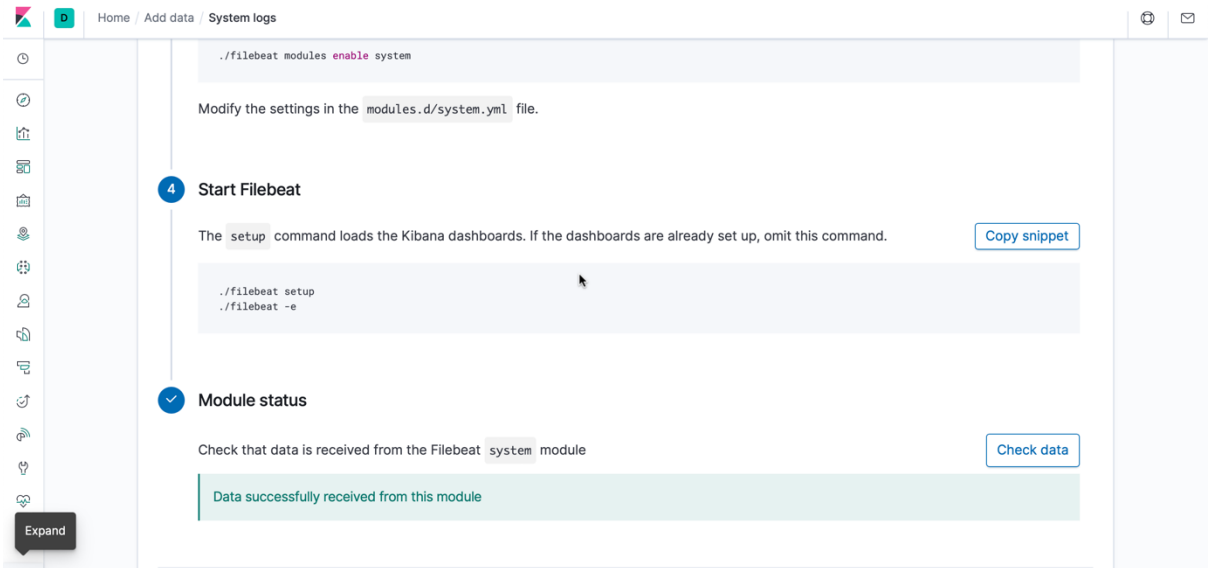
2. Save the file.
3. Run the playbook.


nandhinisreekumar — root@d1a2179a6c78: /etc/ansible/roles — ssh -i .ssh/id_rsa RedAdmin@13.83.47.196 — 148×28
root@d1a2179a6c78:/etc/ansible# cd roles
root@d1a2179a6c78:/etc/ansible/roles# nano filebeat-playbook.yml
root@d1a2179a6c78:/etc/ansible/roles# ansible-playbook filebeat-playbook.yml
[WARNING]: ansible.utils.display.initialize_locale has not been called, this may result in incorrectly calculated text widths that can cause
Display to print incorrect line lengths
PLAY [Installing and launching Filebeat] ***
TASK [Gathering Facts] ***
ok: [10.0.0.6]
ok: [10.0.0.5]
ok: [10.0.0.7]
TASK [download filebeat deb] ***
changed: [10.0.0.5]
changed: [10.0.0.7]
changed: [10.0.0.6]
TASK [install filebeat deb] ***
changed: [10.0.0.5]
changed: [10.0.0.7]
changed: [10.0.0.6]
TASK [drop in filebeat.yml] ***
changed: [10.0.0.6]
changed: [10.0.0.5]
changed: [10.0.0.7]


nandhinisreekumar — root@d1a2179a6c78: /etc/ansible/roles — ssh -i .ssh/id_rsa RedAdmin@13.83.47.196 — 148×28
ok: [10.0.0.5]
TASK [enable and configure system module] ***
changed: [10.0.0.5]
changed: [10.0.0.6]
changed: [10.0.0.7]
TASK [setup filebeat] ***
changed: [10.0.0.5]
changed: [10.0.0.7]
changed: [10.0.0.6]
TASK [start filebeat service] ***
changed: [10.0.0.5]
changed: [10.0.0.6]
changed: [10.0.0.7]
TASK [enable service filebeat on boot] ***
ok: [10.0.0.7]
ok: [10.0.0.5]
ok: [10.0.0.6]
PLAY RECAP ***
10.0.0.5 : ok=8 changed=5 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
10.0.0.6 : ok=8 changed=5 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
10.0.0.7 : ok=8 changed=5 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
root@d1a2179a6c78:/etc/ansible/roles#

