

# ELK Task

Yehia A. Mostafa

## Elastic Stack Setup with Fleet Server and Agent Enrollment

### 1. SSH Access to Ubuntu Server (Better Utility)

Instead of using VirtualBox console or similar, I used **SSH from Windows PowerShell** for better scrolling, search, and copy/paste capabilities.

**Command:**

```
ssh username@<Ubuntu_Server_IP>
```

- **username** → Your Ubuntu VM's username.
- **<Ubuntu\_Server\_IP>** → IP address of the VM (reachable from your host).

### 2. Installing Elasticsearch and Kibana

I already downloaded the `.deb` packages to my **Windows host**, transferred them to the Ubuntu server using **WinSCP (SCP Protocol)**, and then installed them.

#### 2.1 Install Elasticsearch

```
sudo dpkg -i elasticsearch-<version>-amd64.deb
```

- Installs Elasticsearch from the `.deb` file.

#### 2.2 Configure Elasticsearch

```
sudo nano /etc/elasticsearch/elasticsearch.yml
```

- Modify

```
network.host: 0.0.0.0
```

This allows accessing the server on all the interfaces (all IPs assigned to the adapters on the server machine)

## 2.3 Start Elasticsearch

```
sudo systemctl start elasticsearch  
sudo systemctl enable elasticsearch
```

- Starts the service and ensures it launches on boot.

## 2.4 Test Elasticsearch

From your **Windows host browser**:

```
http://<Ubuntu_Server_IP>:9200
```

- If successful, you'll see a **JSON object** with cluster details.
- At installation, Elasticsearch generates a **password** for the default user `elastic` — **memorize or securely store this**.

**If you forget the password:**

**Default `elastic` user**

```
sudo /usr/share/elasticsearch/bin/elasticsearch-reset-password -u elastic
```

- Copy the password exactly (avoid typing to prevent mistakes).

## 2.5 Install Kibana

```
sudo dpkg -i kibana-<version>-amd64.deb
```

## 2.6 Configure Kibana

```
sudo nano /etc/kibana/kibana.yml
```

- Modify:

```
server.host: 0.0.0.0
```

## 2.7 Start Kibana

```
sudo systemctl start kibana  
sudo systemctl enable kibana
```

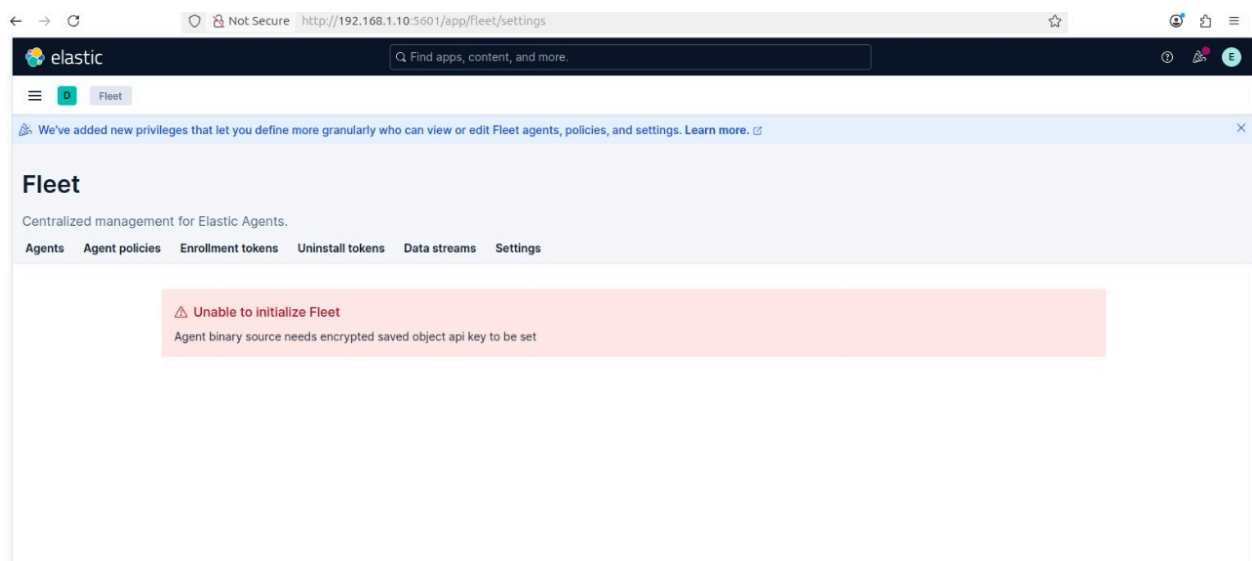
## 2.8 Access Kibana

From your **Windows host browser**:

```
http://<Ubuntu_Server_IP>:5601
```

- Opens the **Kibana GUI**.

