TECH2400

Cyber Security

Workshop 9

Incident Response and Management (Part 1)

KAPLAN BUSINESS SCHOOL AUSTRALIA

Subject Learning Outcomes

LO1:	Explain the terminology associated with cyber security.
LO2:	Explain the vulnerabilities and threats pertaining to the IT infrastructure of organisations.
LO3:	Analyse risk mitigation strategies that address cyber security vulnerabilities and threats.
LO4:	Describe privacy, legal, ethical and security issues and solutions related to the IT infrastructure and use of technologies in organisations.

Weekly Schedule

Week	Topic
Week 1	Introduction and Cyber Security Foundations
Week 2	Cyber Threat Landscape
Week 3	Risk Management in Cyber Security
Week 4	Cryptography Basics and Network Fundamentals Review
Week 5	Network Security Fundamentals
Week 6	Study Success Week
Week 7	Access Control and Authentication
Week 8	Ethics and Legal Aspects of Cyber Security
Week 9	Incident Response and Management (Part 1)
Week 10	Incident Response and Management (Part 2)
Week 11	Introduction to Secure Software Development
Week 12	In-Class Assessment

Weekly Schedule

Week	Topic
Week 1	Introduction and Cyber Security Foundations
Week 2	Cyber Threat Landscape
Week 3	Risk Management in Cyber Security
Week 4	Cryptography Basics and Network Fundamentals Review
Week 5	Network Security Fundamentals
Week 6	Study Success Week
Week 7	Access Control and Authentication
Week 8	Ethics and Legal Aspects of Cyber Security
Week 9	Incident Response and Management (Part 1)
Week 10	Incident Response and Management (Part 2)
Week 11	Introduction to Secure Software Development
Week 12	In-Class Assessment

What to expect from this workshop

Incident Response Lifecycle

- Key phases: Preparation, Detection & Analysis, Containment, Eradication, Recovery, and Lessons Learned

Hands-on Exercise: ELK Stack

- Using Elasticsearch, Logstash, and Kibana for incident detection & analysis

Incident Response

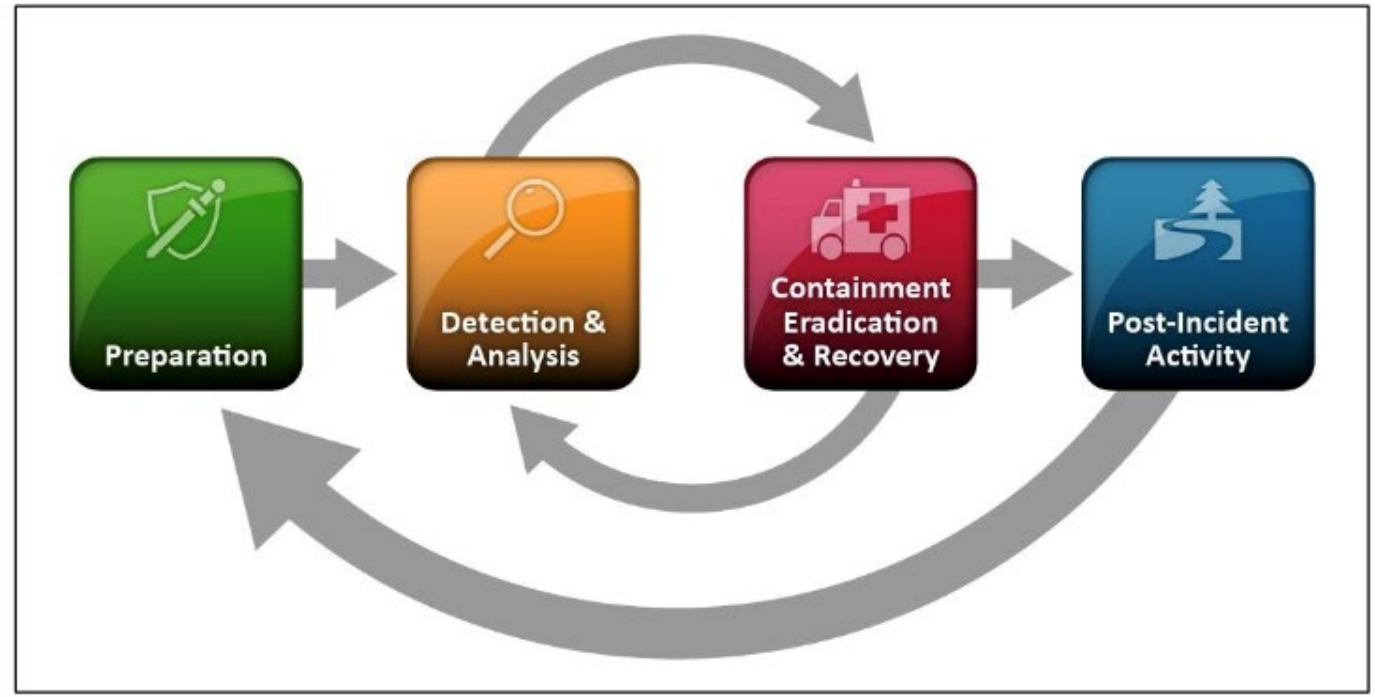
Incident response is the structured approach organisations take to handle and manage security incidents, cyber threats, or breaches to minimise damage and restore operations.

A strong incident response strategy helps:

- Reduce operational downtime
- Mitigate financial and reputational loss
- Ensure compliance with regulations

Incident Response Lifecycle

This model from the National Institute of Standards and Technology (NIST) is a widely adopted standard for incident response teams to systematically manage and mitigate cyber security incidents.



National Institute of Standards and Technology (NIST), 2012. *Computer Security Incident Handling Guide (NIST SP 800-61 Rev. 2).* Gaithersburg, MD: U.S. Department of Commerce. Available at: https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-61r2.pdf [Accessed 27 Feb. 2025].

Incident Response Lifecycle

Sometimes referred to as PICERL framework or model

NIST	PICERL
Preparation	Preparation
Detection & Analysis	Identification
Containment, Eradication, & Recovery	Containment
	Eradication
	Recovery
Post-Incident Activity	Lessons Learned

Preparation

Goal: Establish readiness to respond to incidents.

Key Activities:

- Develop & maintain an Incident Response Plan (IRP)
- Define roles & responsibilities of the Incident Response Team (IRT)
- Implement security tools (SIEM, IDS/IPS, endpoint protection)
- Conduct risk assessments & threat modeling
- Perform incident response training & simulations (tabletop exercises, red/blue team drills)

Identification

Goal: Detect and confirm an incident.

Key Activities:

- Monitor security alerts and logs (SIEM, IDS/IPS)
- Analyse anomalies & correlate threat intelligence
- Classify incidents (low, medium, high severity)
- Gather evidence for investigation
- Activate the Incident Response Team (IRT)

Containment

Goal: Limit the impact and prevent further damage.

Key Activities:

- Short-term containment (isolating affected systems, blocking malicious activity)
- Long-term containment (applying patches, implementing network segmentation)
- Preserve forensic evidence for root cause analysis
- Coordinate with stakeholders (IT, legal, management)

Eradication

Goal: Remove the threat and restore integrity.

Key Activities:

- Identify & remove malware, vulnerabilities, and compromised accounts
- Patch affected systems and update security configurations
- Validate that no backdoors or persistent threats remain
- Conduct forensic analysis to understand attack vectors

Recovery

Goal: Restore normal operations securely.

Key Activities:

- Validate clean systems before reconnecting
- Monitor for any signs of reinfection or ongoing threats
- Restore data from backups (if necessary)
- Conduct post-recovery testing to ensure stability

Lessons Learned

Goal: Improve future response and strengthen security.

Key Activities:

- Conduct post-incident review (PIR)
- Document findings, root causes, and response effectiveness
- Update incident response plans & playbooks
- Implement additional security controls to prevent recurrence
- Provide training based on insights from the incident

Role of Software

Software systems enhance efficiency, accuracy, and speed in incident response.

Different tools support various phases of the incident response lifecycle (detection, analysis, containment, recovery, and reporting).

A well-integrated set of tools ensures proactive threat management and rapid response.

Examples of Software

Category	Examples	Purpose
Security Information & Event Management (SIEM)	**ELK Stack**, Splunk, Microsoft Sentinel	Centralised log collection, correlation, and alerting
Endpoint Detection & Response (EDR)	CrowdStrike, SentinelOne, Microsoft Defender	Detects and responds to endpoint threats
Threat Intelligence Platforms	MISP, Recorded Future, VirusTotal	Provides intelligence on emerging threats
Security Orchestration, Automation, & Response	Cortex XSOAR, IBM Resilient, Splunk SOAR	Automates incident response workflows
Forensic & Investigation Tools	Autopsy, Velociraptor, Wireshark	Helps analyse and reconstruct cyber incidents

Elasticsearch, Logstash, and Kibana (ELK)

ELK is an open-source log management and analytics solution used for security monitoring.

Component	Function	Benefit
Elasticsearch	Stores and indexes large volumes of log data for fast searching.	Enables quick log analysis and threat hunting.
Logstash	Collects, processes, and transforms log data from multiple sources.	Aggregates logs from firewalls, endpoints, and network devices.
Kibana	Provides dashboards and visualisations for data analysis.	Helps monitor threats, detect anomalies, and conduct forensic investigations.

Elasticsearch, Logstash, and Kibana (ELK)

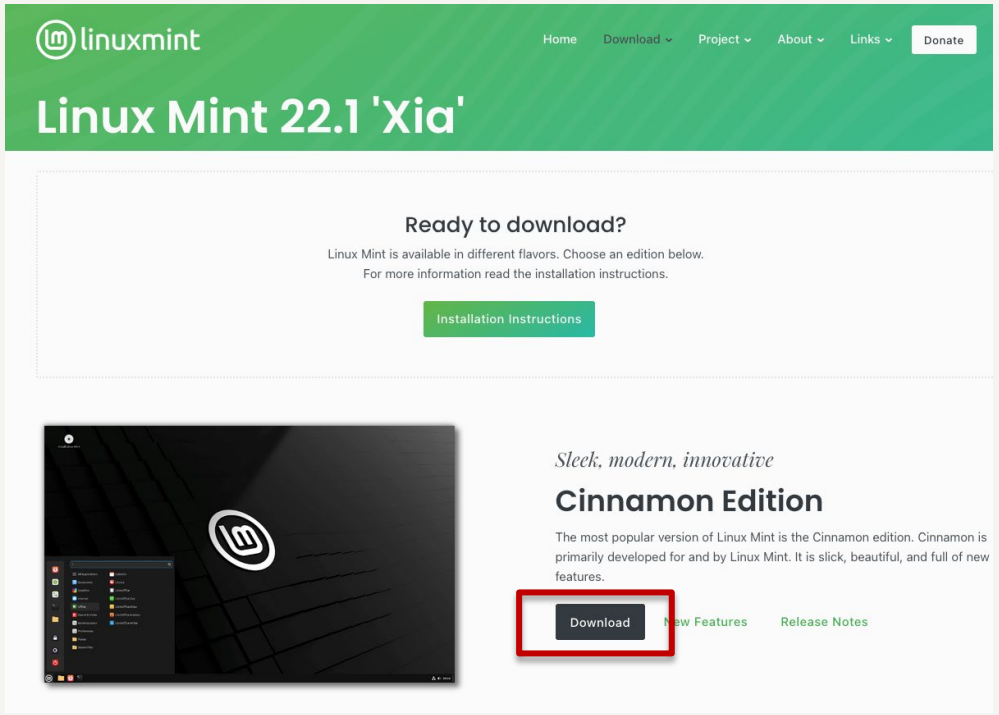
- Core component of many SIEM solutions due to its powerful log management and visualisation capabilities.
- When configured for security monitoring, ELK supports real-time threat detection, incident response, and compliance reporting.
- When integrated with Security Orchestration, Automation, and Response (SOAR) tools, ELK enables automated threat mitigation.
- ELK can function as a lightweight, open-source alternative to commercial SIEMs or as a supplementary tool for deeper log analysis.

Activity: Linux Mint Installation

Linux Mint provides a straightforward, native environment for ELK, ensuring better performance and easier setup.

Step 1: Download the Linux Mint ISO

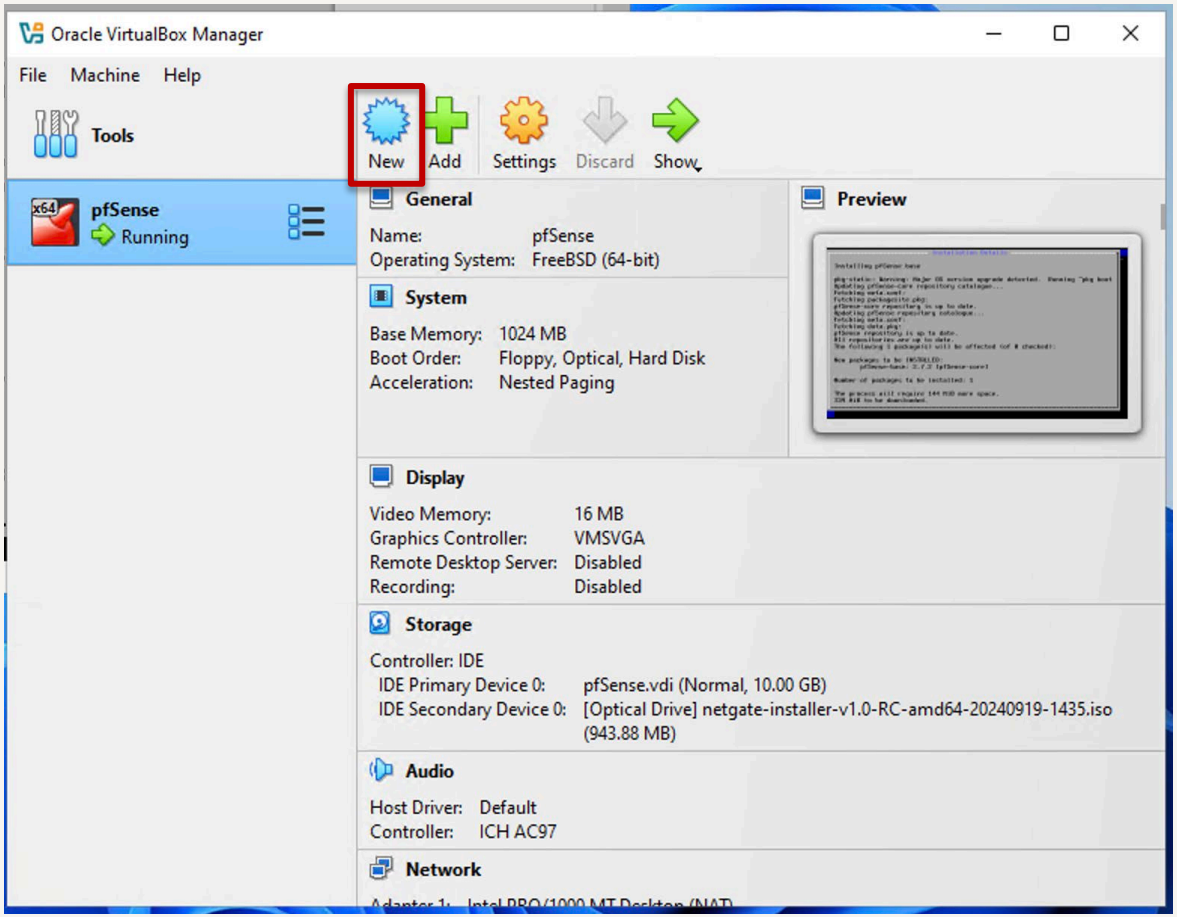
- Go to https://www.linuxmint.com/download.php
- Choose the Cinnamon edition
- Select a mirror near your location
- Download the ISO file



Activity: Linux Mint Installation

Step 2: Create a new virtual machine

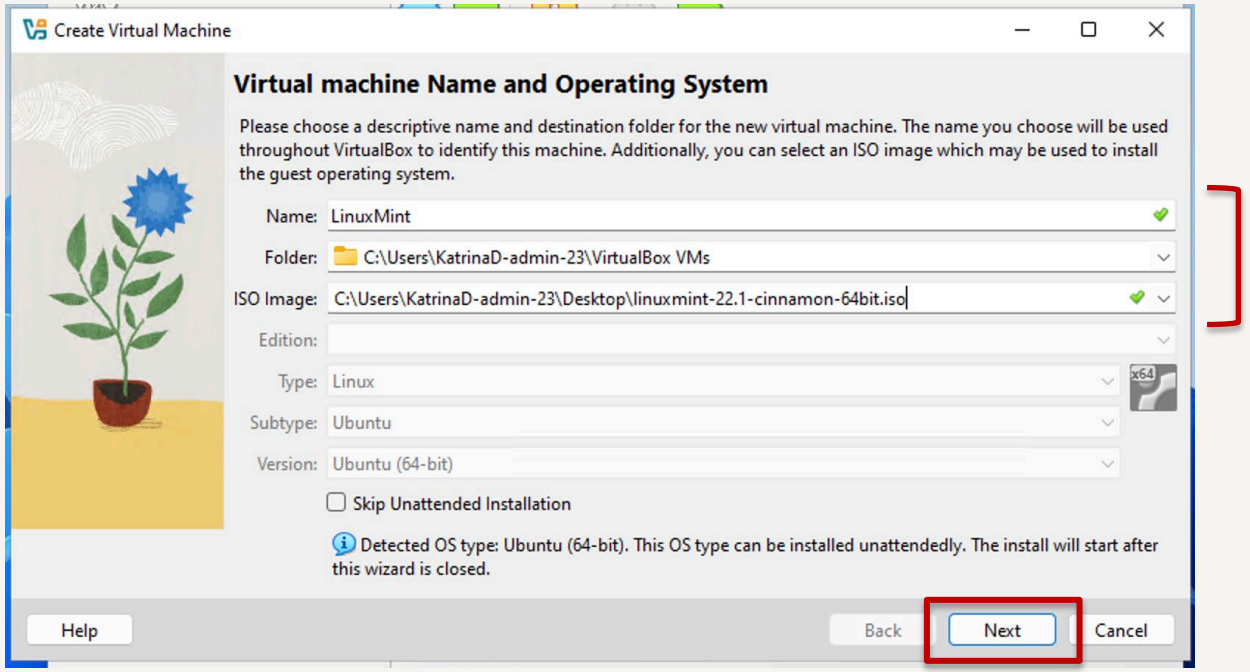
- Launch VirtualBox
- Click on the New button to create a new VM