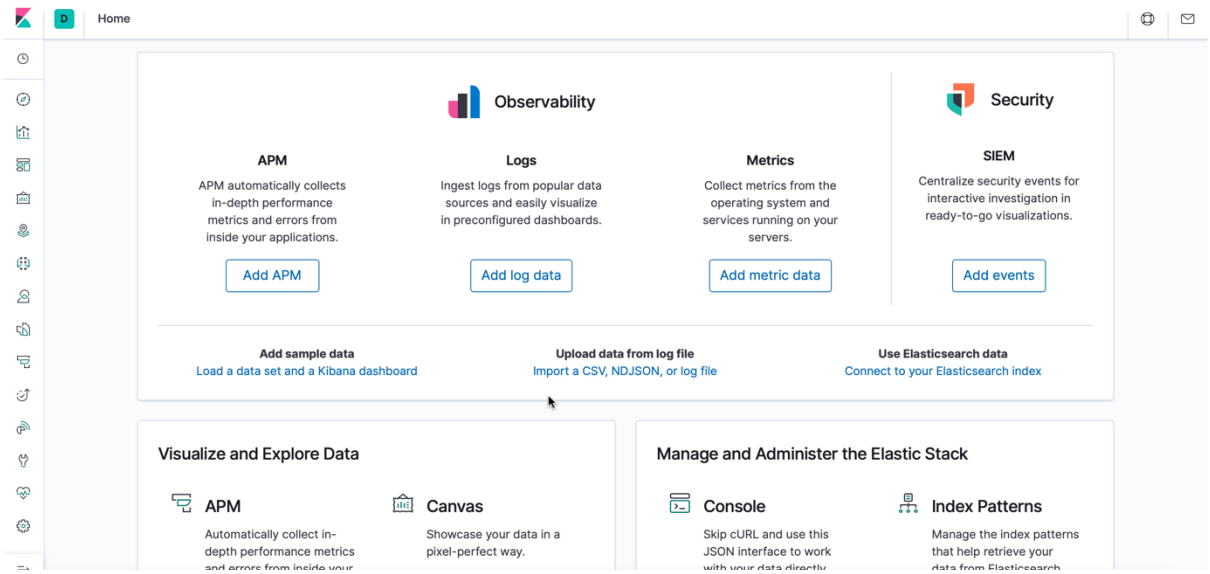
Chapter 8: Verifying Installation and Playbook

1. Navigate to the Filebeat installation page on Kibana.
2. On the same page, scroll to **Step 5: Module Status** and **click Check Data**.
3. Scroll to the bottom of the page and click **Verify Incoming Data.**