## 3. Adding Kibana Security Keys



Kibana needs encryption keys for secure features. You won't be able to access it unless you do the following step.

## Generate Keys

```
sudo /usr/share/kibana/bin/kibana-encryption-keys generate
```

- Outputs **three keys**: `xpack.encryptedSavedObjects.encryptionKey` , `xpack.reporting.encryptionKey` , and `xpack.security.encryptionKey` .

```
ubuntu@server1:~$ sudo /usr/share/kibana/bin/kibana-encryption-keys generate
## Kibana Encryption Key Generation Utility

The 'generate' command guides you through the process of setting encryption keys for:

xpack.encryptedSavedObjects.encryptionKey
  Used to encrypt stored objects such as dashboards and visualizations
  https://www.elastic.co/guide/en/kibana/current/xpack-security-secure-saved-objects.html#xpack-security-secure-saved-objects

xpack.reporting.encryptionKey
  Used to encrypt saved reports
  https://www.elastic.co/guide/en/kibana/current/reporting-settings-kb.html#general-reporting-settings

xpack.security.encryptionKey
  Used to encrypt session information
  https://www.elastic.co/guide/en/kibana/current/security-settings-kb.html#security-session-and-cookie-settings

Already defined settings are ignored and can be regenerated using the --force flag. Check the documentation links for instructions on how to rotate encryption keys.
Definitions should be set in the kibana.yml used to configure Kibana.

Settings:
xpack.encryptedSavedObjects.encryptionKey: 3cae08682864321f2af6963f6b6f9fbb
xpack.reporting.encryptionKey: f43224c1466e00f03681ea88409da263
xpack.security.encryptionKey: a1aef643161ebf904b056ec652af125b

ubuntu@server1:~$ sudo nano /etc/kibana/kibana.yml
ubuntu@server1:~$ sudo systemctl restart kibana.service
ubuntu@server1:~$ sudo systemctl status kibana.service
● kibana.service - Kibana
```

## Add to Config

```
sudo nano /etc/kibana/kibana.yml
```

Copy only the highlighted and paste the keys into the config file under the appropriate fields (anywhere).

## Create Kibana Enrollment Token

```
sudo /usr/share/elasticsearch/bin/elasticsearch-create-enrollment-token -s kibana
```

- Generates a token to connect Kibana to Elasticsearch.

## Get Kibana Verification Code

```
sudo /usr/share/kibana/bin/kibana-verification-code
```

- Used during initial Kibana setup to verify connection.

## 4. Fleet Server Installation

In Kibana GUI → **Fleet** → **Add Fleet Server**, choose:

- **Linux x86\_64 TAR (not .deb)** option.
- Follow the commands provided in the GUI.

Example:

```
tar xzvf elastic-agent-<version>-linux-x86_64.tar.gz
cd elastic-agent-<version>-linux-x86_64

sudo ./elastic-agent install \
  --fleet-server-es=https://<Ubuntu_Server_IP>:9200 \
  --fleet-server-service-token=<token> \
  --fleet-server-policy=<policy_id> \
  --fleet-server-insecure-http
```

## 5. Enrolling Elastic Agent on Any Machine

You can enroll agents from **Windows, Linux, or Mac.....** You must follow the instructions when you choose the software.

### Example: Windows Enrollment

Open **PowerShell (Run as Administrator)**:

```
cd "C:\elastic-agent-<version>-windows-x86_64"
```

```
.\elastic-agent.exe install `
  --url=https://<Fleet_Server_IP>:8220 `
  --enrollment-token=<enrollment_token> `
  --insecure
```

- `--insecure` → bypasses TLS certificate validation if Fleet Server uses a self-signed cert.

## 6. Accessing Logs in Kibana

Once agents are connected:

1. Menu → Discover
2. You'll see incoming logs from all enrolled agents.