Activity: Linux Mint Installation

Step 2: Create a new virtual machine

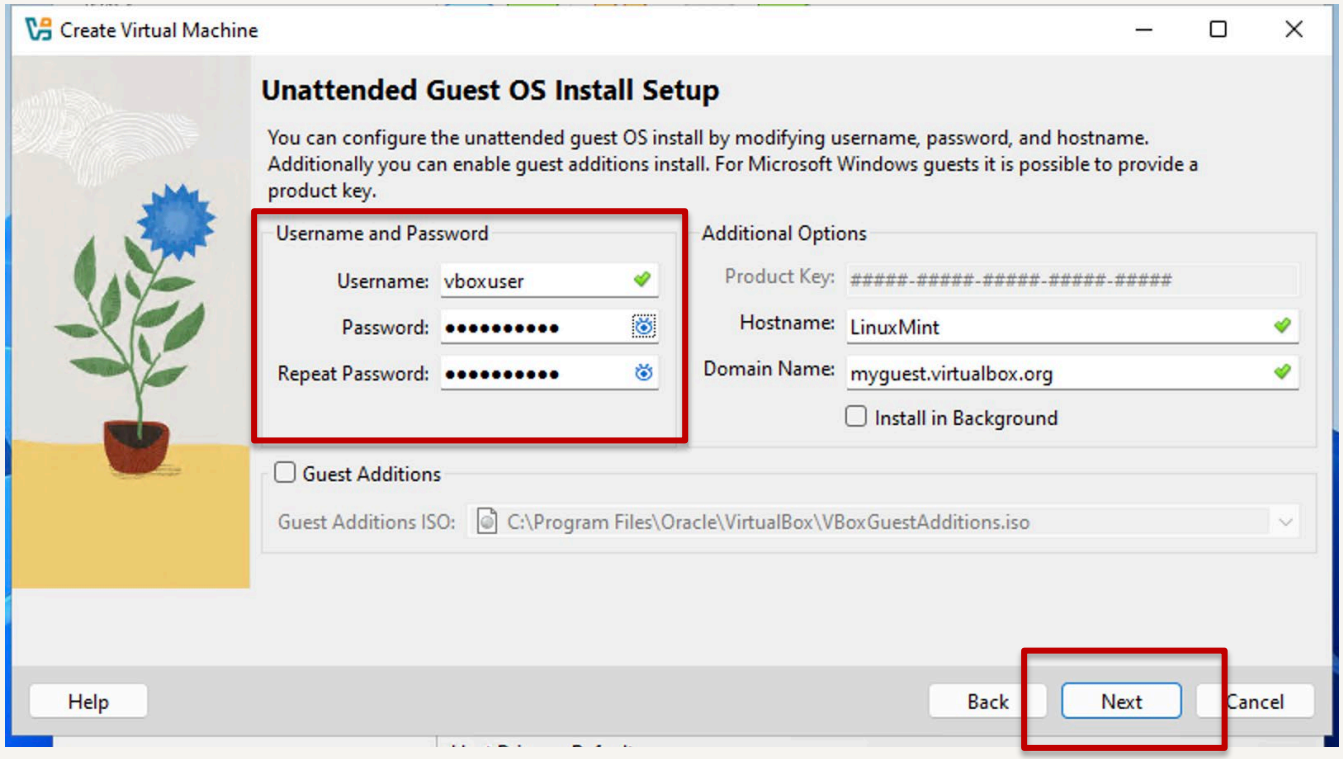
- Configure the VM
  - Name: LinuxMint
  - ISO Image: Select downloaded iso file
- Click Next



Activity: Linux Mint Installation

Step 2: Create a new virtual machine

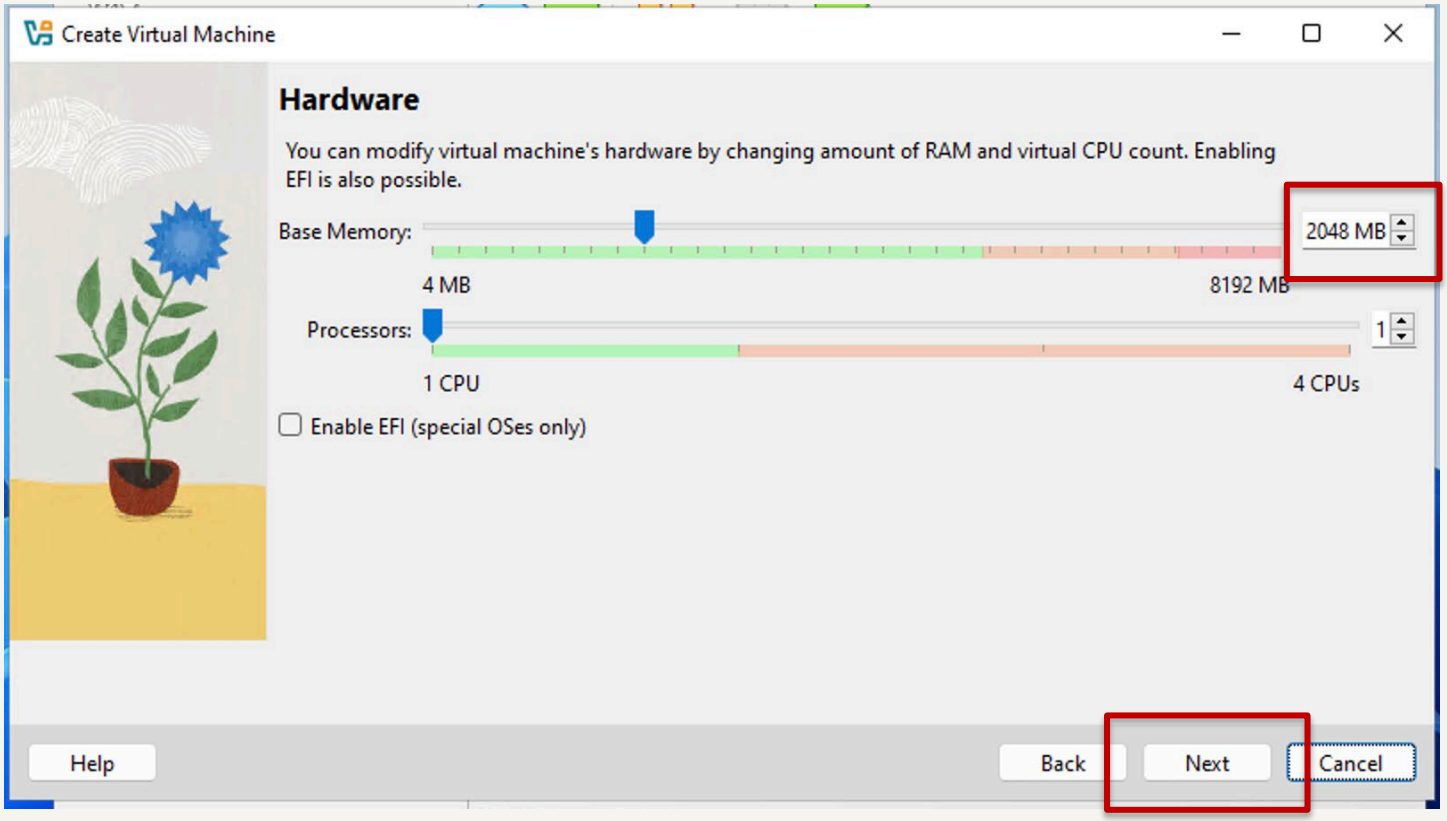
- Change default password
- Click Next



Activity: Linux Mint Installation

Step 2: Create a new virtual machine

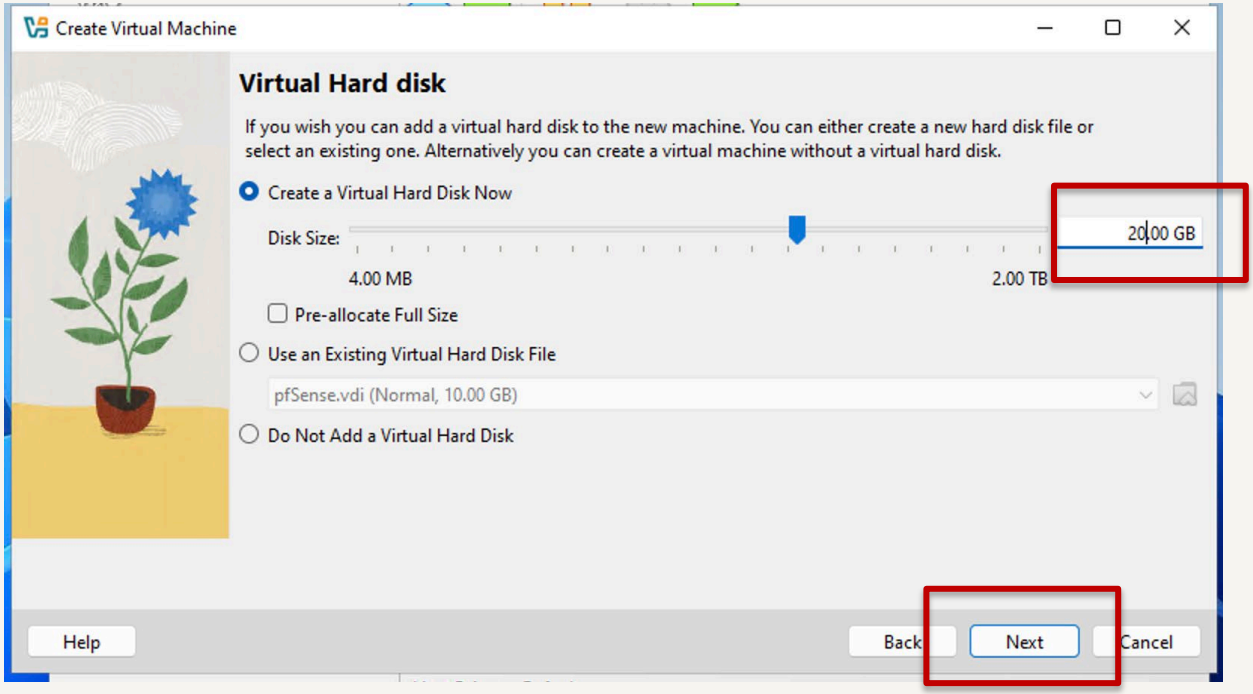
- Allocate at least 2048 MB (2GB) of RAM



Activity: Linux Mint Installation

Step 2: Create a new virtual machine

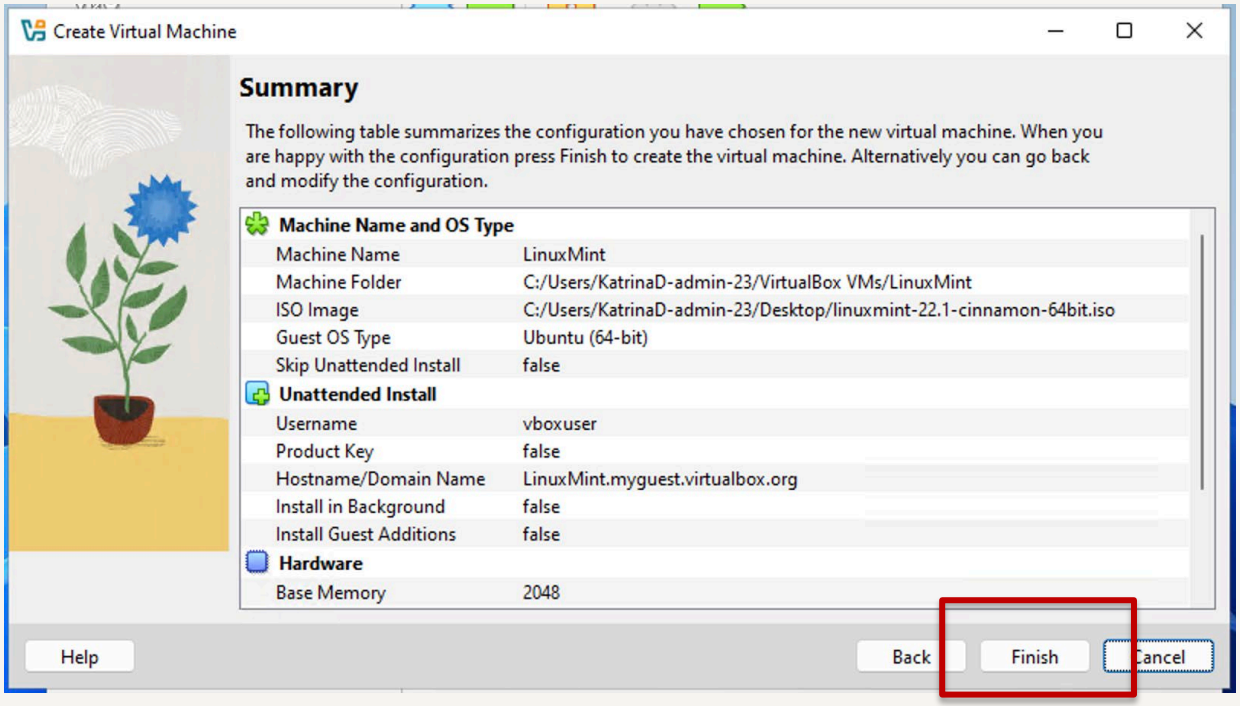
- Create a Virtual Hard Disk:
  - Choose Create a Virtual Hard Disk Now
  - Set the size of the virtual hard disk to 20GB or more
  - Click Next



Activity: Linux Mint Installation

Step 2: Create a new virtual machine

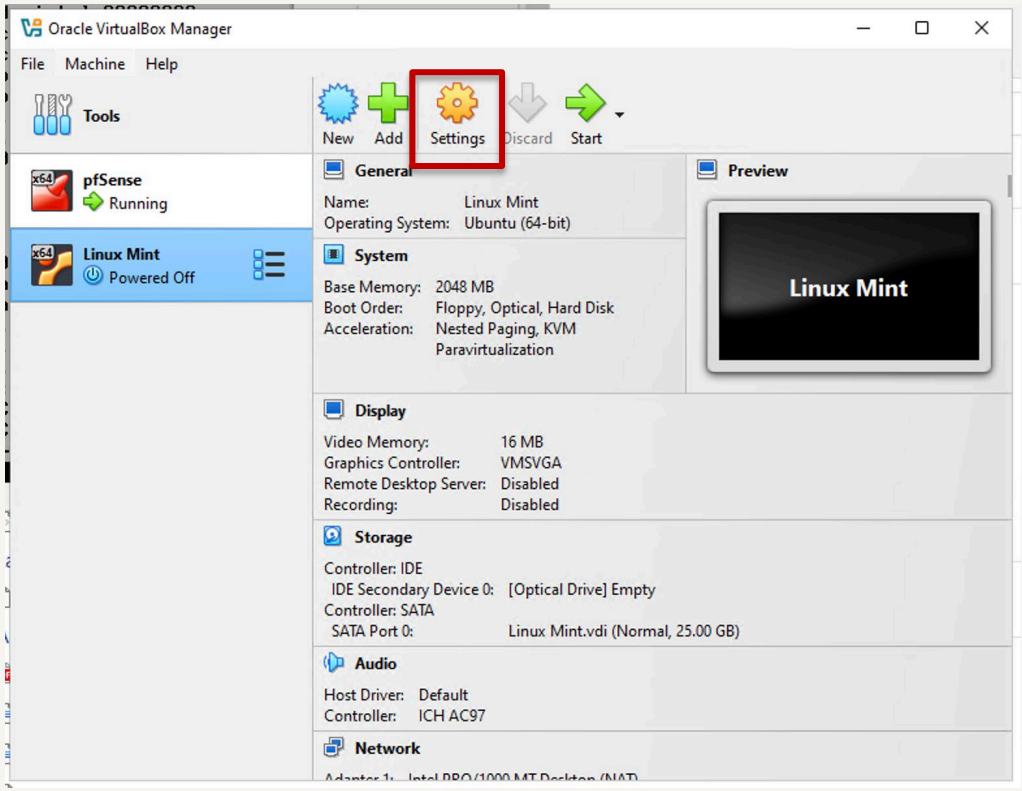
- Review the Summary
- Click Finish



Activity: Linux Mint Installation

Step 3: Mount the Linux Mint ISO to the Virtual Machine

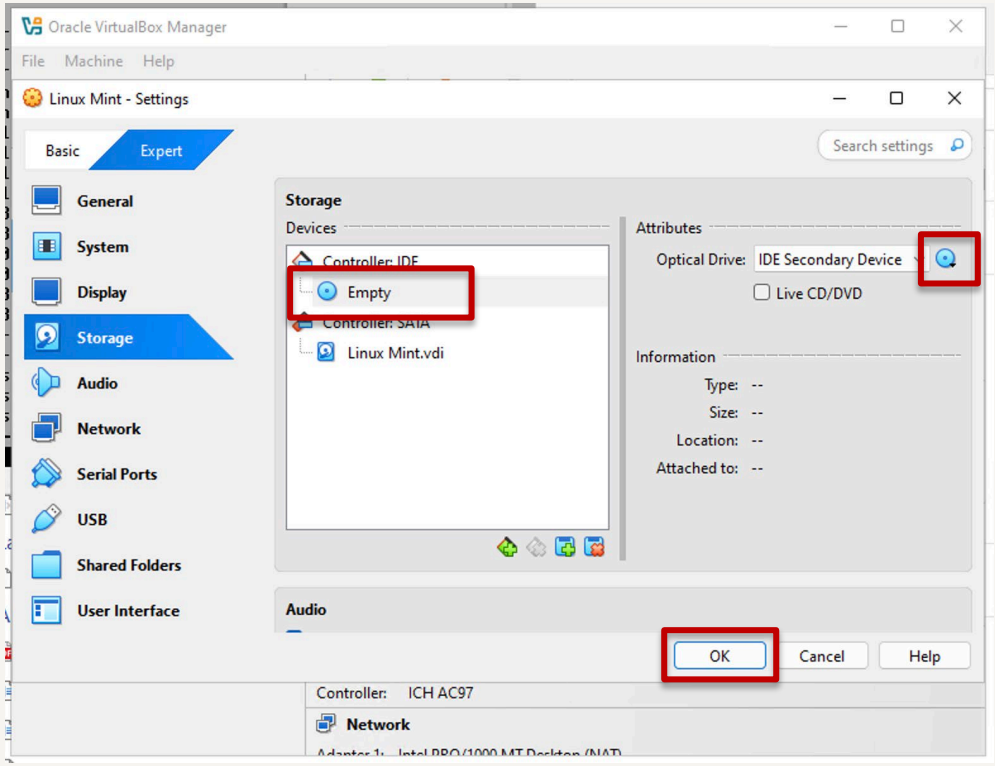
- Select the Linux Mint VM in VirtualBox and click on Settings



Activity: Linux Mint Installation

Step 3: Mount the Linux Mint ISO to the Virtual Machine

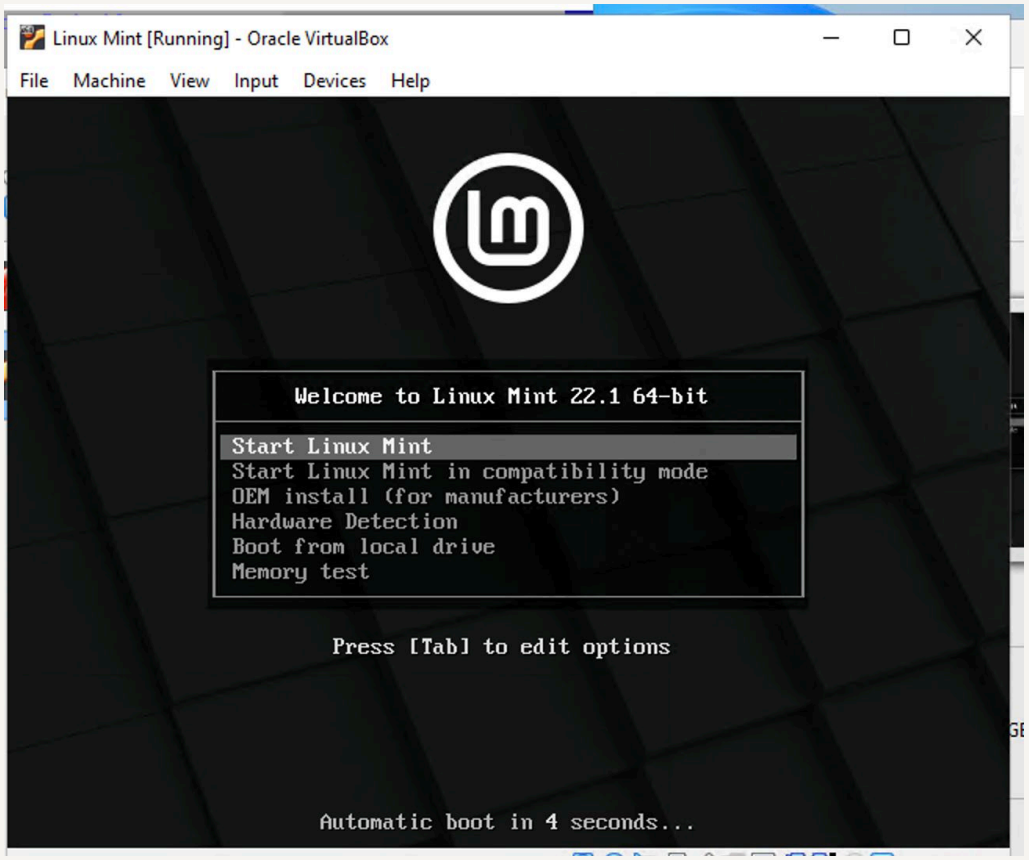
- Go to Storage:
  - Under the "Controller: IDE section," click on the empty disk icon
  - On the right side, click the disk icon next to "Optical Drive" and select Choose a disk file
  - Browse to the location where you saved the Linux Mint ISO file and select it
- Click Open
- Click OK