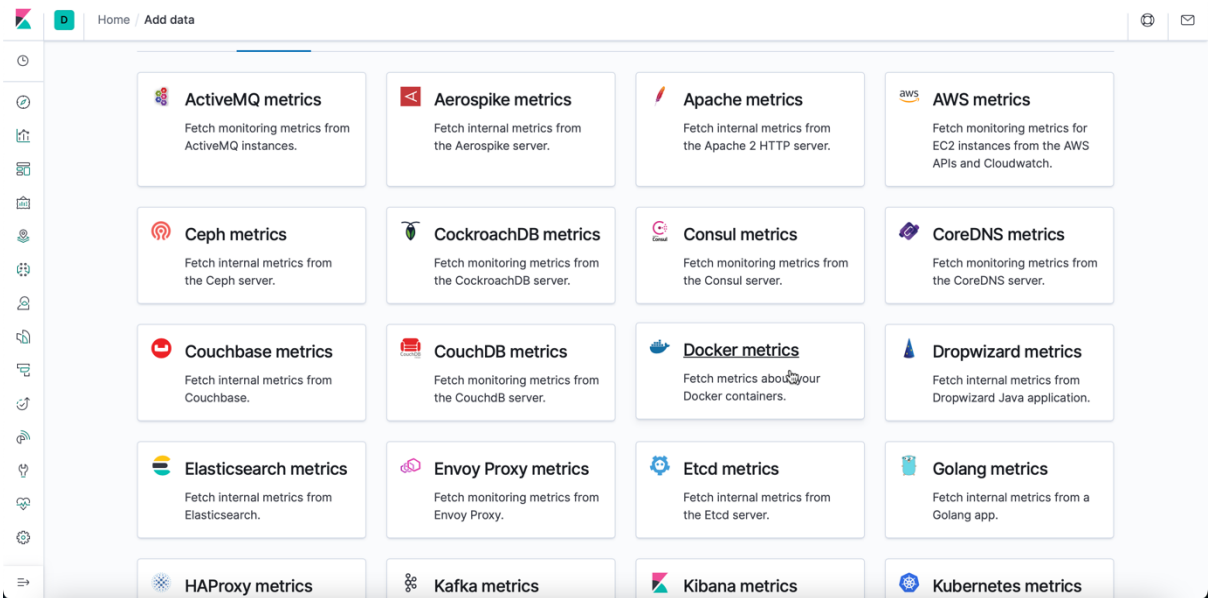


Chapter 9: Creating a Play to Install Metricbeat

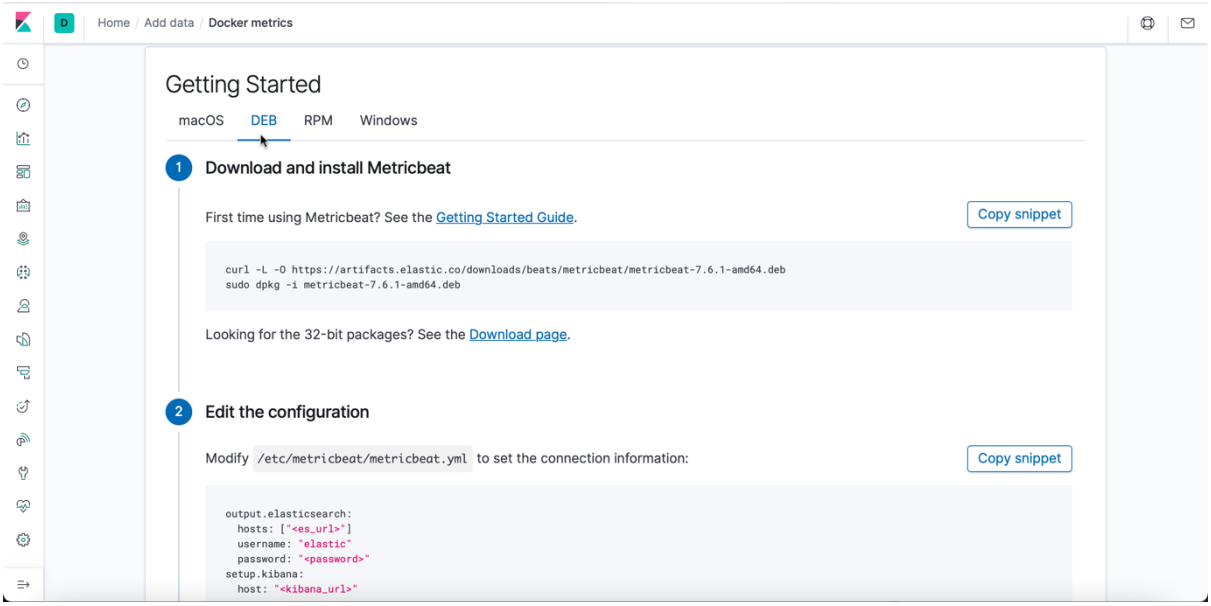
1. Click 'Explore on my Own' on the Kibana server landing page.



2. Click Add Metric Data.

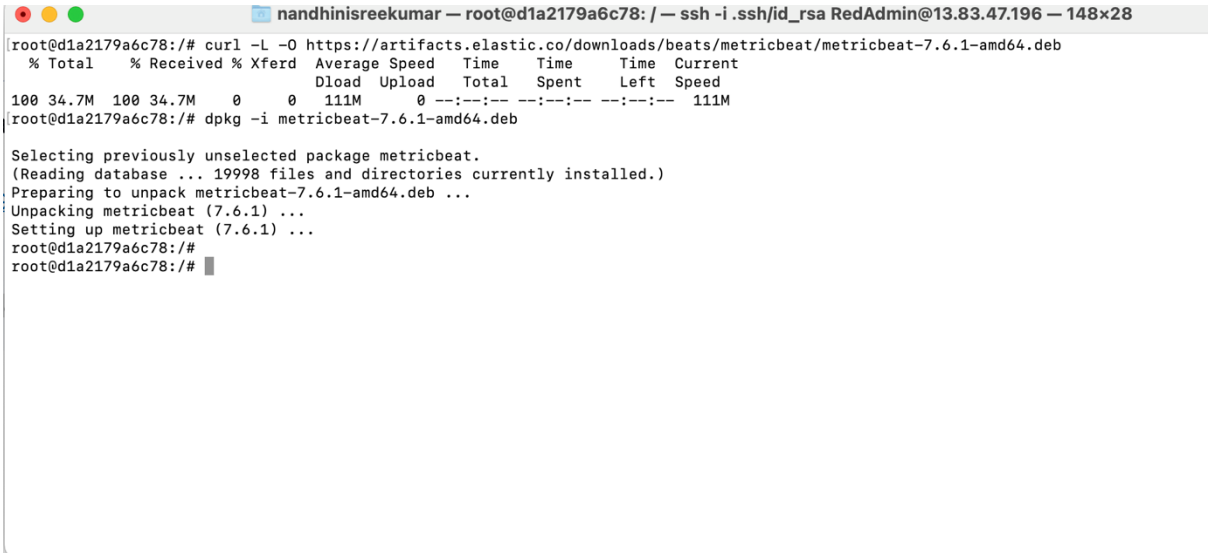


3. Click **Docker Metrics**.
   Click the **DEB** tab under Getting Started.



The Metricbeat installation instructions can be found here which is used for installing Metricbeat using ansible playbook.

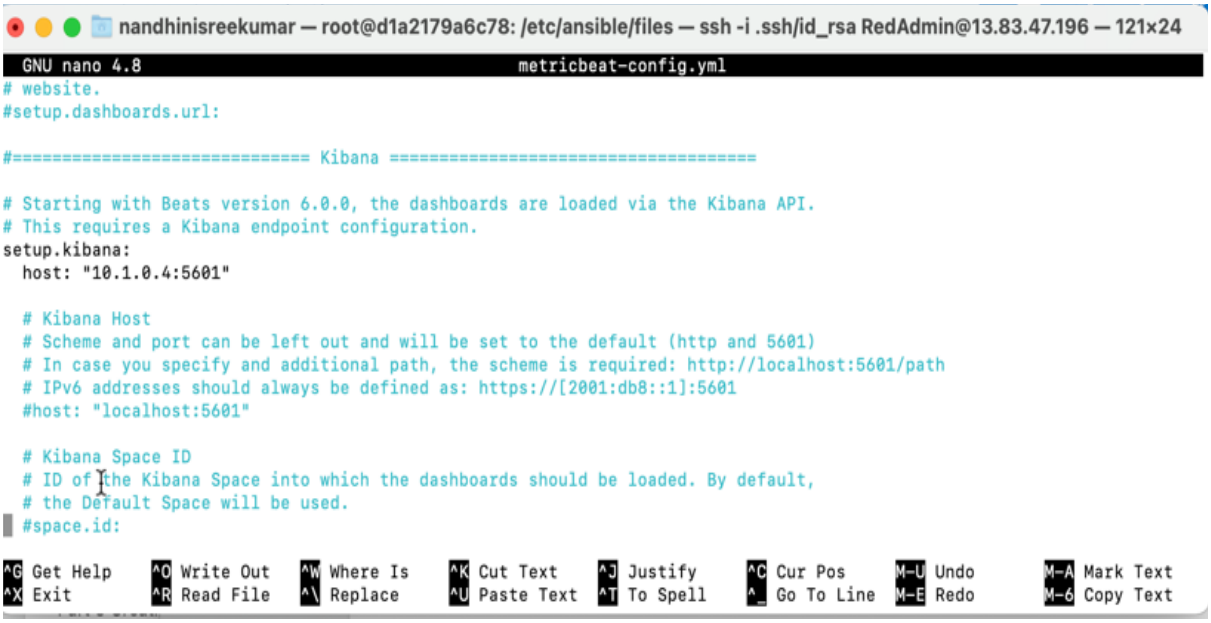
4. Install Metricbeat using the commands displayed on the Kibana server landing page.