Activity: Linux Mint Installation

Step 4: Start the Virtual Machine

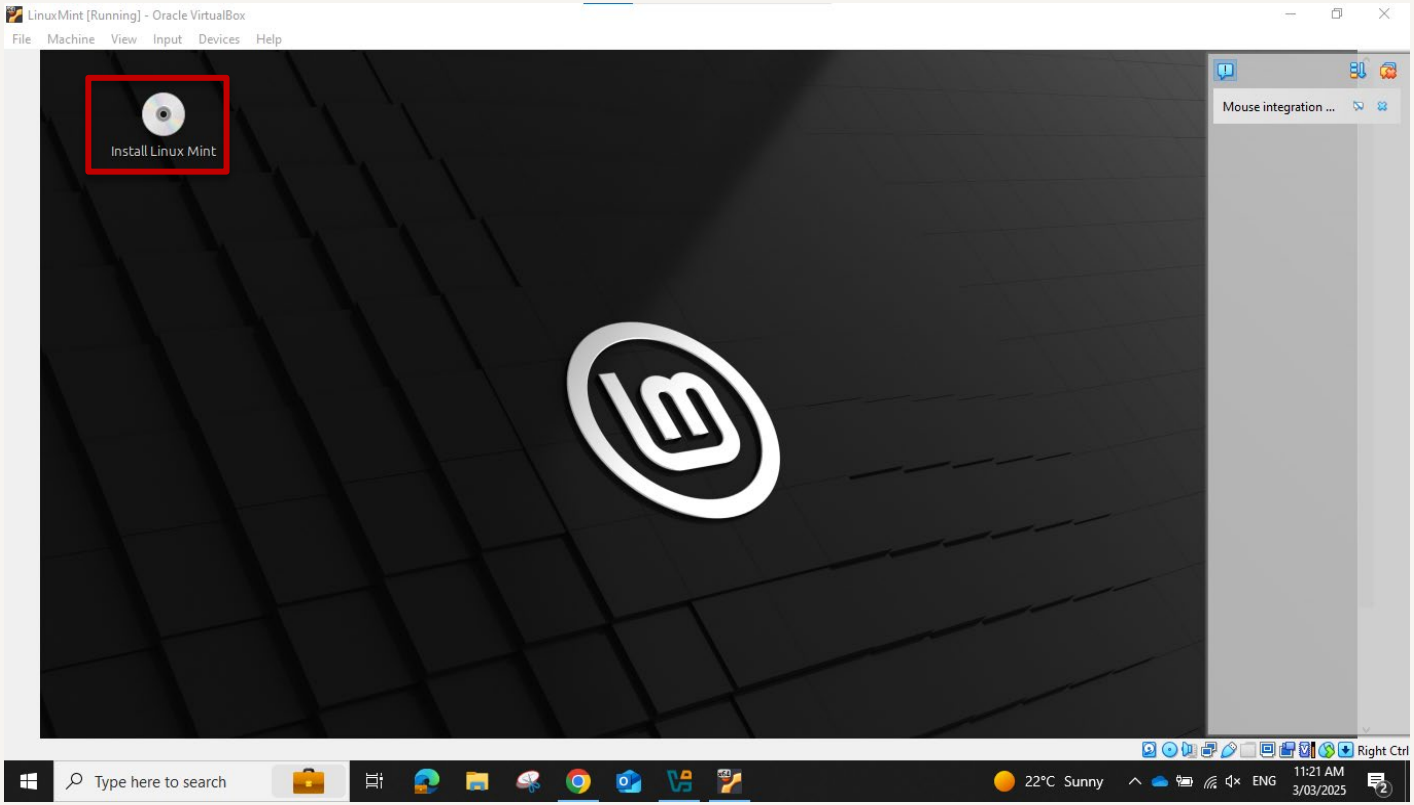
- Click Start in VirtualBox to boot from the Linux Mint ISO
- Wait for the Linux Mint welcome screen to appear
- Select Start Linux Mint and press Enter



Activity: Linux Mint Installation

Step 5: Install Linux Mint

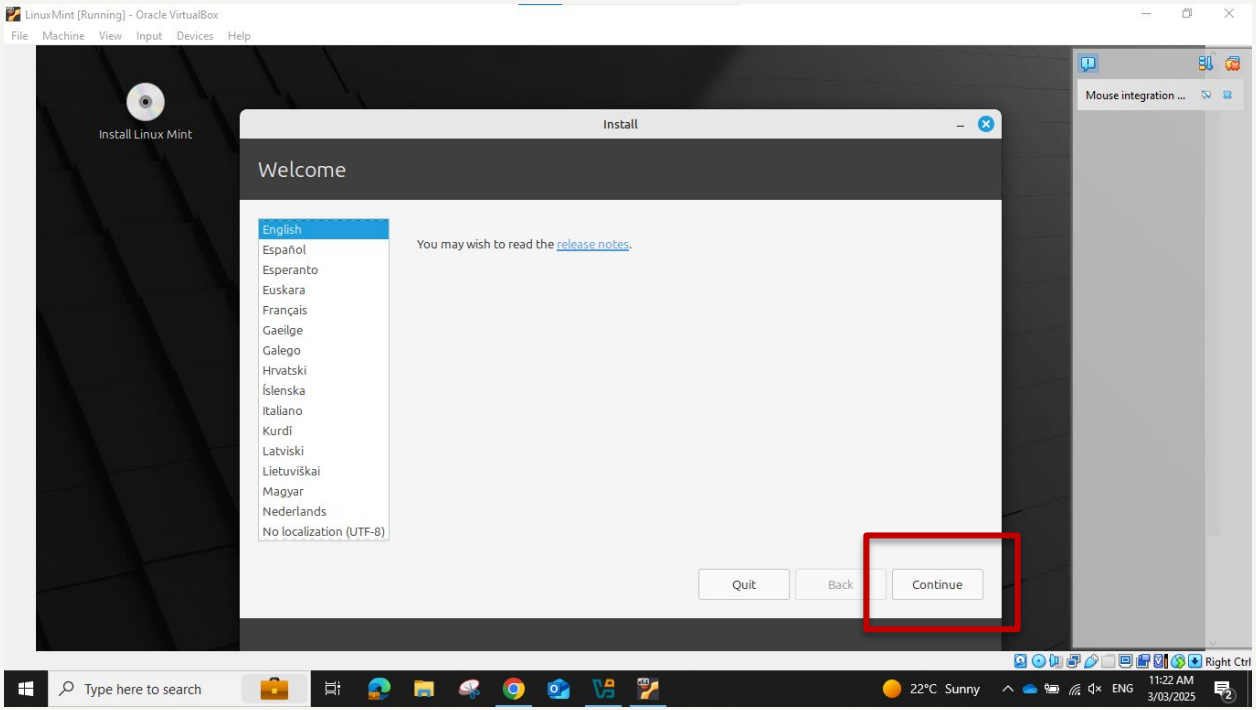
- Double-click on Install Linux Mint to launch



Activity: Linux Mint Installation

Step 5: Install Linux Mint

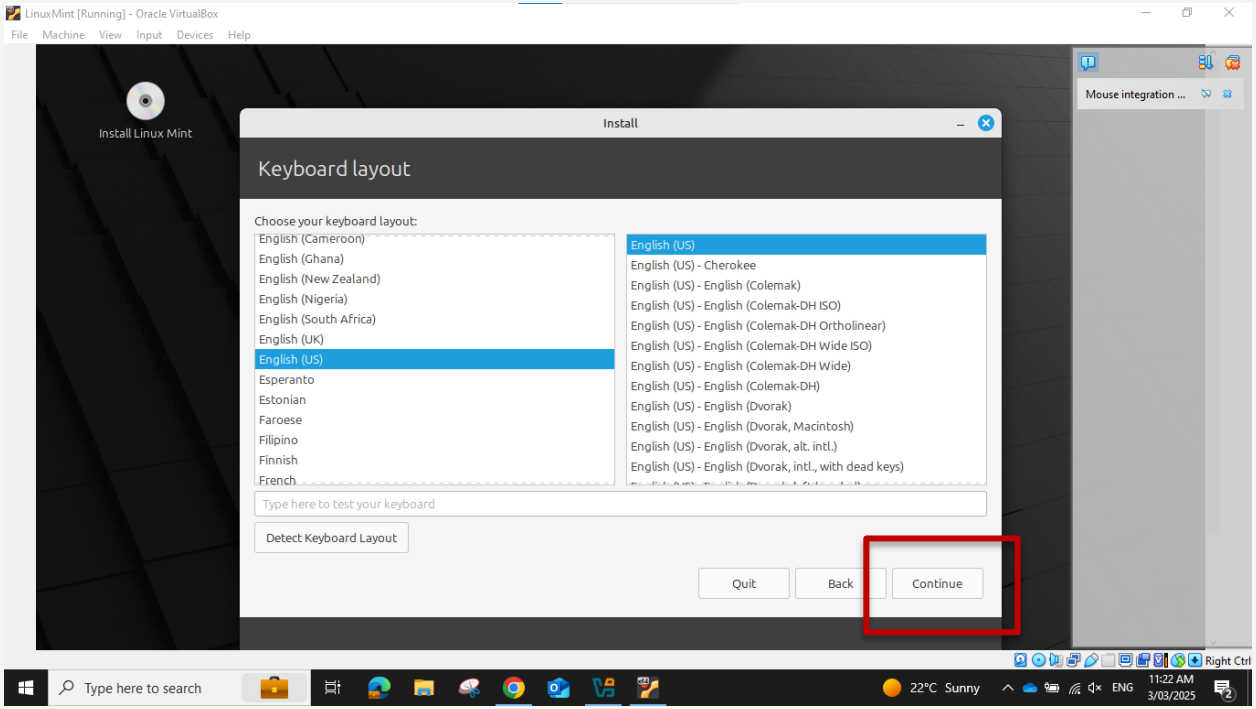
- Select English and click Continue



Activity: Linux Mint Installation

Step 5: Install Linux Mint

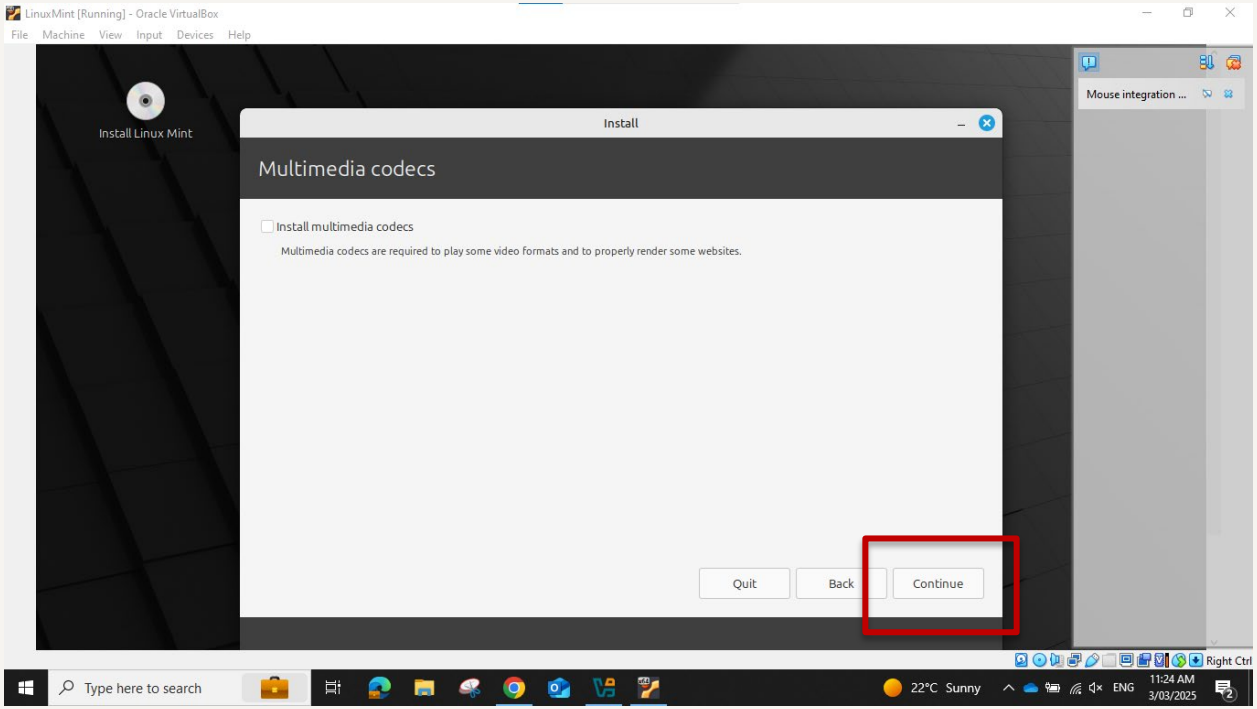
- Choose your preferred keyboard layout and click Continue



Activity: Linux Mint Installation

Step 5: Install Linux Mint

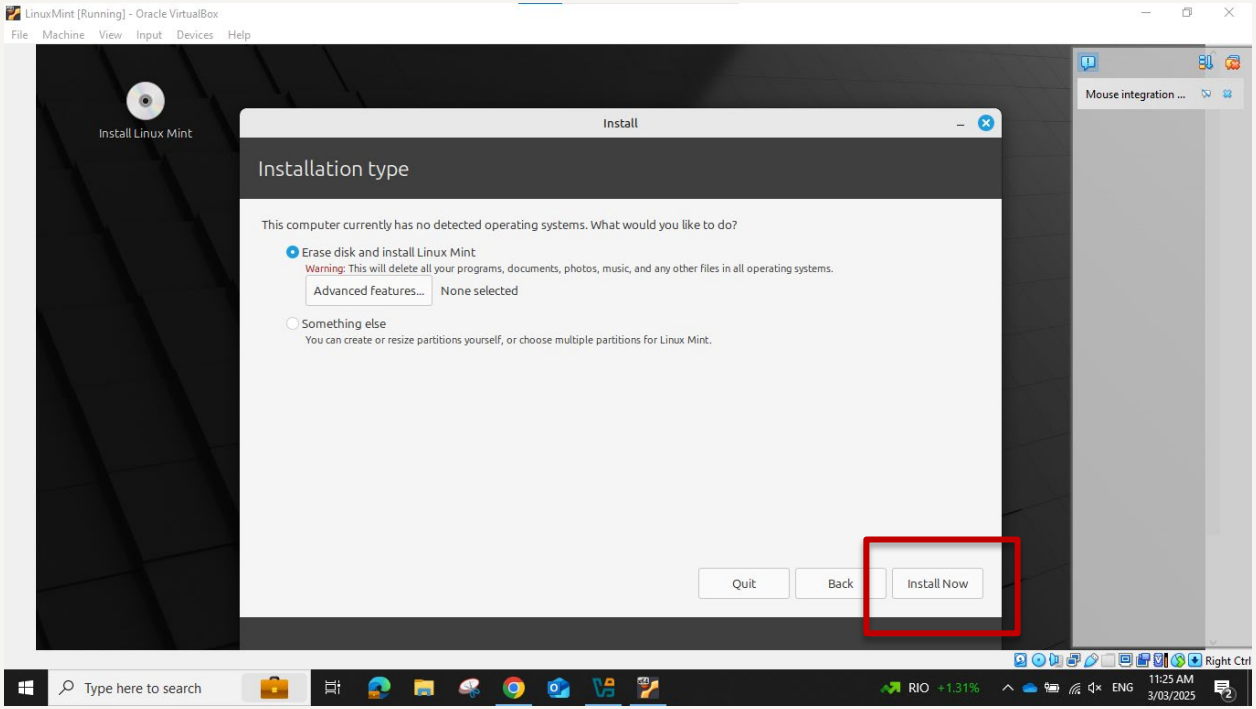
- No need to install Multimedia codecs. Click Continue



Activity: Linux Mint Installation

Step 5: Install Linux Mint

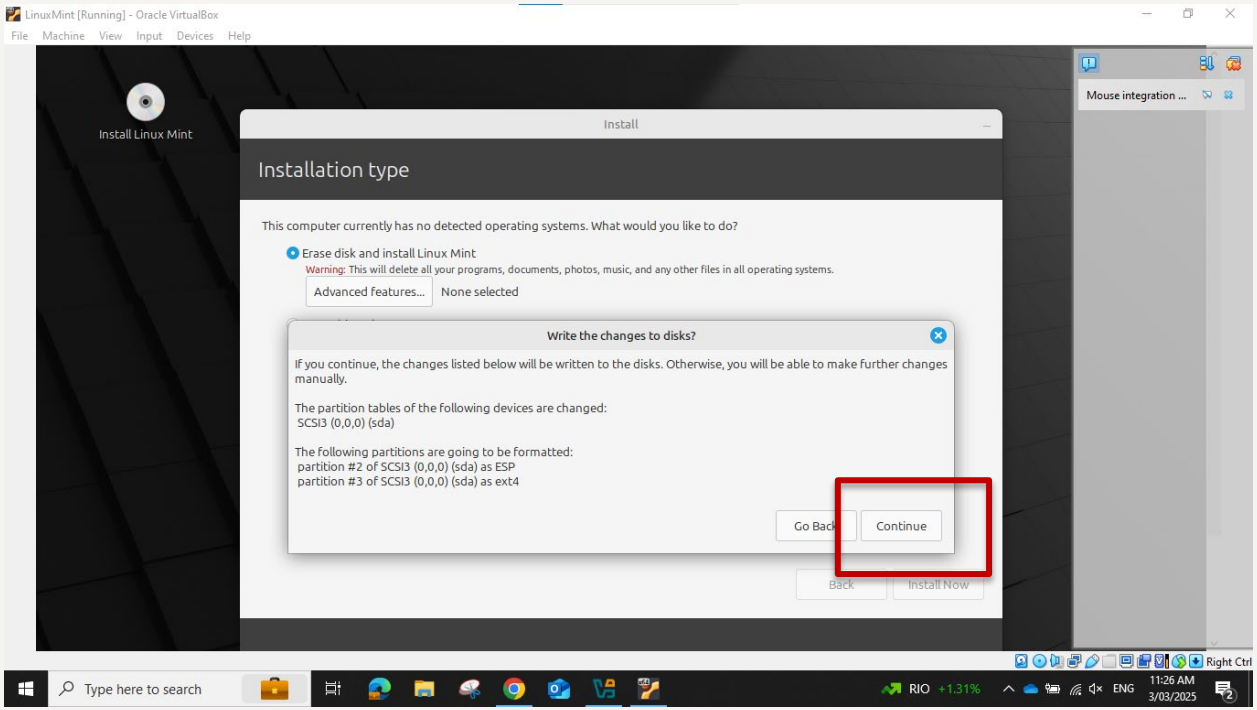
- Select Erase disk and install Linux Mint. Click Install now.