5. SSH into your Jumpbox.
6. Start and attach to the docker container.
7. Copy the content of the Metricbeat configuration file template to a new file (**metricbeat-config.yml**)

7.1 Edit the username and password to **elastic** and **changeme** respectively in the configuration file. Also, replace the IP address with the IP address of your ELK Server VM.

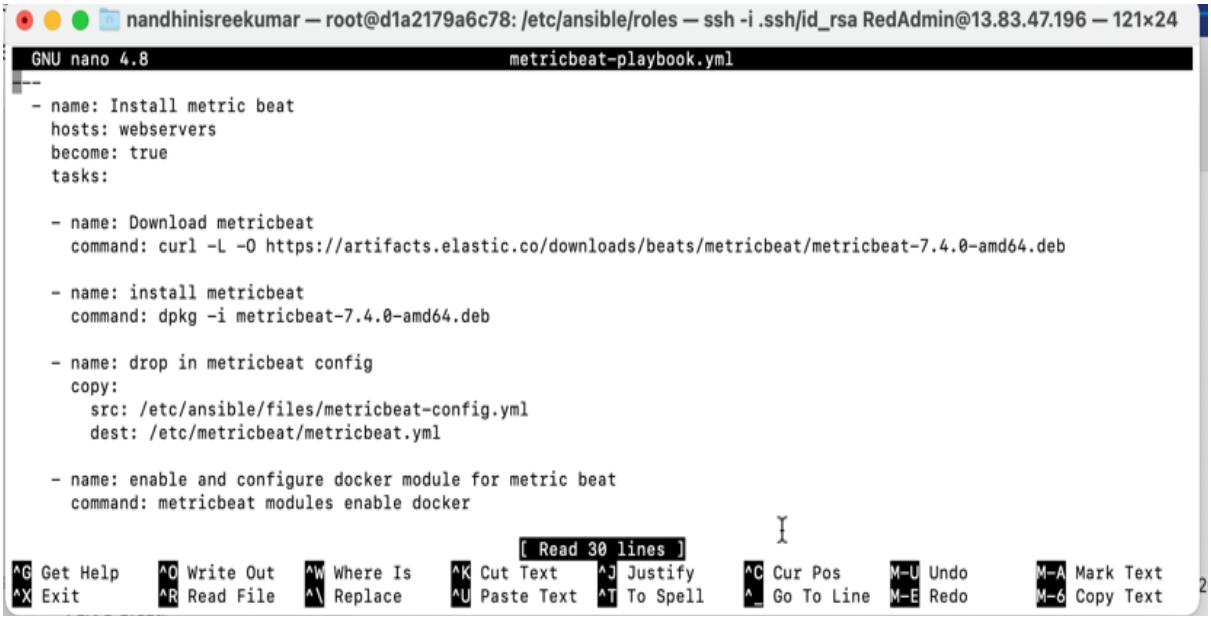
7.2 Scroll to line #1806 (setup.kibana) and replace the IP address with the IP address of your ELK machine.

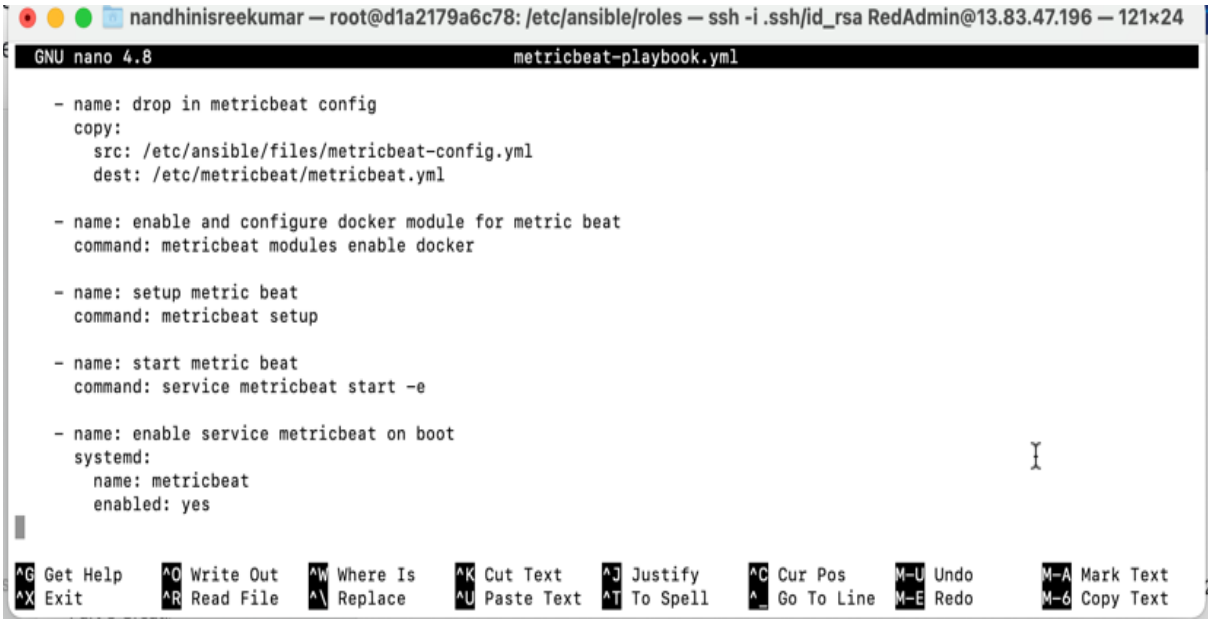


8. Create a playbook to install Filebeat in /etc/ansible/roles/ directory. (**filebeat-playbook.yml**) with the following tasks:

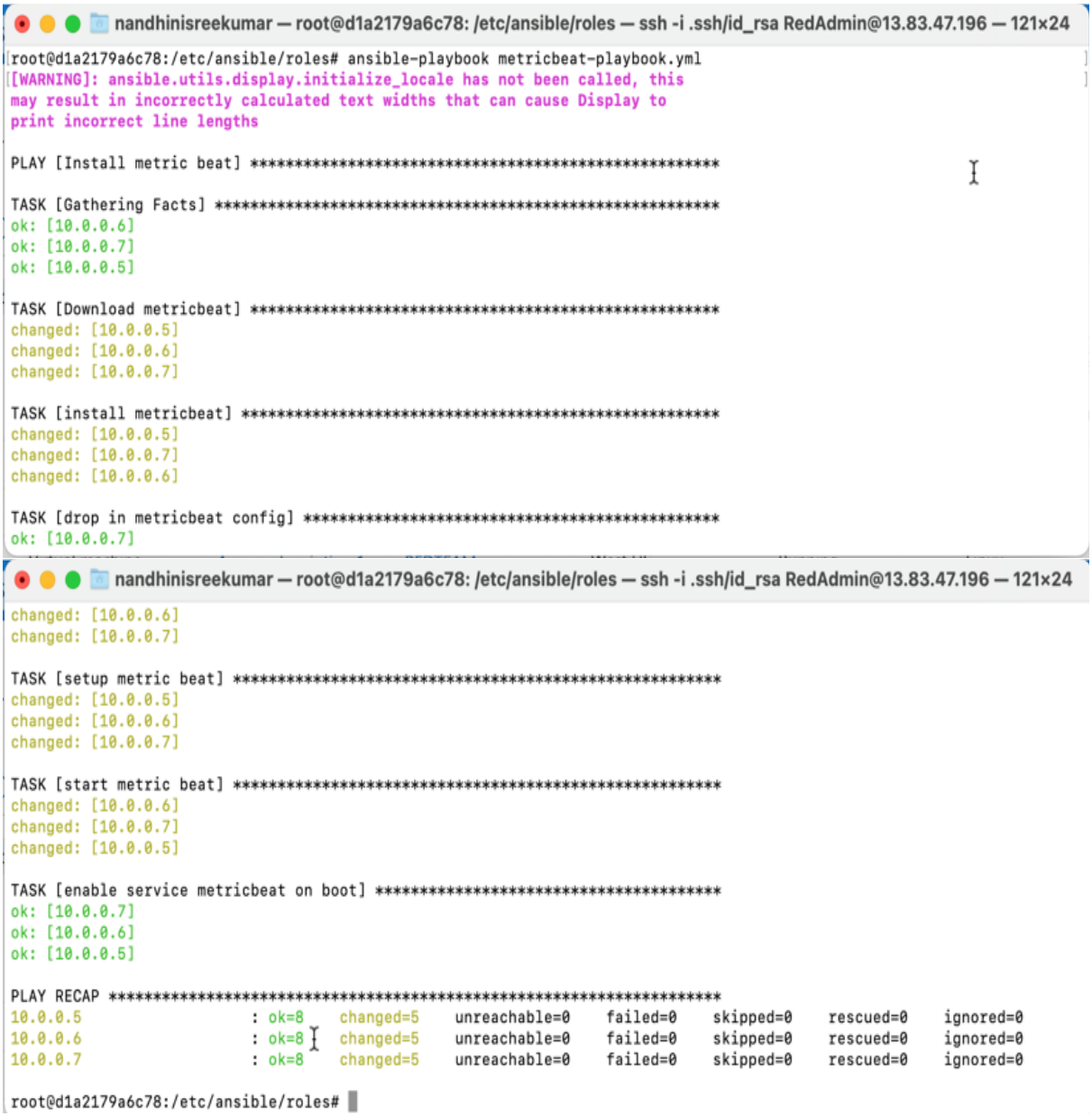
    - Download the .deb file from artifacts.elastic.co.
    - Install the .deb file.