Activity: Linux Mint Installation

Step 5: Install Linux Mint

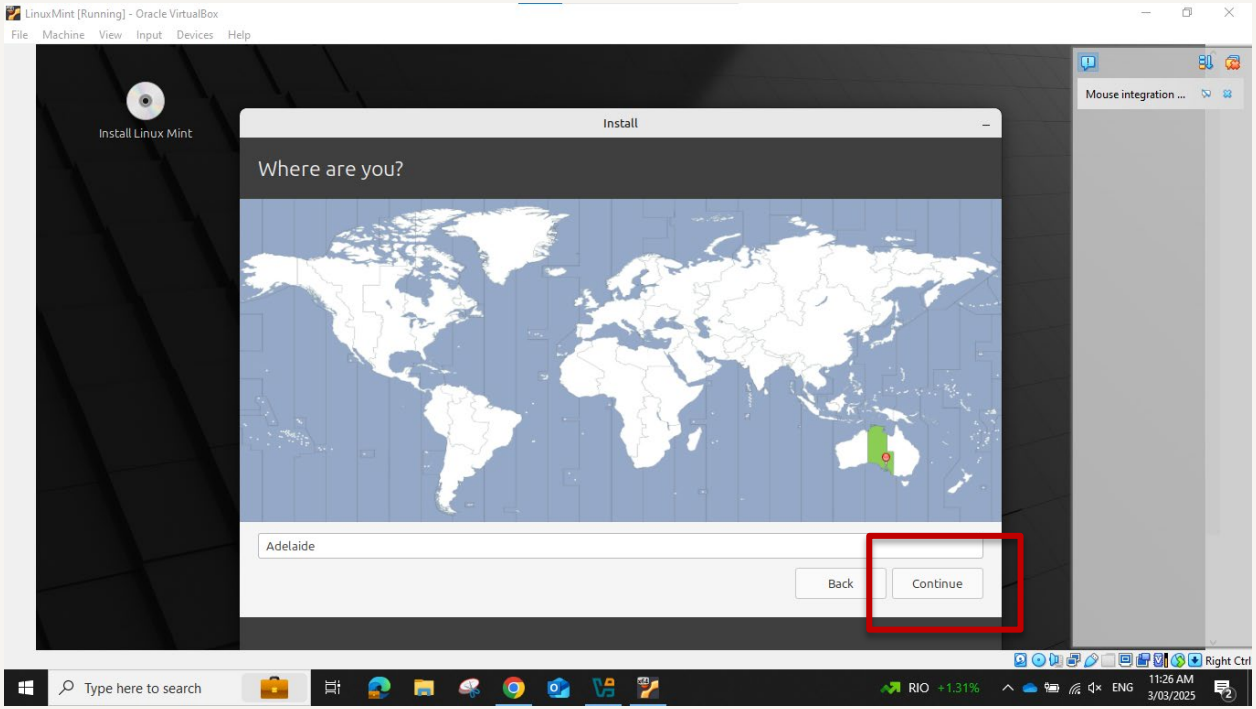
- You will get a warning message for the changes to the disk. Confirm by clicking on Continue



Activity: Linux Mint Installation

Step 5: Install Linux Mint

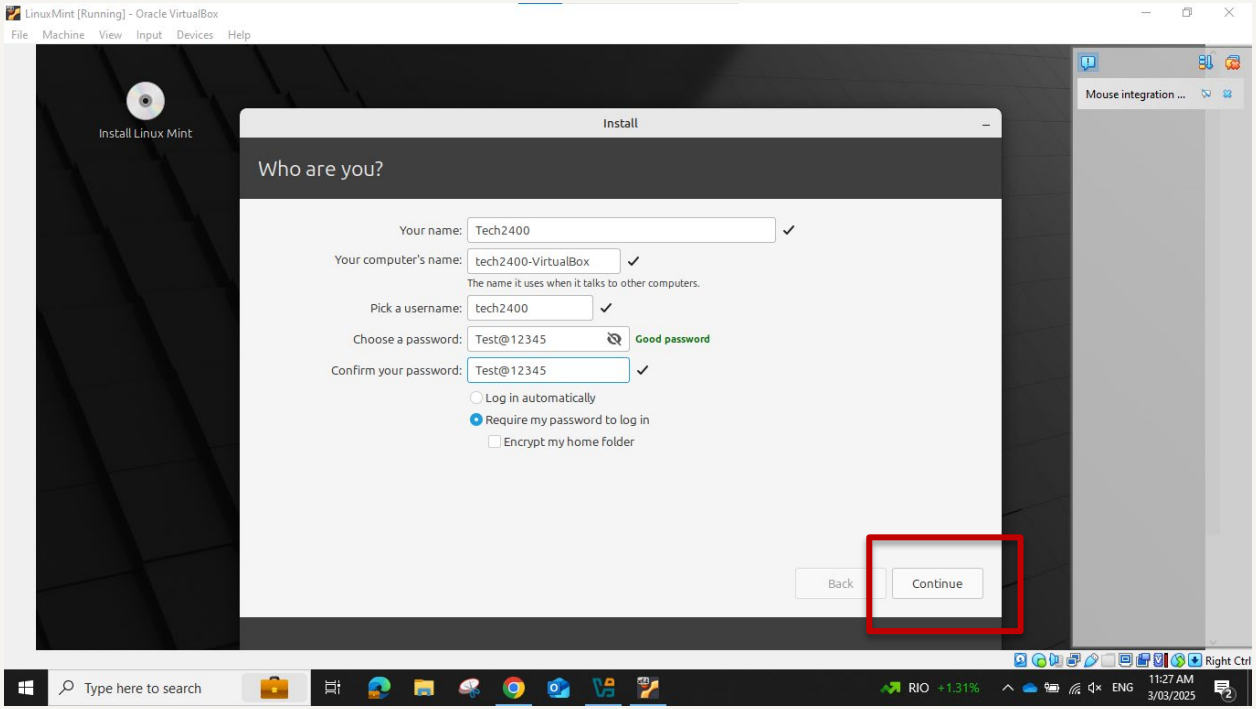
- Select time zone and click Continue



Activity: Linux Mint Installation

Step 5: Install Linux Mint

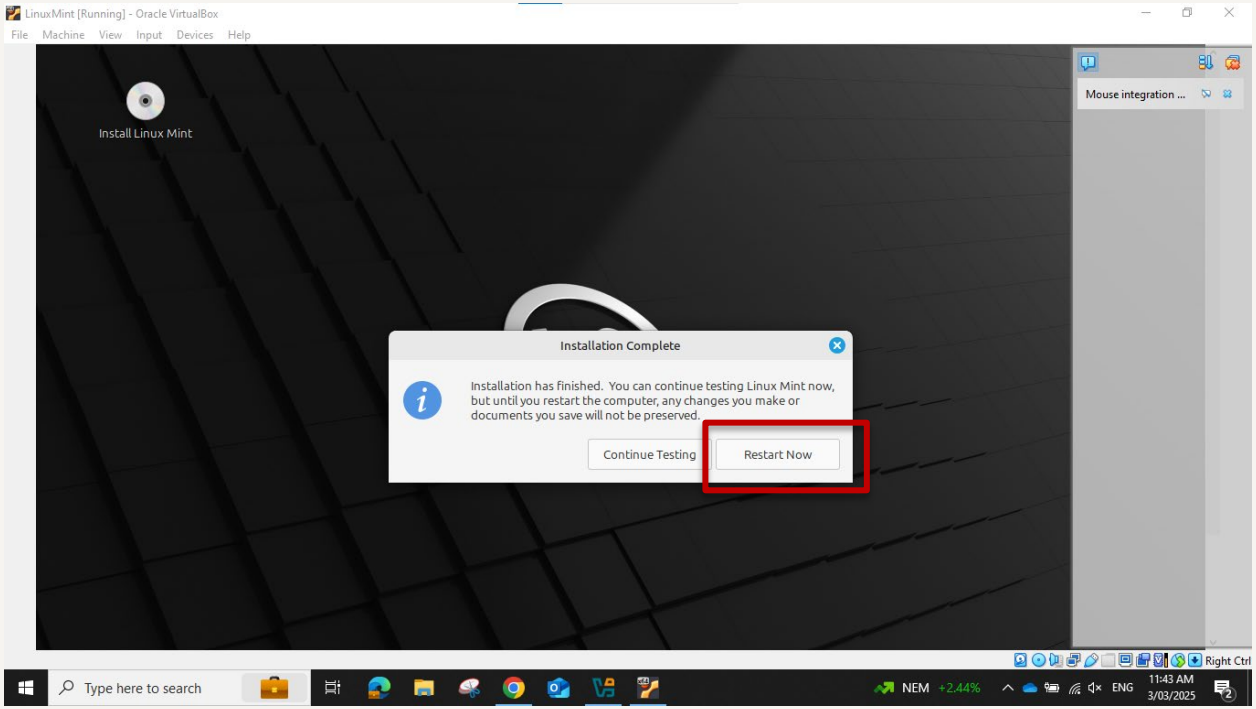
- Enter your name, computer name, username, and password.
- Click Continue.



Activity: Linux Mint Installation

Step 5: Install Linux Mint

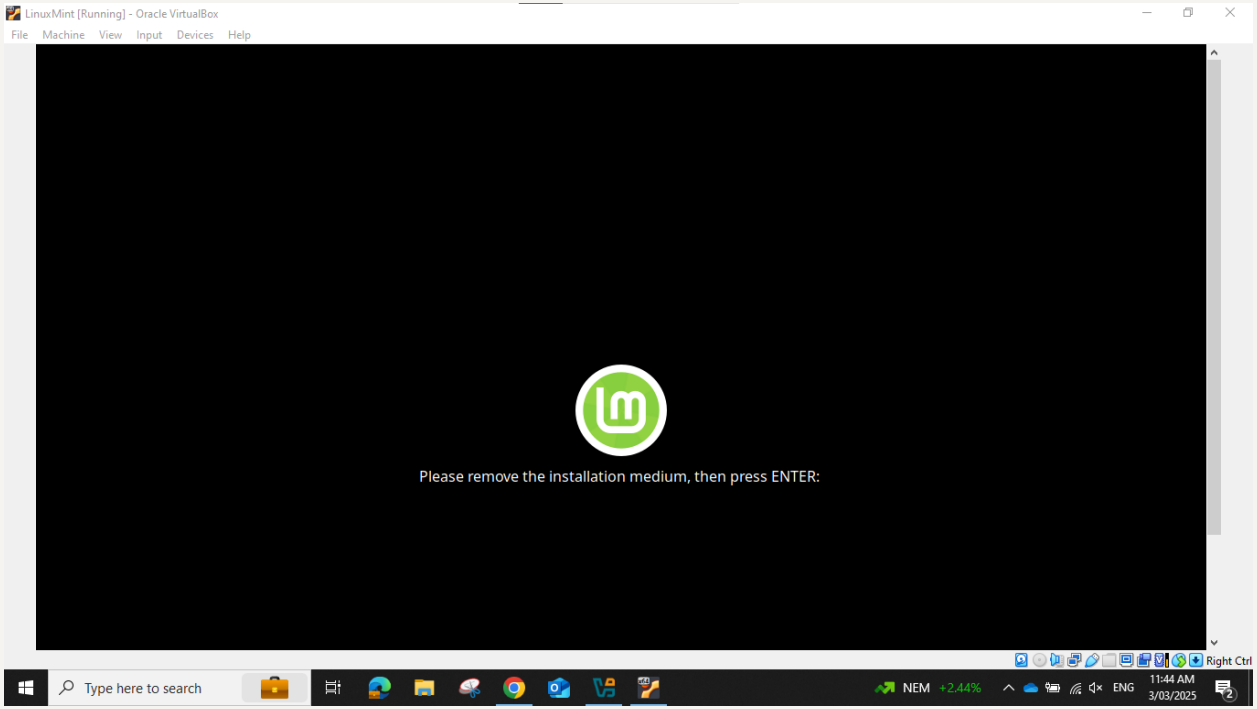
- The installation process will take a few minutes to complete. Once done, you'll be asked to reboot the system.



Activity: Linux Mint Installation

Step 6: Reboot and Log In

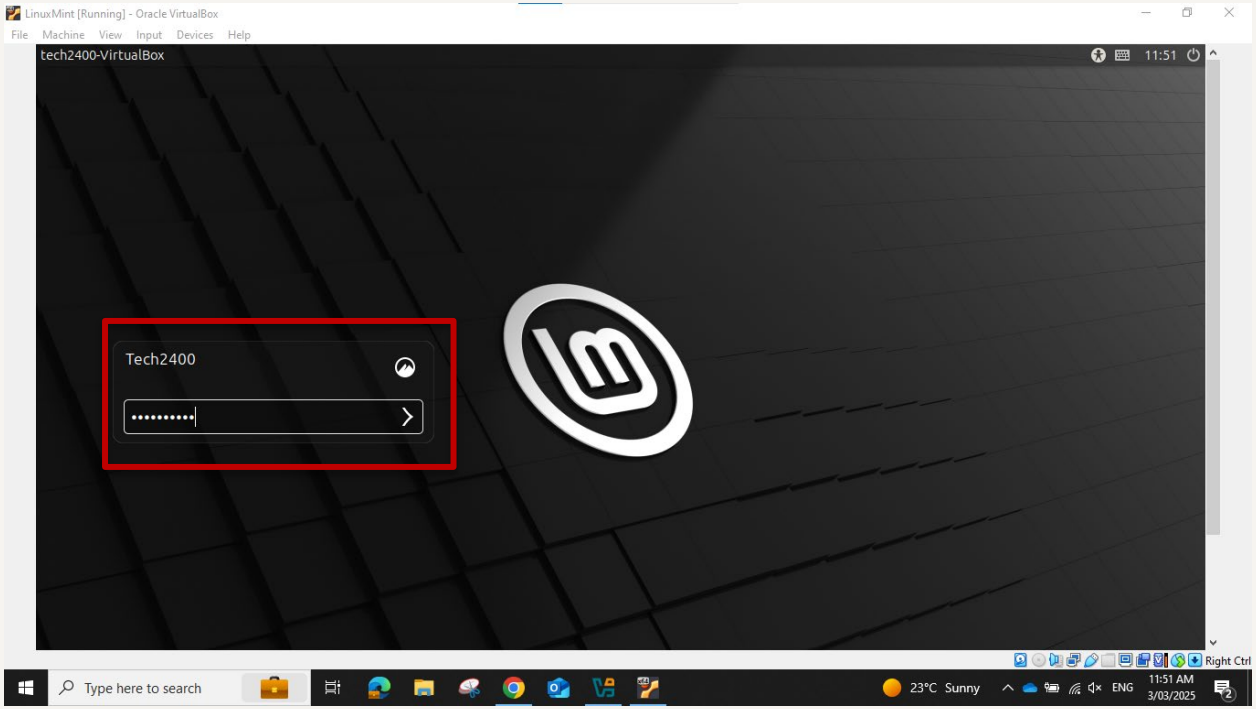
- Press Enter key when prompted to remove the installation media.



Activity: Linux Mint Installation

Step 6: Reboot and Log In

- After restarting, you should see the login screen.
- Enter your username and password to log in to Linux Mint



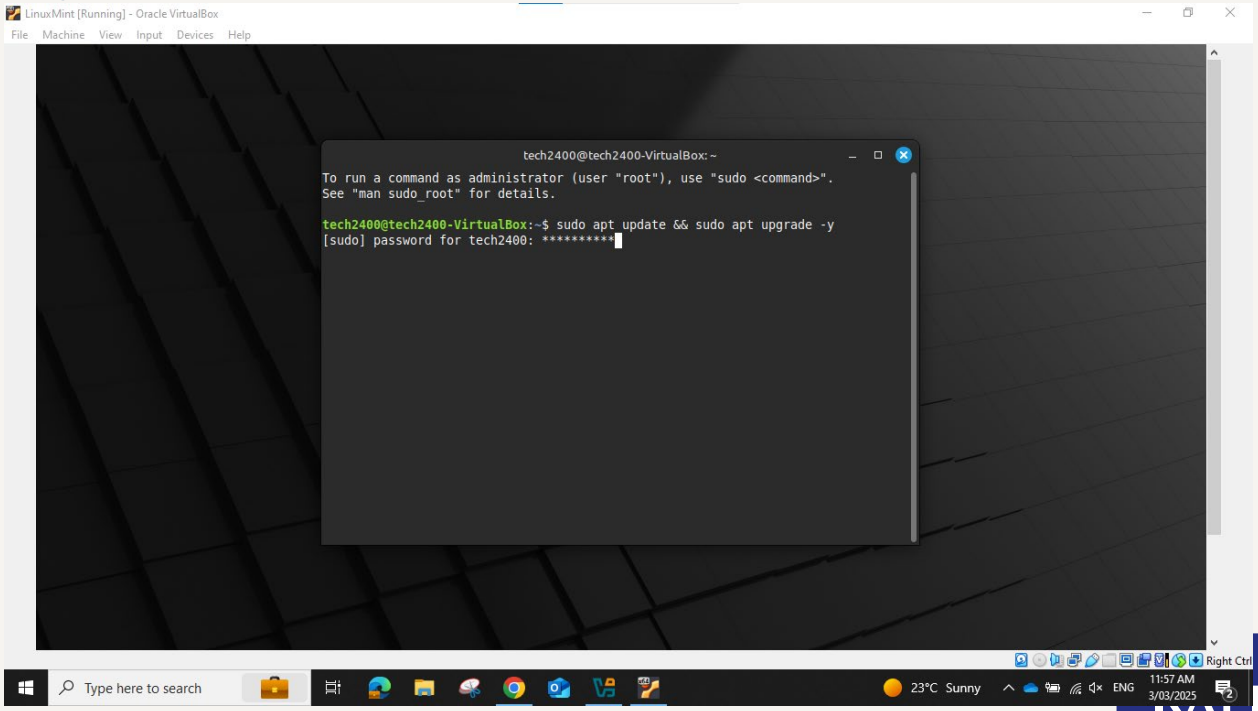
Activity: Linux Mint Installation

Step 7: Update the system

- After logging in, open a terminal and run the following command to update the system:

```
sudo apt update && sudo apt upgrade –y
```

- Enter your password when prompted

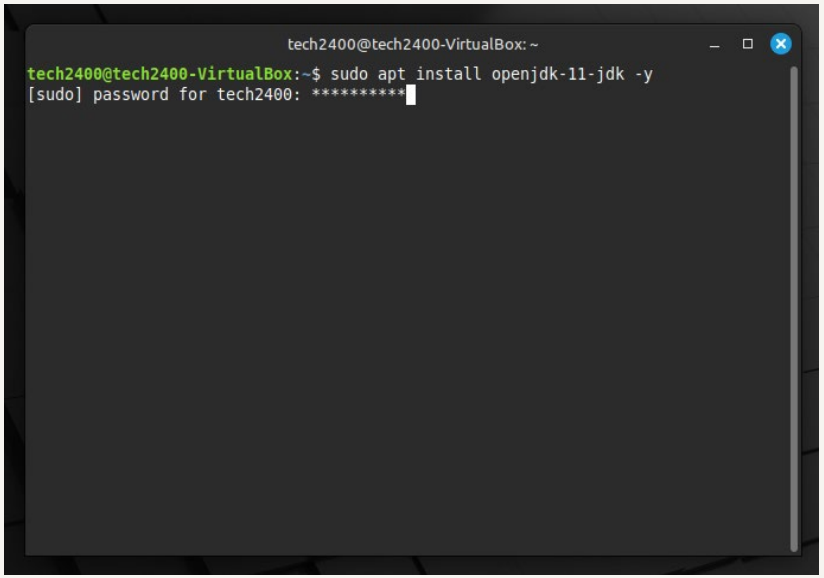


Activity: Install Java

Elasticsearch and Logstash both require Java.
OpenJDK 11 is an open-source implementation of Java SE.