- Copy the Metribeat configuration file from the Ansible container to the WebVM(s) Metricbeat was installed.
- Enable and configure the docker module for metricbeat
- Setup Metricbeat
- Start Metricbeat service
- Enable the Metricbeat service on boot

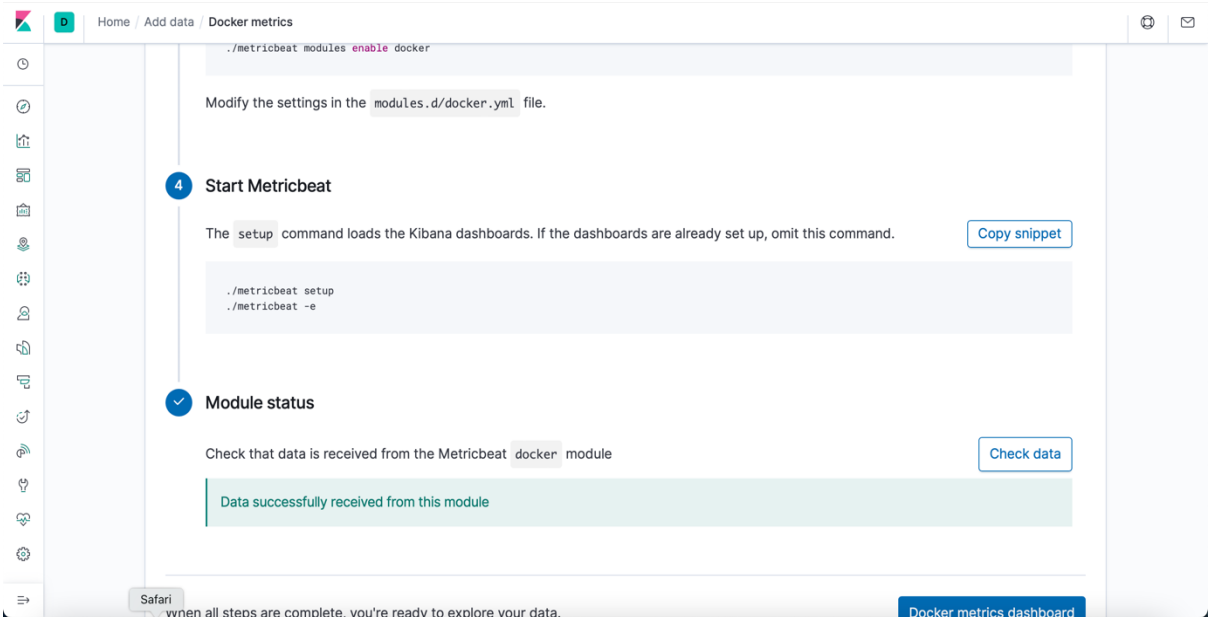




9. Save the file.
10. Run the playbook.



11. Navigate to the Filebeat installation page on Kibana.
12. On the same page, scroll to **Step 5: Module Status** and **click Check Data**.
13. Scroll to the bottom of the page and click **Verify Incoming Data.**


Home / Add data / Docker metrics
./metricbeat modules enable docker
Modify the settings in the modules.d/docker.yml file.
4
Start Metricbeat
The setup command loads the Kibana dashboards. If the dashboards are already set up, omit this command.
Copy snippet
./metricbeat setup
./metricbeat -e
Module status
Check that data is received from the Metricbeat docker module
Check data
Data successfully received from this module
Safari
Docker metrics dashboard