Step 1: Install OpenJDK 11

- From your Terminal window
- Run sudo apt install openjdk-11-jdk -y
- Enter your password when prompted

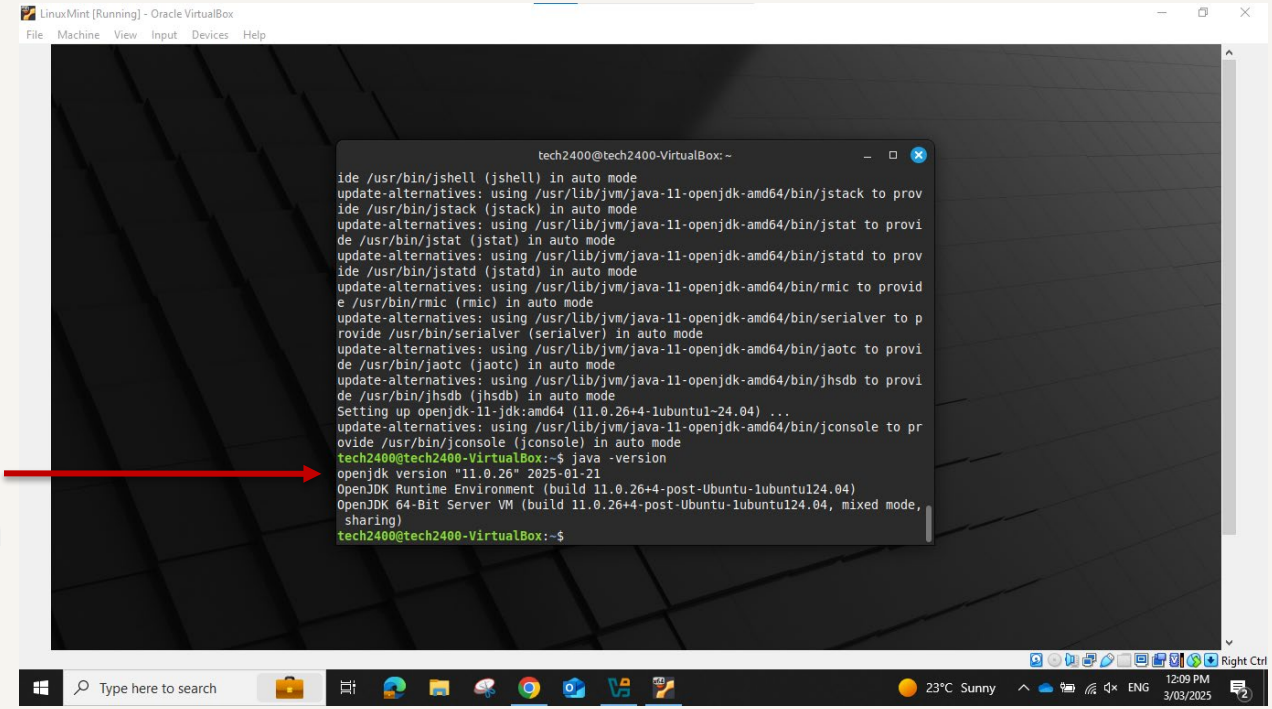


Activity: Install Java

Step 2: Verify Java installation

- Run java -version
- This should return the current OpenJDK version (11.x.x)

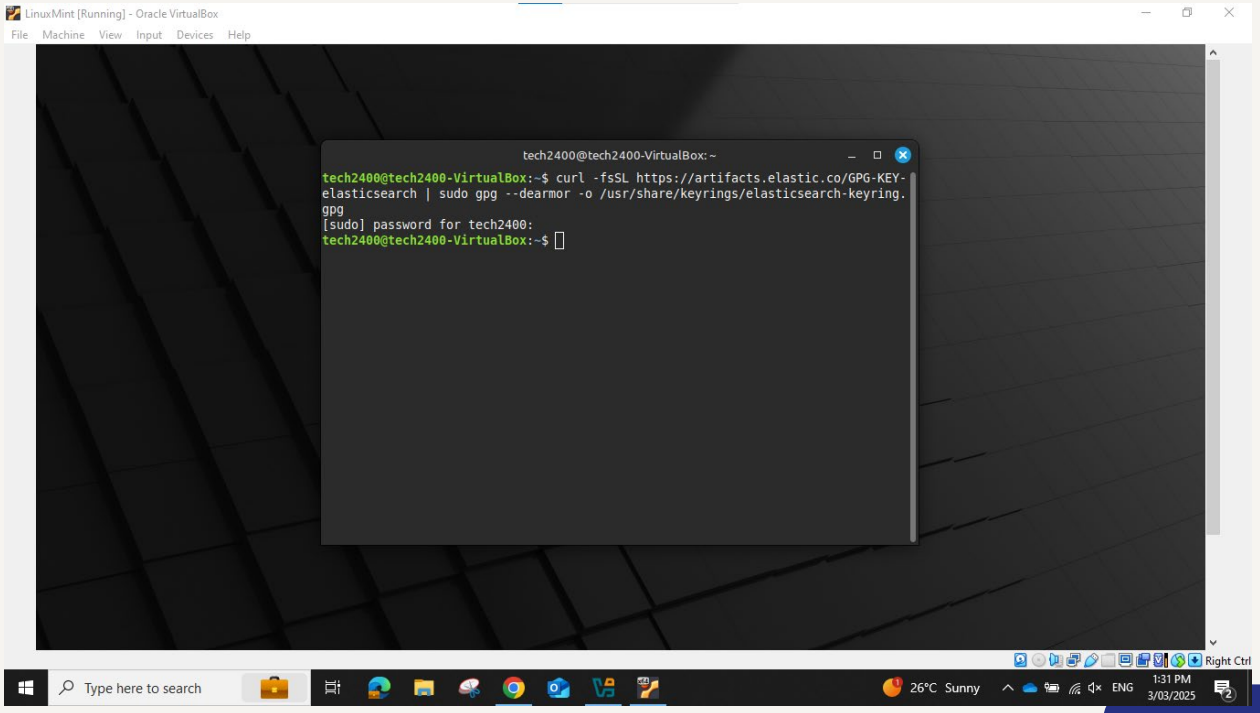
Indicates successful installation



Activity: Install Elasticsearch

Step 1: Add Elasticsearch GPG key

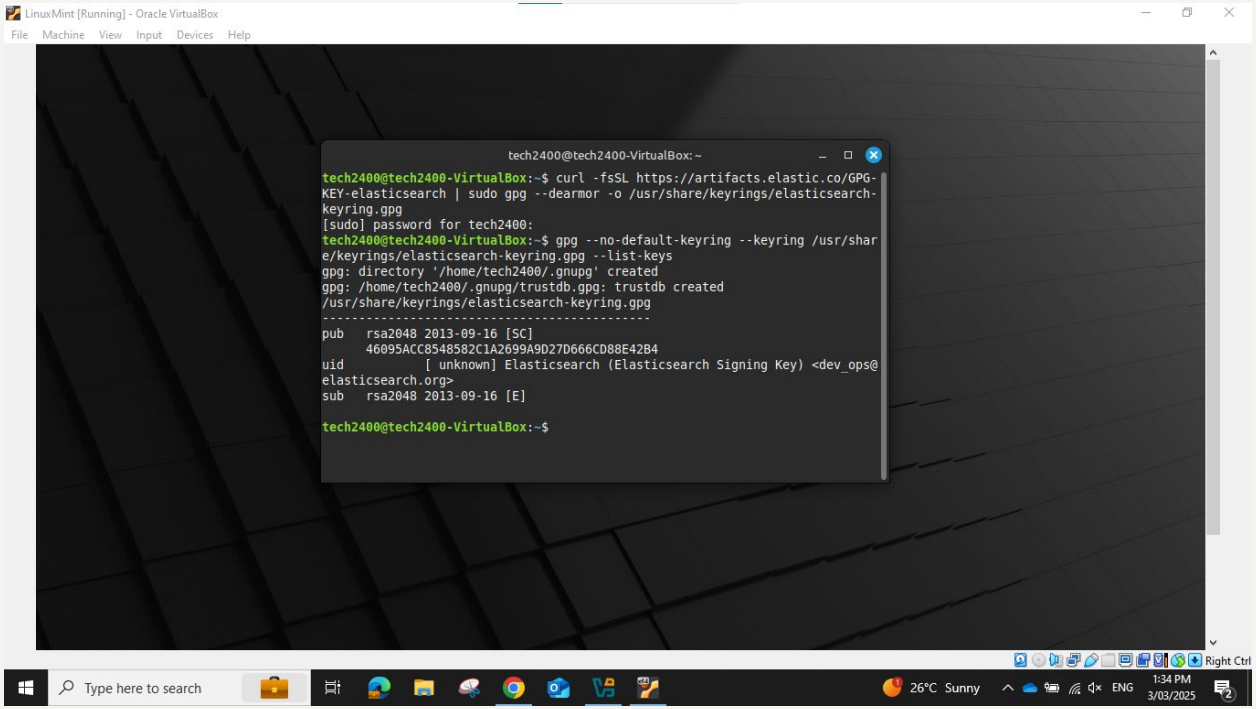
- Run curl -fsSL https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo gpg --dearmor -o /usr/share/keyrings/elasticsearch-keyring.gpg



Activity: Install Elasticsearch

Step 2: Verify that key was added successfully

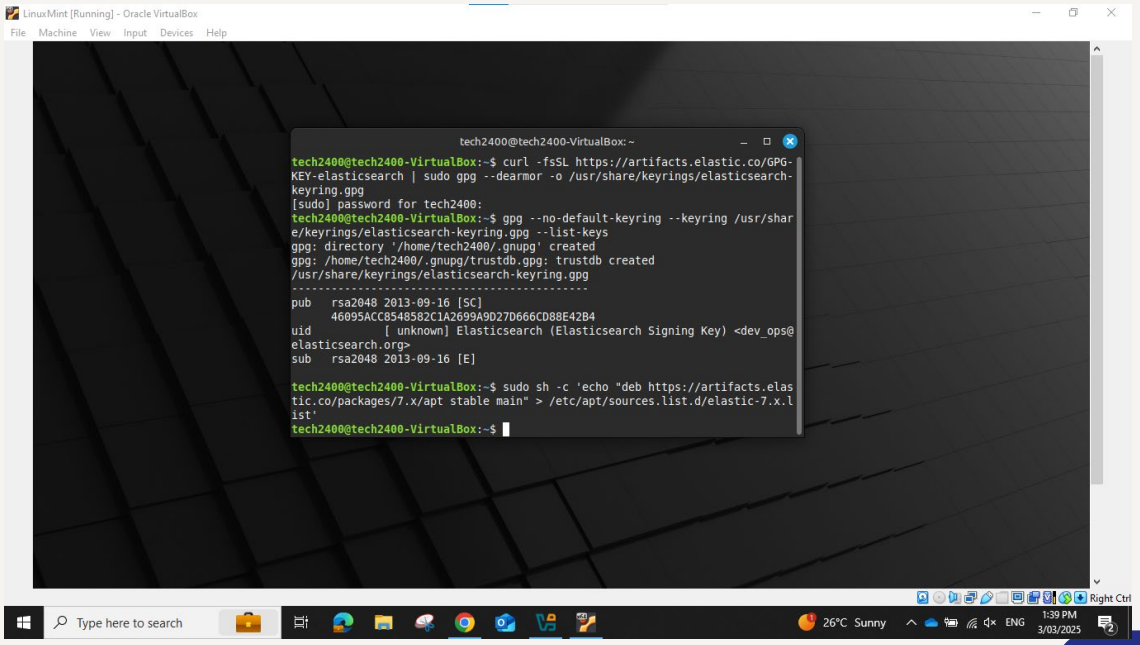
- Run gpg --no-default-keyring --keyring /usr/share/keyrings/elasticsearch-keyring.gpg --list-keys



Activity: Install Elasticsearch

Step 3: Add Elasticsearch APT repository to your system

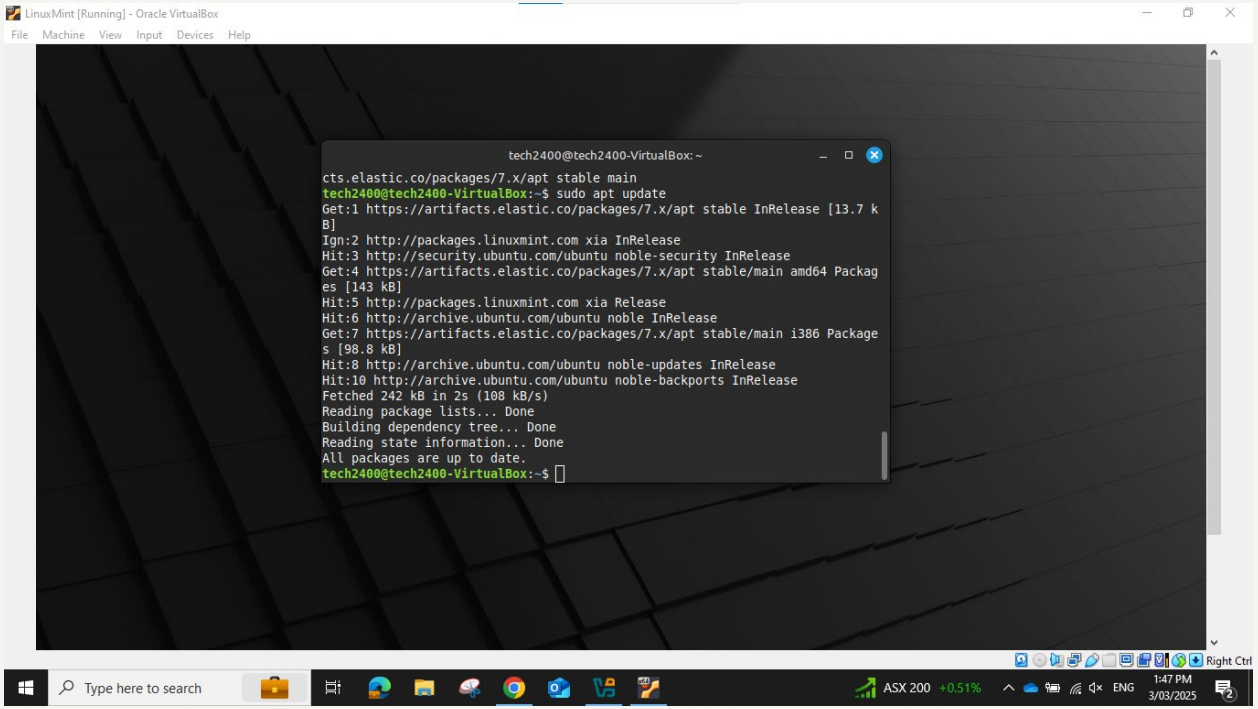
- Run `echo "deb [signed-by=/usr/share/keyrings/elasticsearch-keyring.gpg] https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee /etc/apt/sources.list.d/elastic-7.x.list`



Activity: Install Elasticsearch

Step 4: Update the APT package index

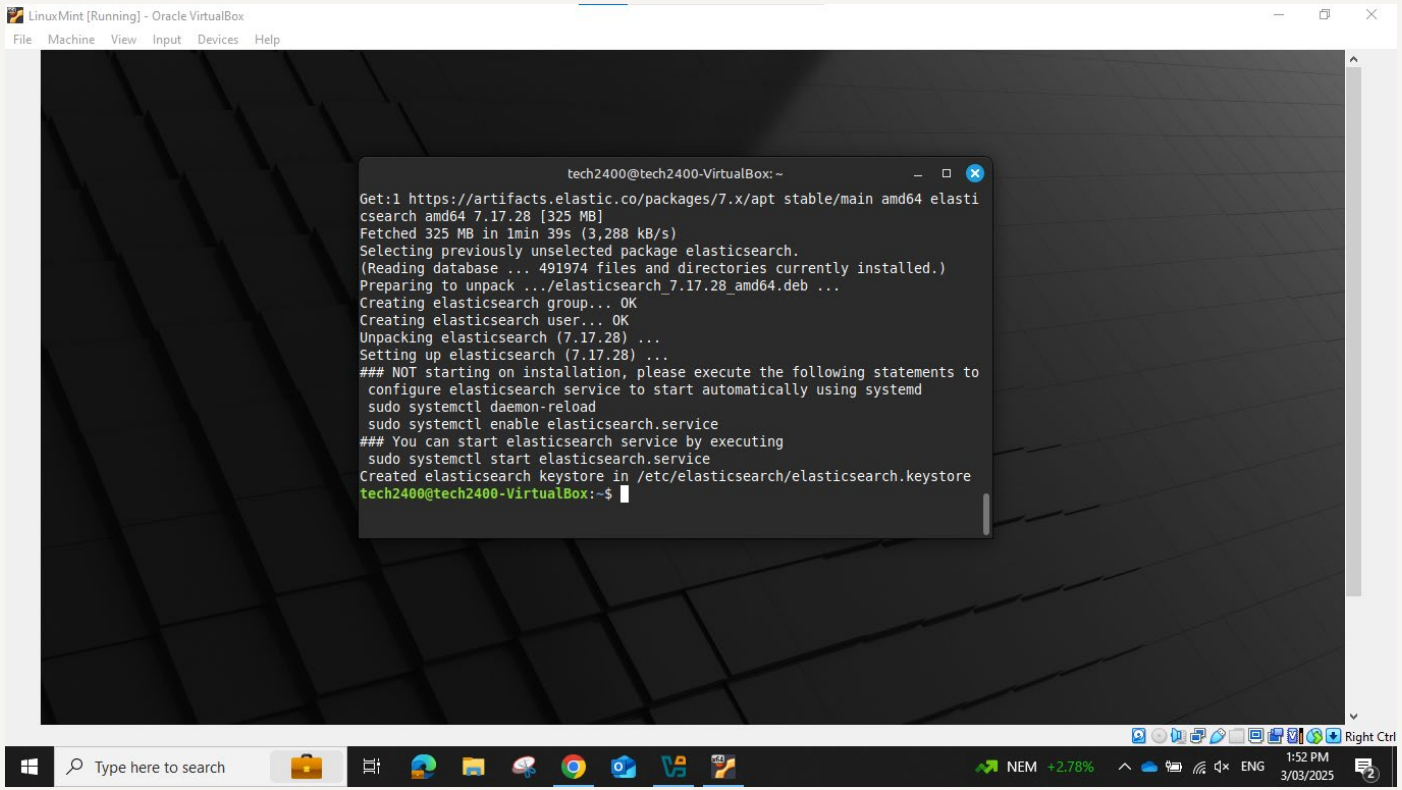
- Run sudo apt update



Activity: Install Elasticsearch

Step 5: Install Elasticsearch

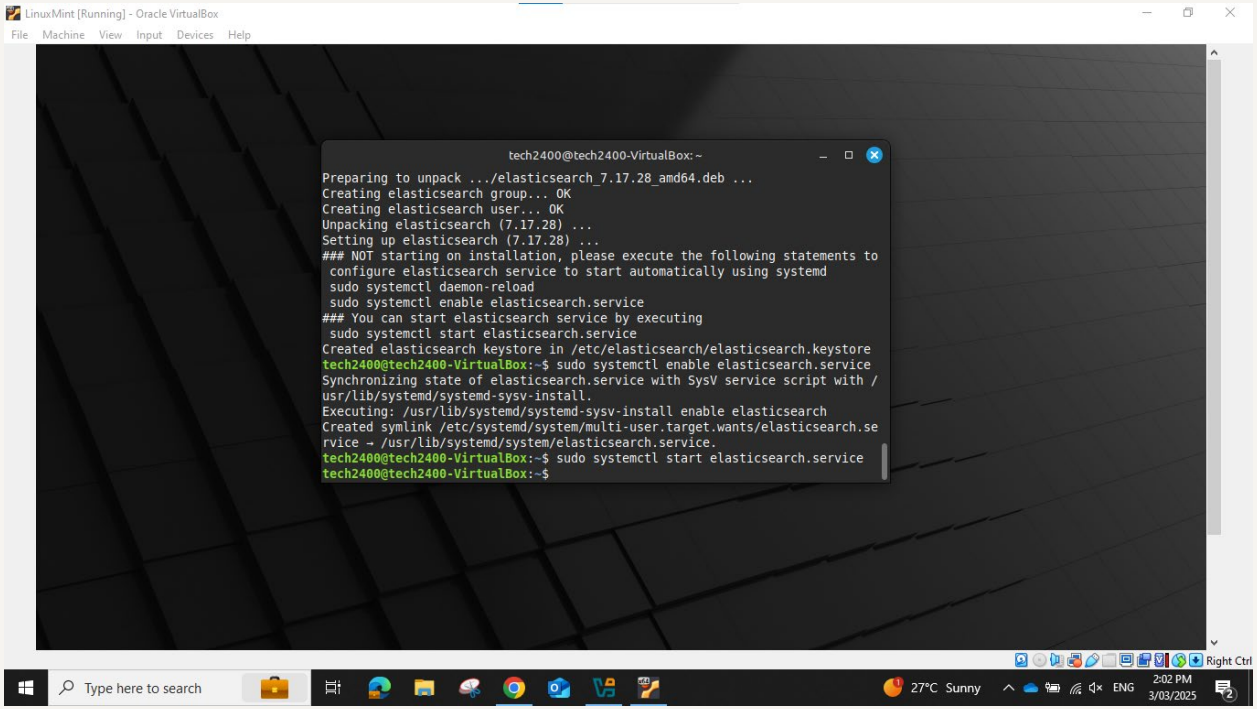
- Run sudo apt install elasticsearch -y



Activity: Install Elasticsearch

Step 6: Enable and start Elasticsearch

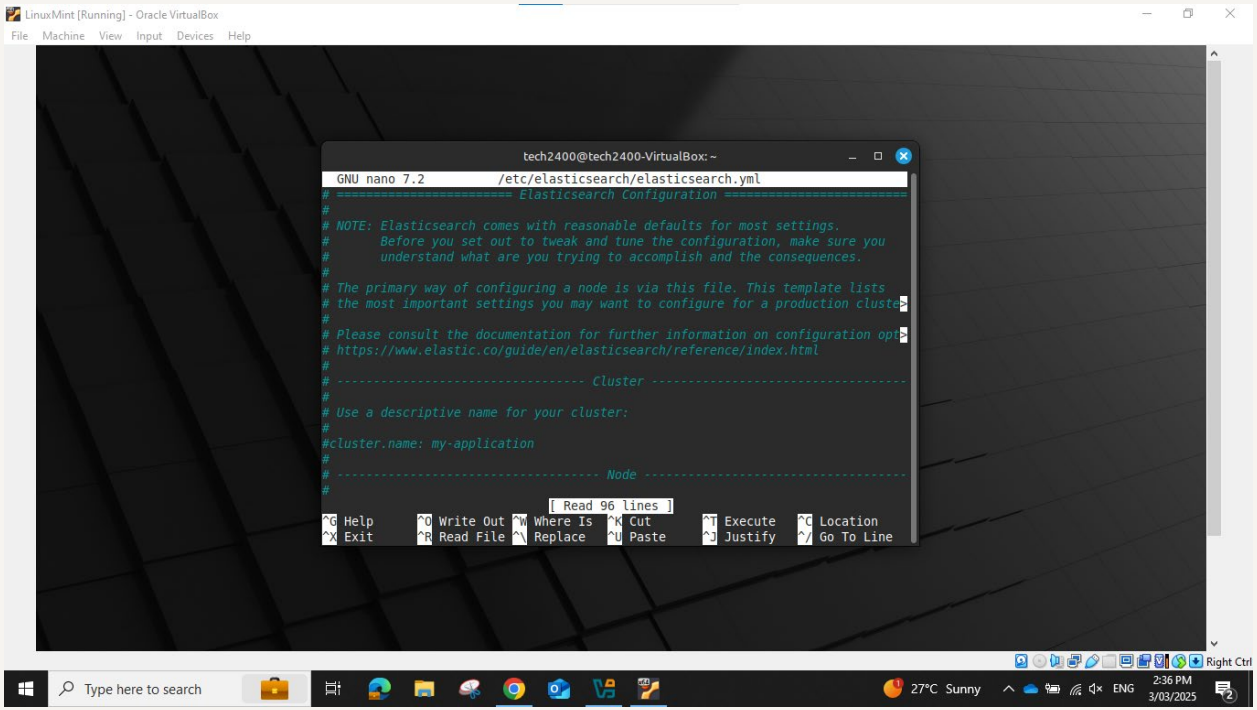
- Run sudo systemctl enable elasticsearch.service
- Run sudo systemctl start elasticsearch.service



Activity: Configure Elasticsearch

Step 1: Open the Elasticsearch configuration file

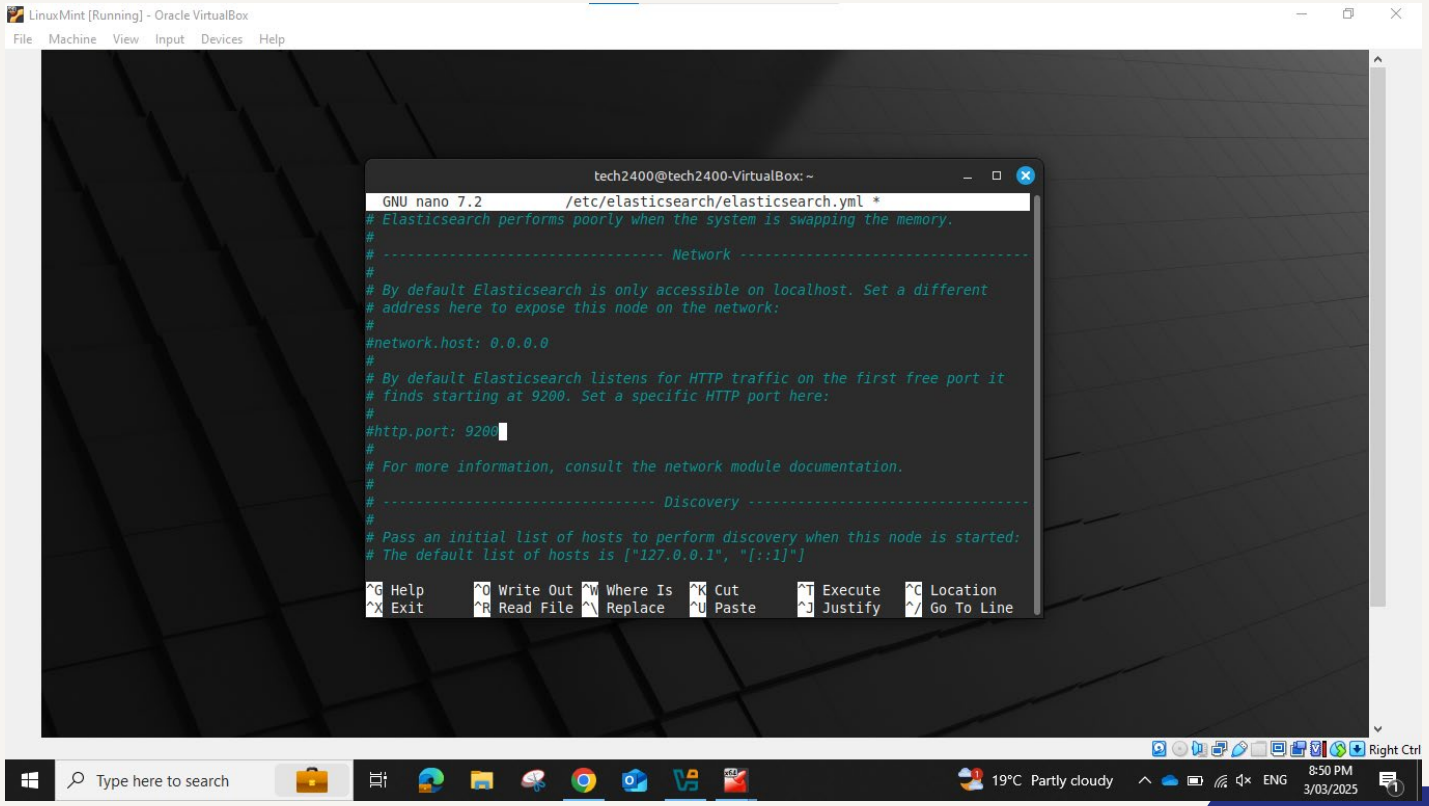
- Run sudo nano /etc/elasticsearch/elasticsearch.yml



Activity: Configure Elasticsearch

Step 2: Modify the configuration file

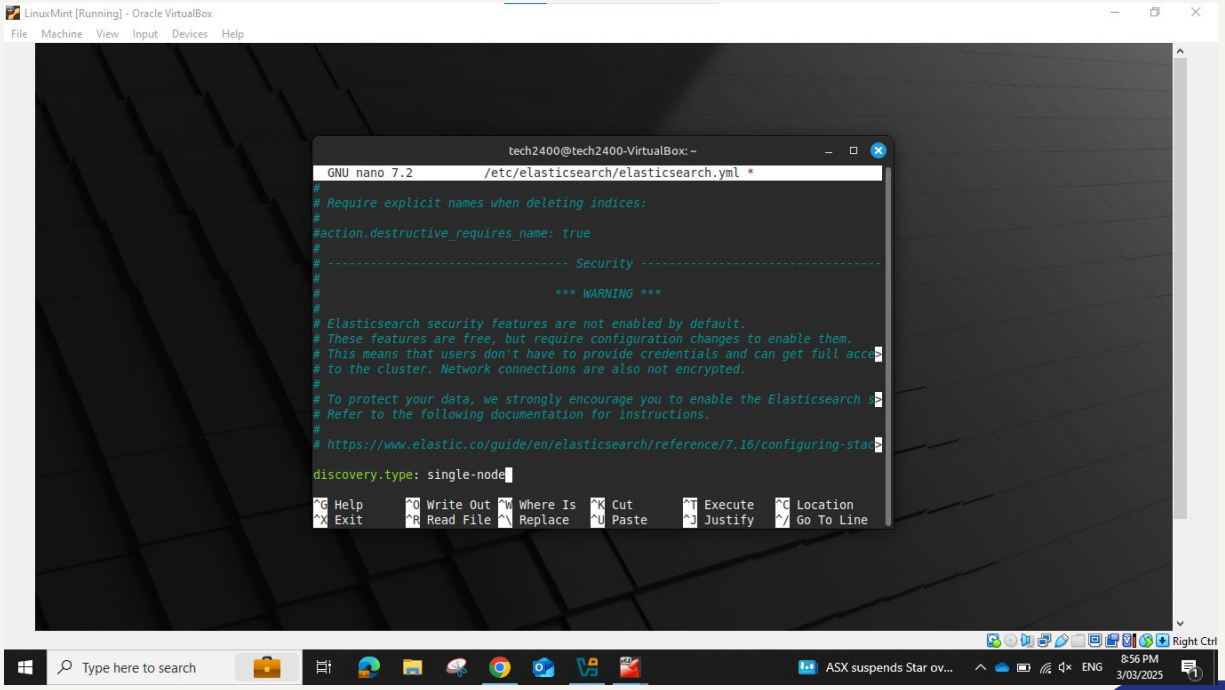
- Set the network.host line to 0.0.0.0 and http.port to 9200



Activity: Configure Elasticsearch

Step 2: Modify the configuration file

- Add an additional line under Discovery towards the bottom of the file: discovery.type: single-node
- Save (Ctrl+O), Enter to accept file name, and Close (Ctlr+X)



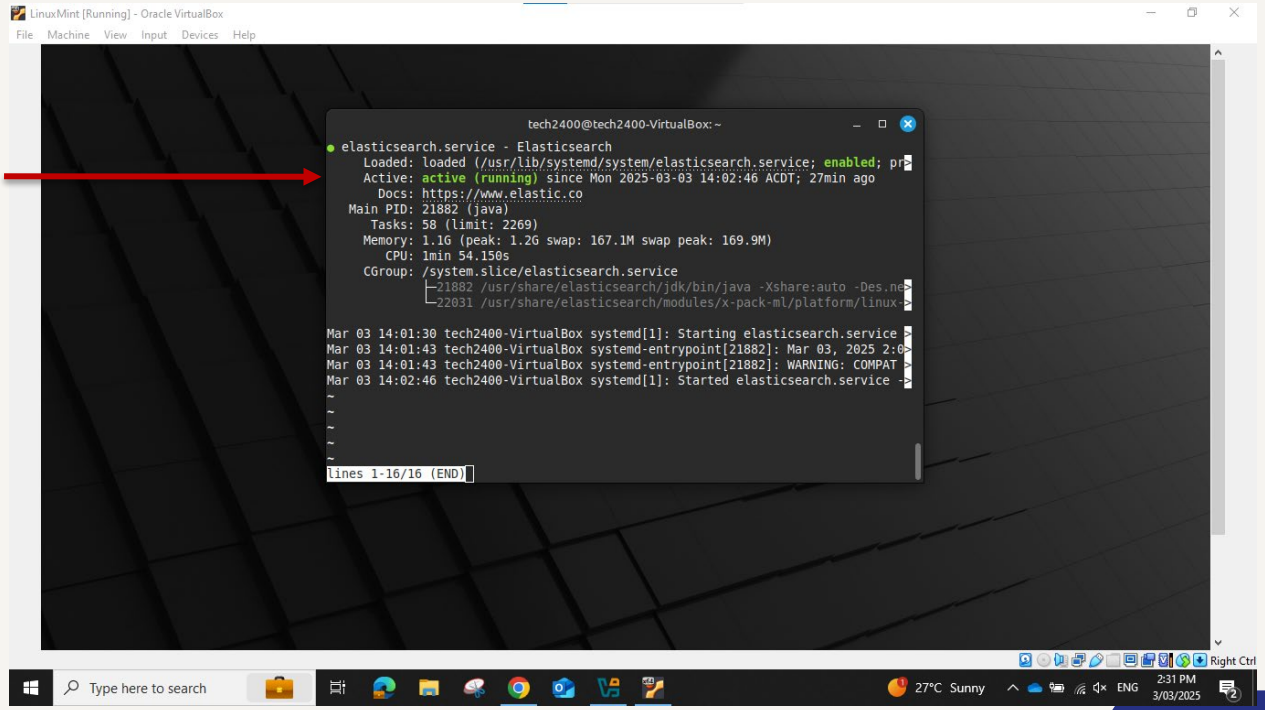
KAPLAN BUSINESS SCHOOL AUSTRALIA

Activity: Configure Elasticsearch

Step 3: Verify that Elasticsearch is running

- Run sudo systemctl status elasticsearch
- The command should return a response saying it has been active (running) since [date and time]

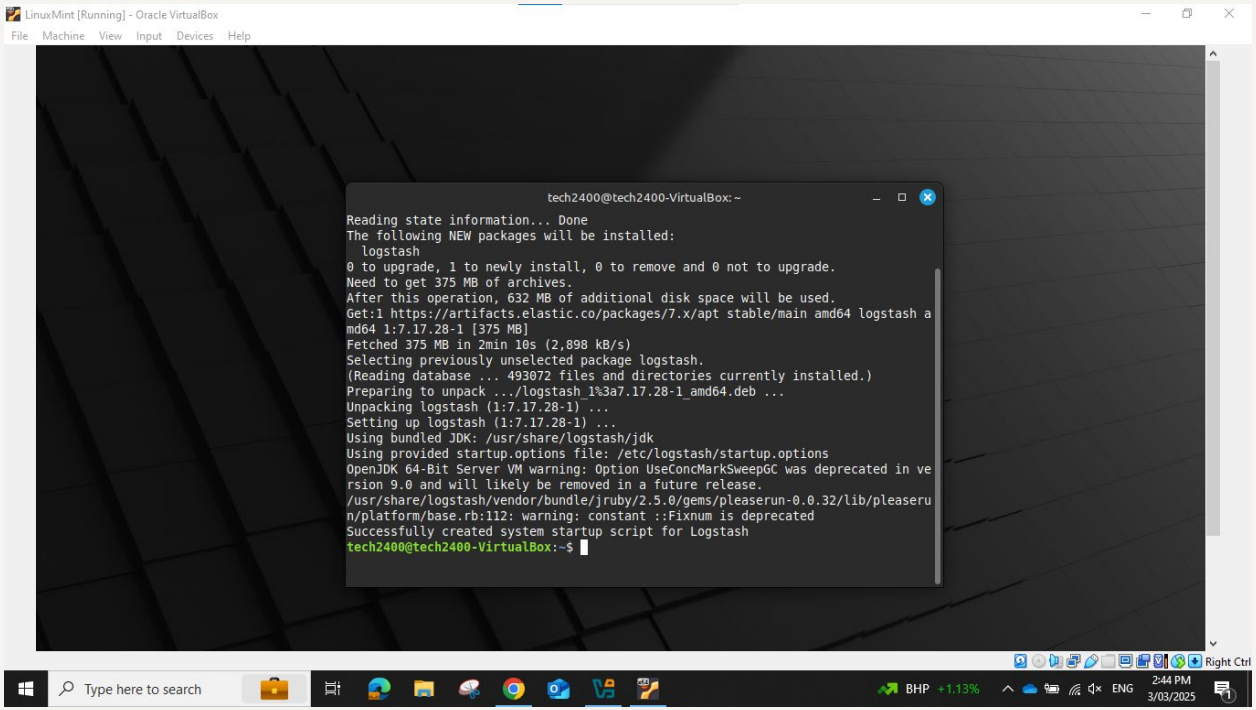
Indicates Elasticsearch is running



Activity: Install Logstash

Step 1: Install Logstash

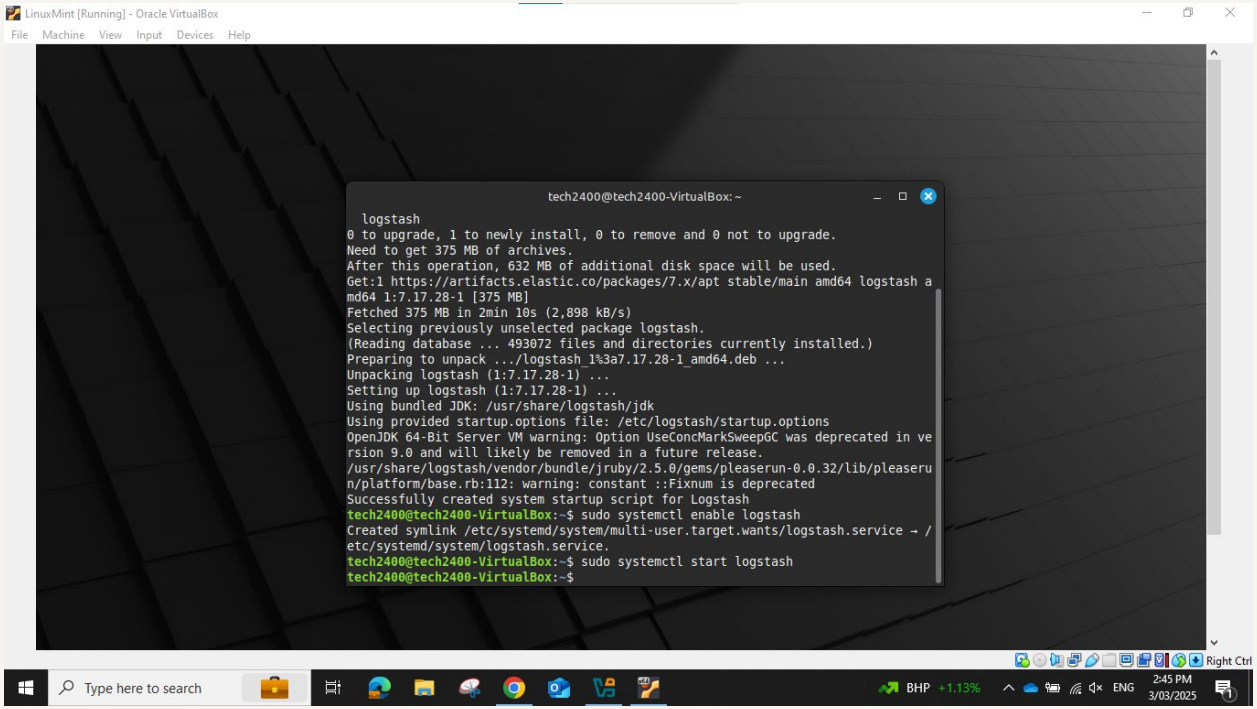
- Run sudo apt install logstash -y



Activity: Install Logstash

Step 2: Enable and start Logstash

- Run sudo systemctl enable logstash
- Run sudo systemctl start logstash

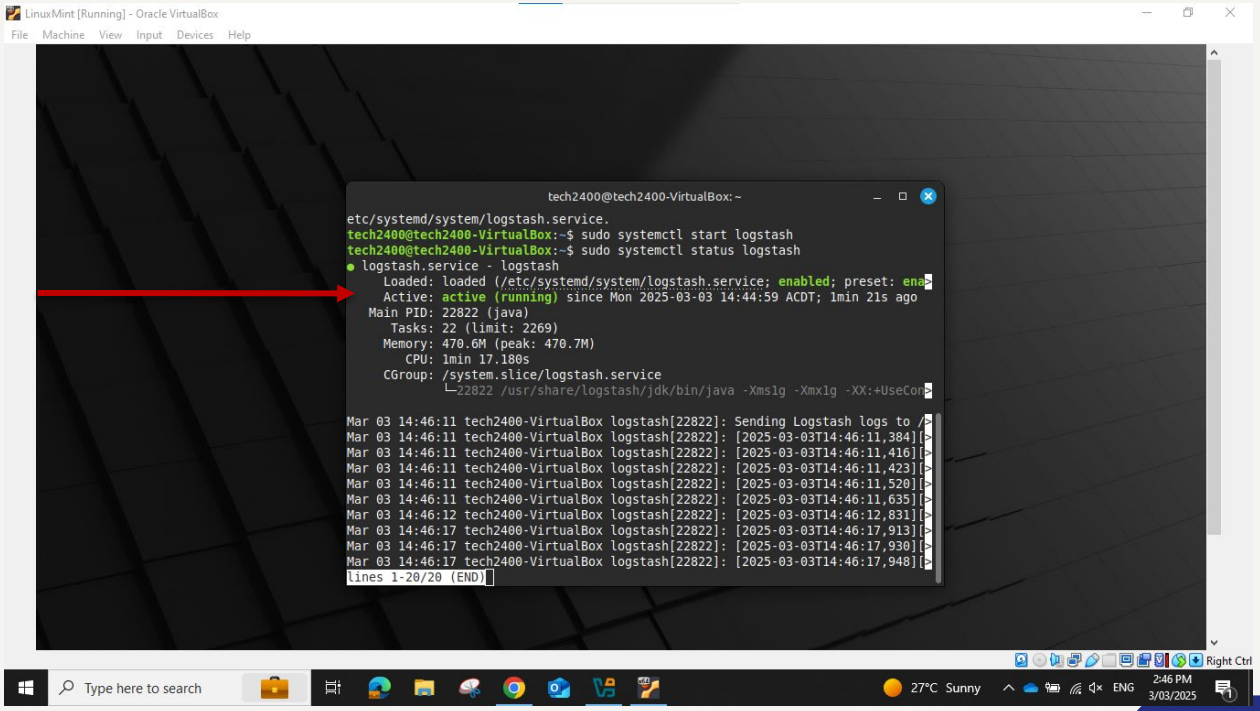


Activity: Install Logstash

Step 3: Verify that Logstash is running

- Run sudo systemctl status logstash
- If Logstash is running, the command will return a response saying it has been active (running) since [date and time]

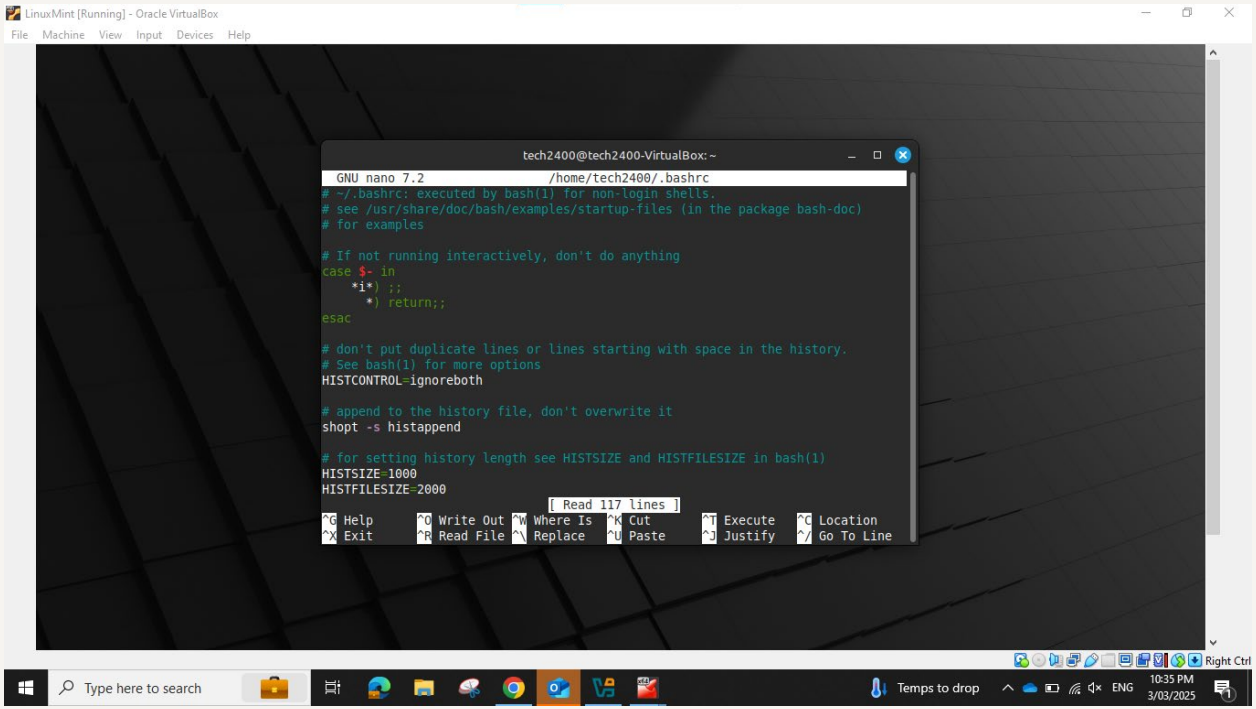
Indicates Logstash is running



Activity: Install Logstash

Step 4: Add Logstash to your $PATH

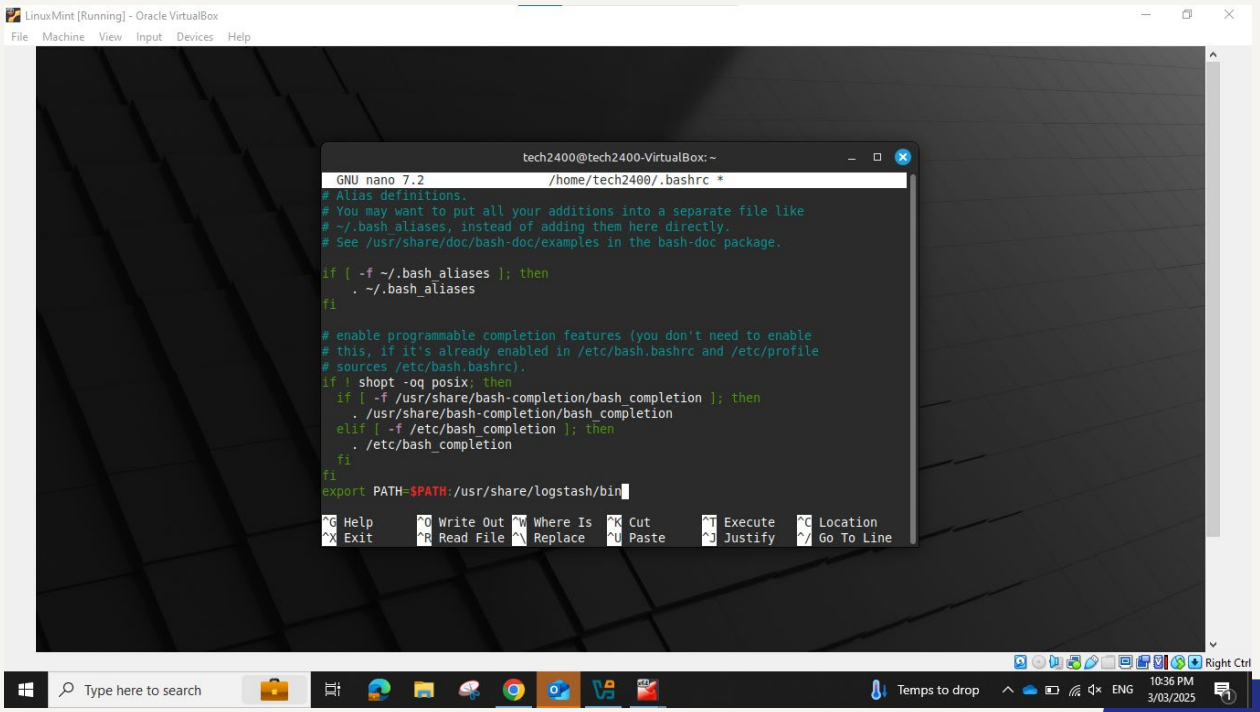
- Run `nano ~/.bashrc` to open your shell configuration file



Activity: Install Logstash

Step 4: Add Logstash to your $PATH

- Navigate to the end of the file and add the following line:
  `export PATH=$PATH:/usr/share/logstash/bin`
- Save (Ctrl+O) > Enter > Close (Ctrl+X)

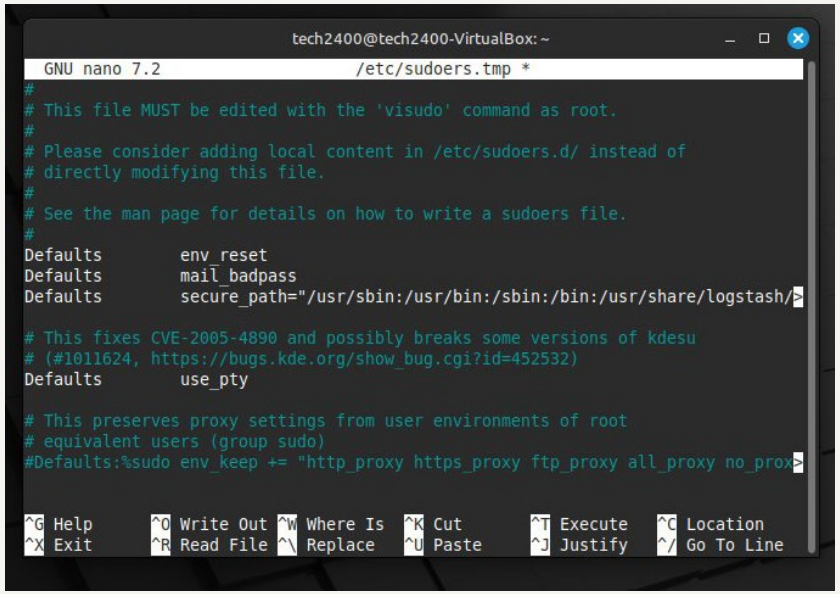


Activity: Install Logstash

Step 4: Add Logstash to your $PATH

- Open sudo visudo to configure sudo to preserve $PATH
- Navigate to Defaults secure_path
- Add the path to Logstash

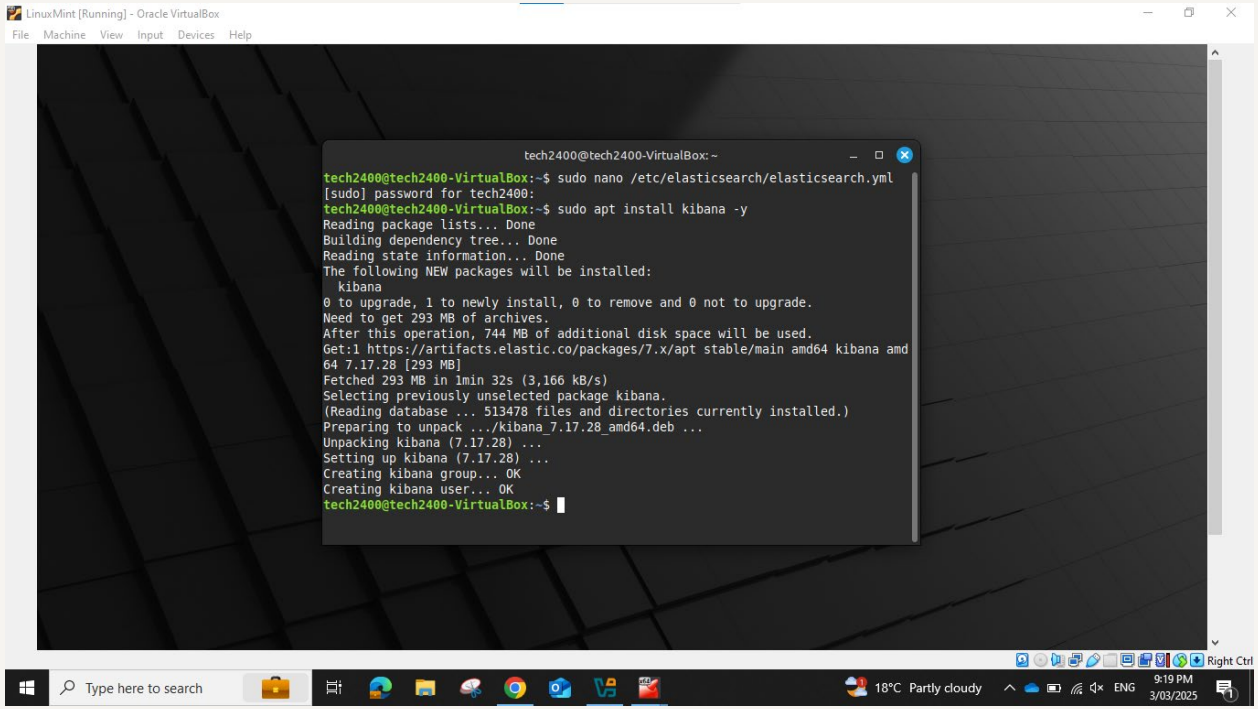
```
Defaults
secure_path="/usr/sbin:/usr/bin:/sbin:/bin:/usr/share/logstash/
bin"
```



Activity: Install Kibana

Step 1: Install Kibana

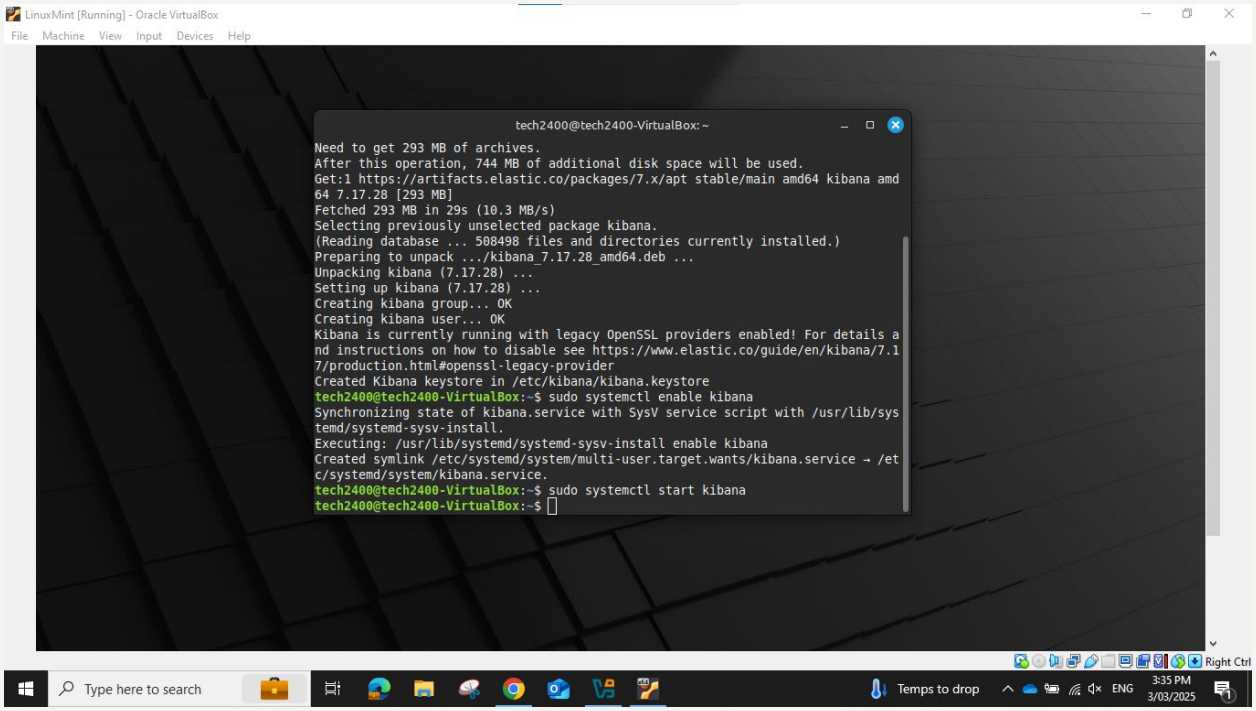
- Run sudo apt install kibana -y



Activity: Install Kibana

Step 2: Enable and start Kibana

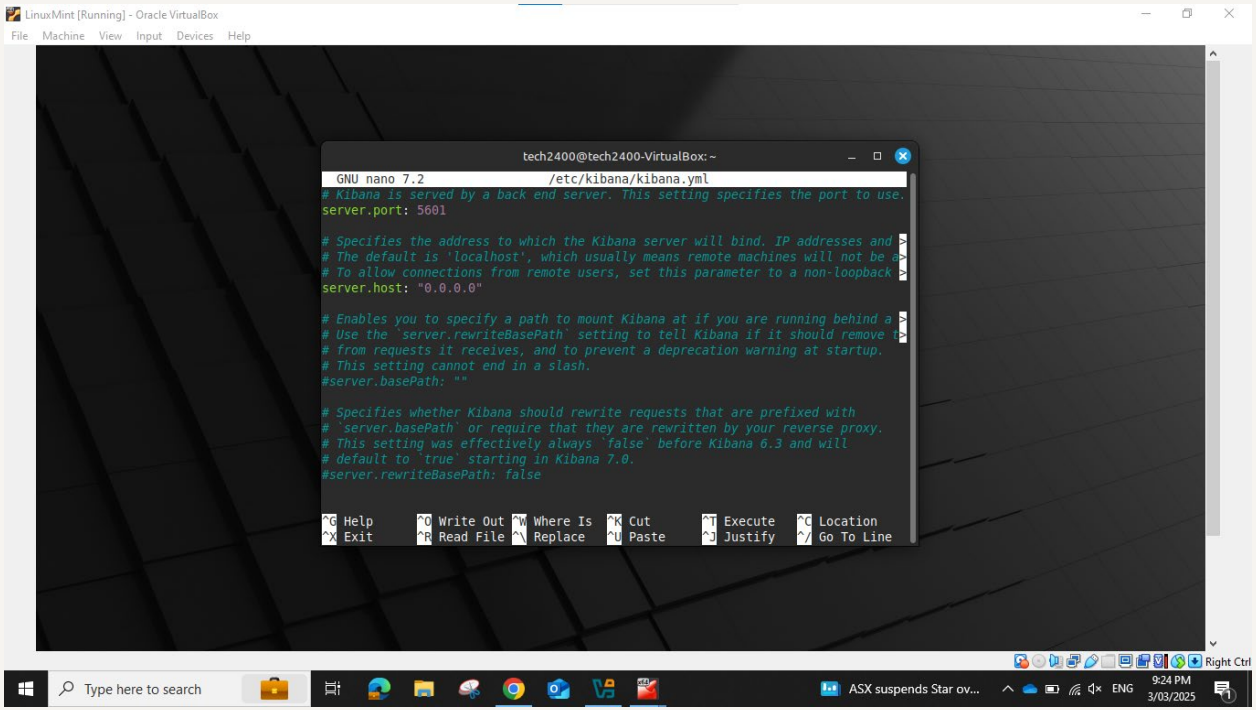
- Run sudo systemctl enable kibana
- Run sudo systemctl start kibana



Activity: Install Kibana

Step 3: Edit the Kibana configuration file

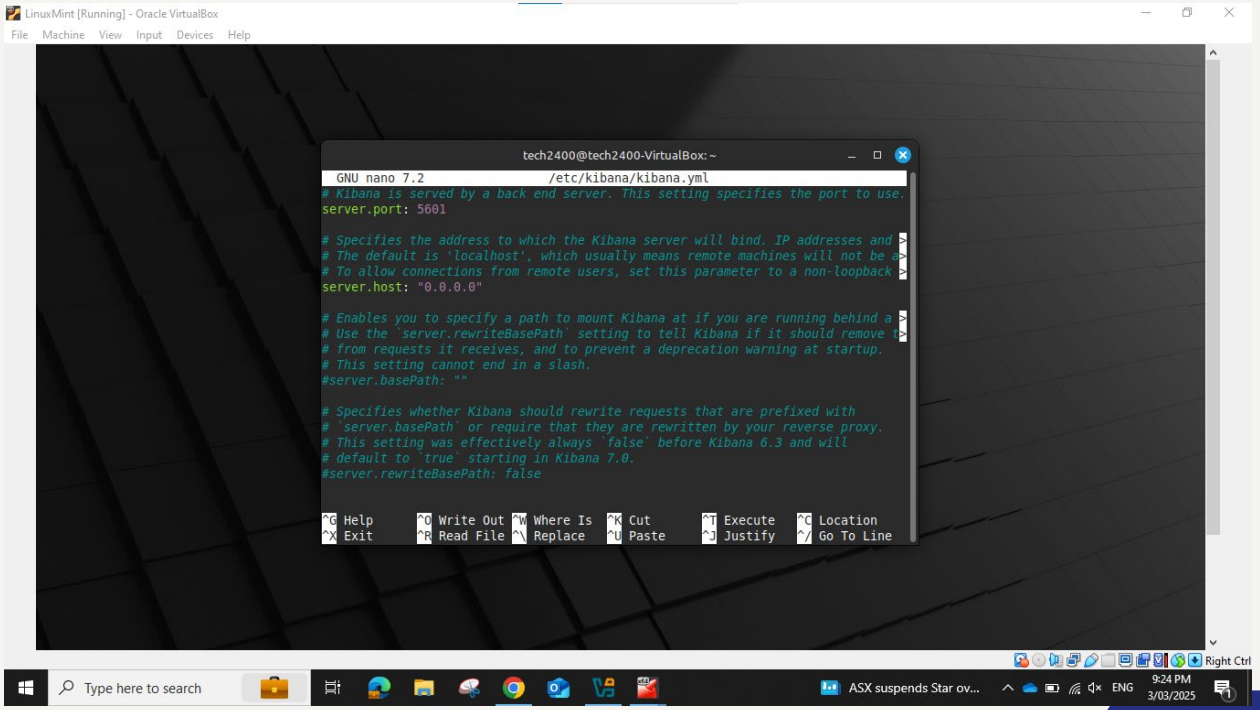
- Run sudo nano /etc/kibana/kibana.yml



Activity: Install Kibana

Step 3: Edit the Kibana configuration file

- Uncomment server.host
- Set the IP to 0.0.0.0 to listen on all interfaces
- Uncomment server.port (leave 5601)

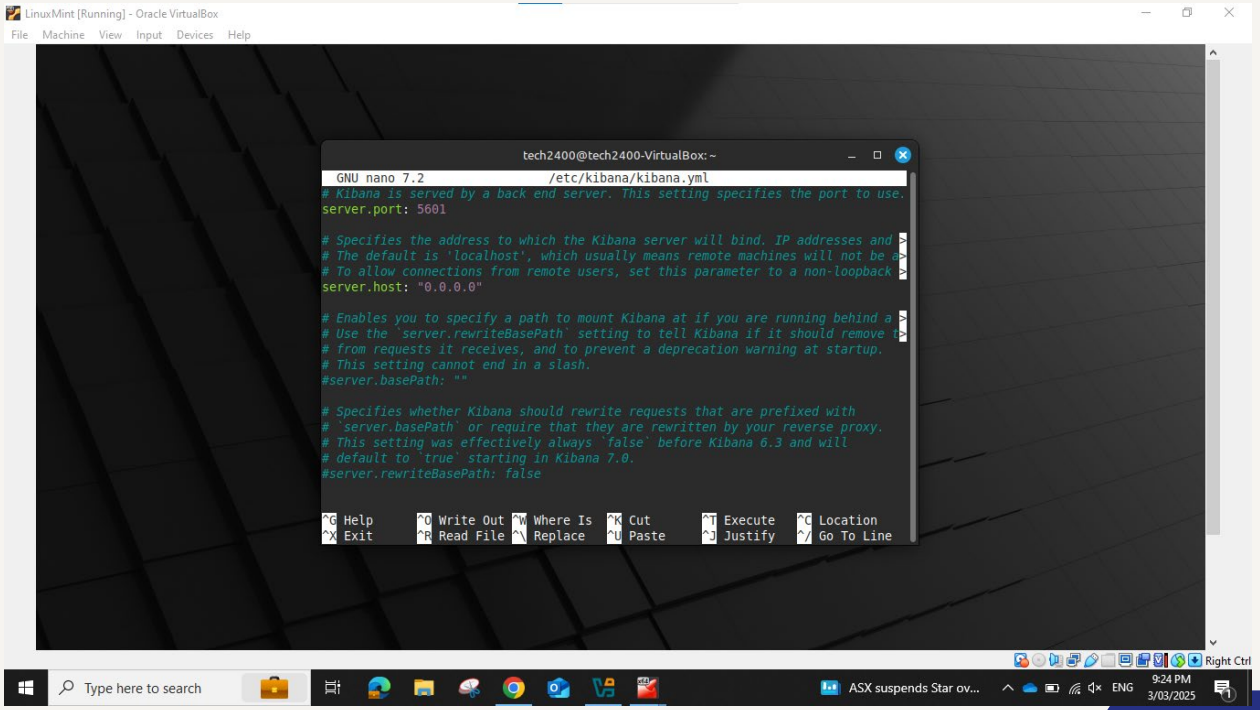


KAPLAN BUSINESS SCHOOL AUSTRALIA

Activity: Install Kibana

Step 3: Edit the Kibana configuration file

- Save (Ctrl+O)
- Press Enter key to accept file name
- Close (Ctrl+X)

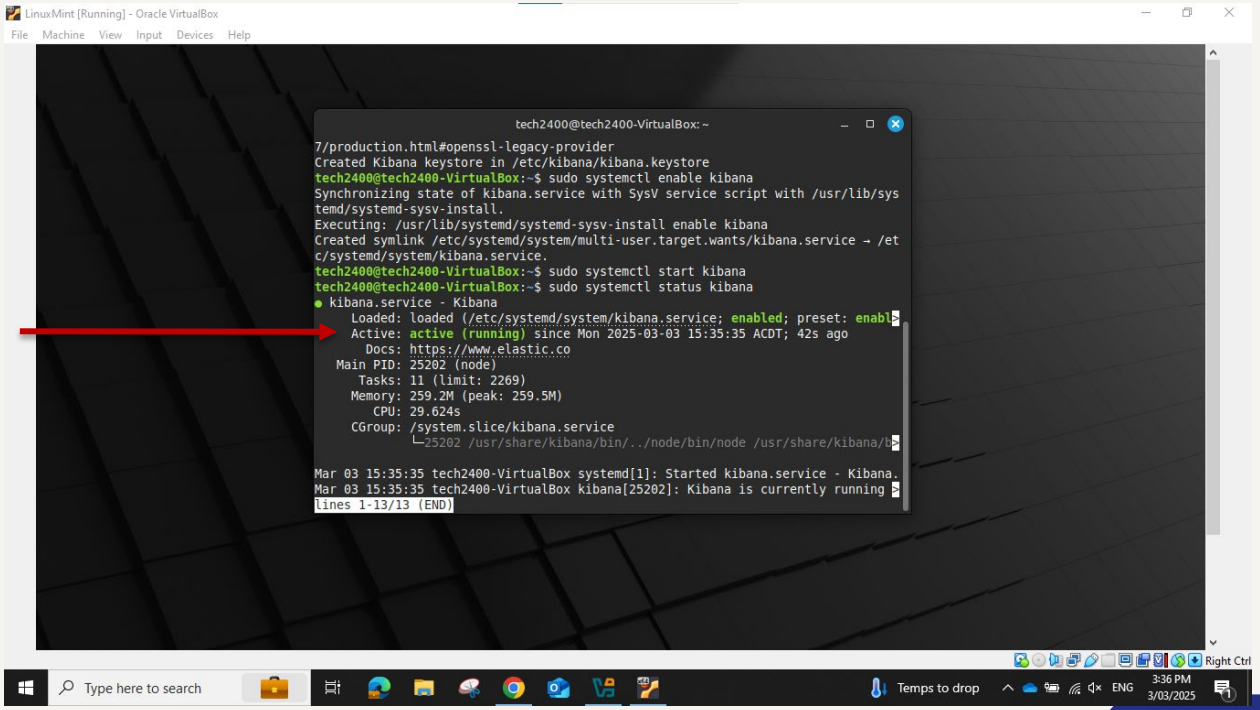


Activity: Install Kibana

Step 4: Verify that Kibana is running

- Run sudo systemctl status kibana
- If Kibana is running, the command will return a response saying it has been active (running) since [date and time]

Indicates Kibana is running



Activity:
Generate SSH Logs for Analysis

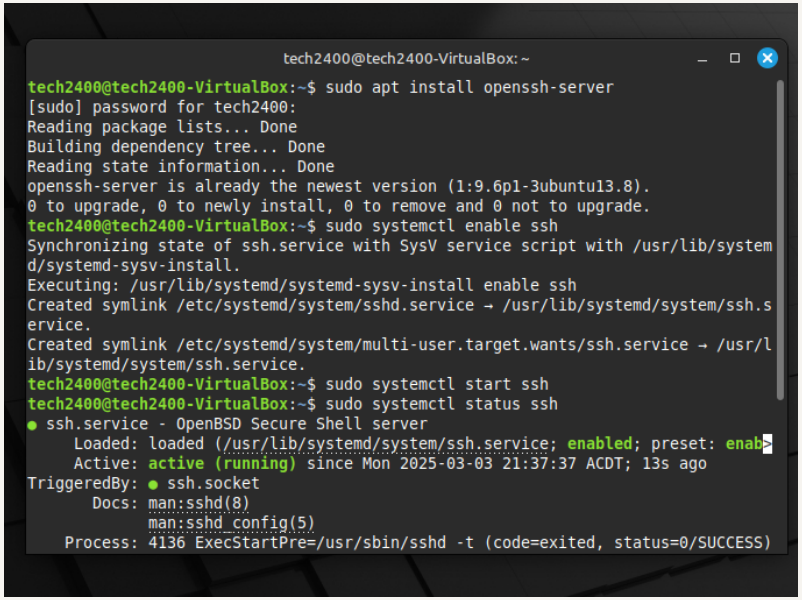
Goal: To simulate a security scenario where unauthorised users attempt to gain access to a system

Step 1: Ensure that SSH is enabled

1) Open Terminal on Linux Mint
2) Run sudo systemctl status ssh to ensure SSH is enabled.

If not:

sudo apt install openssh-server
sudo systemctl start ssh
sudo systemctl enable ssh

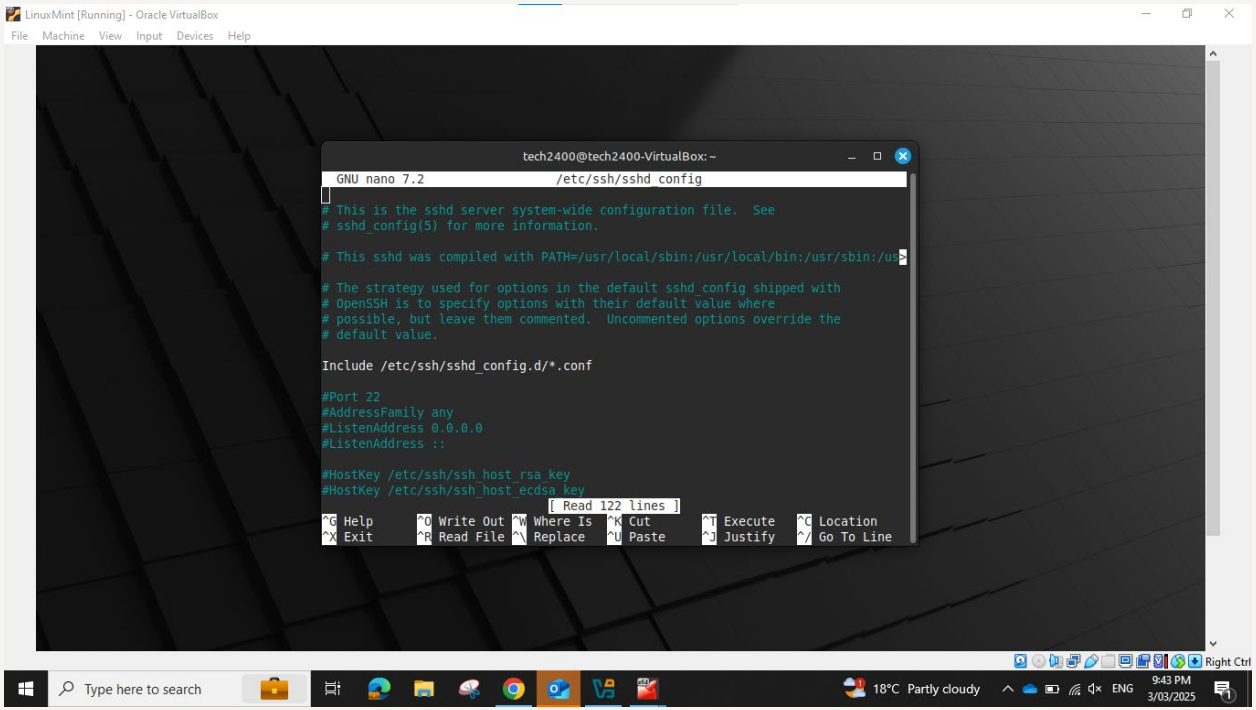


Activity:
Generate SSH Logs for Analysis

Step 2: Configure SSH to log unsuccessful login attempts

- Edit the SSH configuration file

```
sudo nano /etc/ssh/sshd_config
```



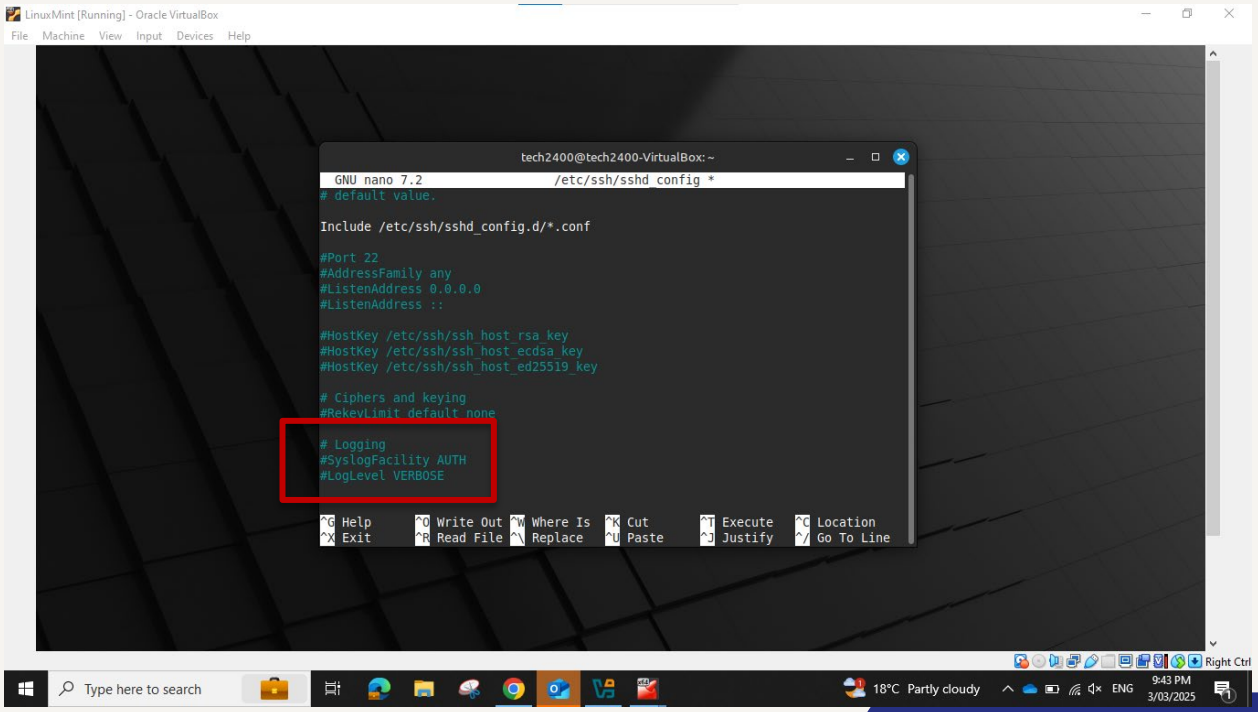
Activity:
Generate SSH Logs for Analysis

Step 2: Configure SSH to log unsuccessful login attempts

- Ensure the following logging parameters are set to ensure detailed logging of SSH activity, including failed login attempts

LogLevel VERBOSE

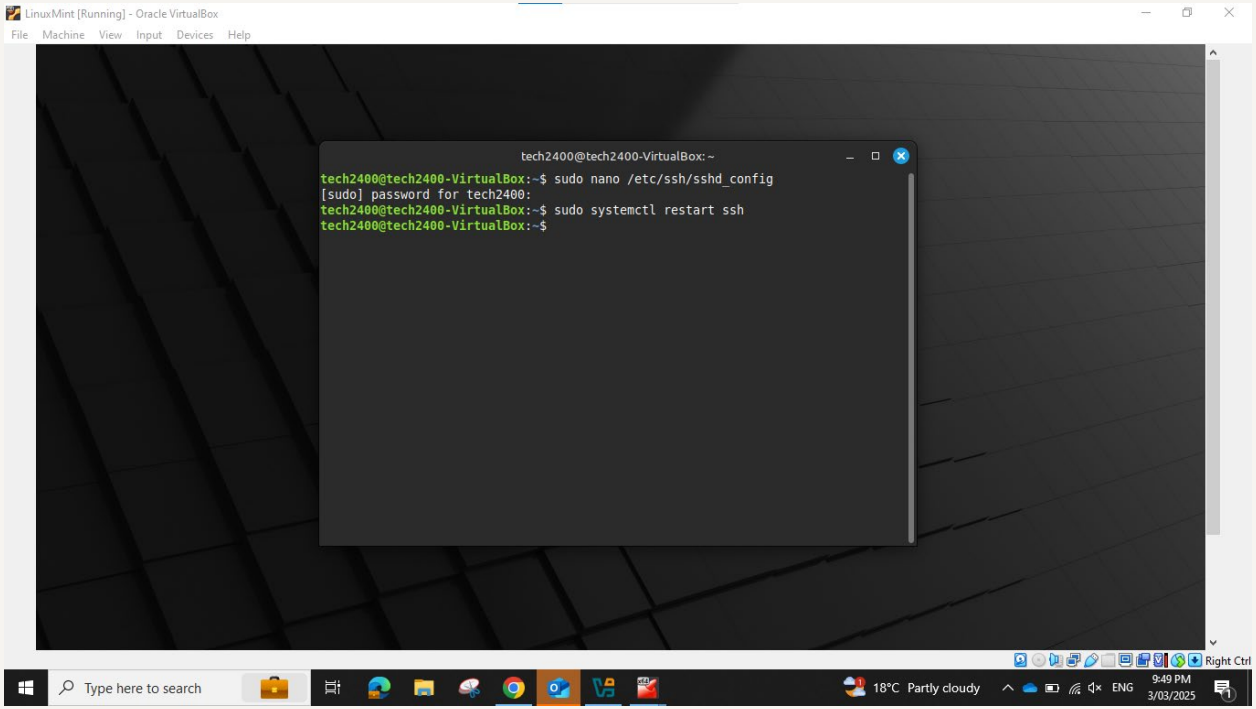
- Save (Ctrl+O)
- Enter
- Close (Ctrl+X)



Activity:
Generate SSH Logs for Analysis

Step 2: Configure SSH to log unsuccessful login attempts

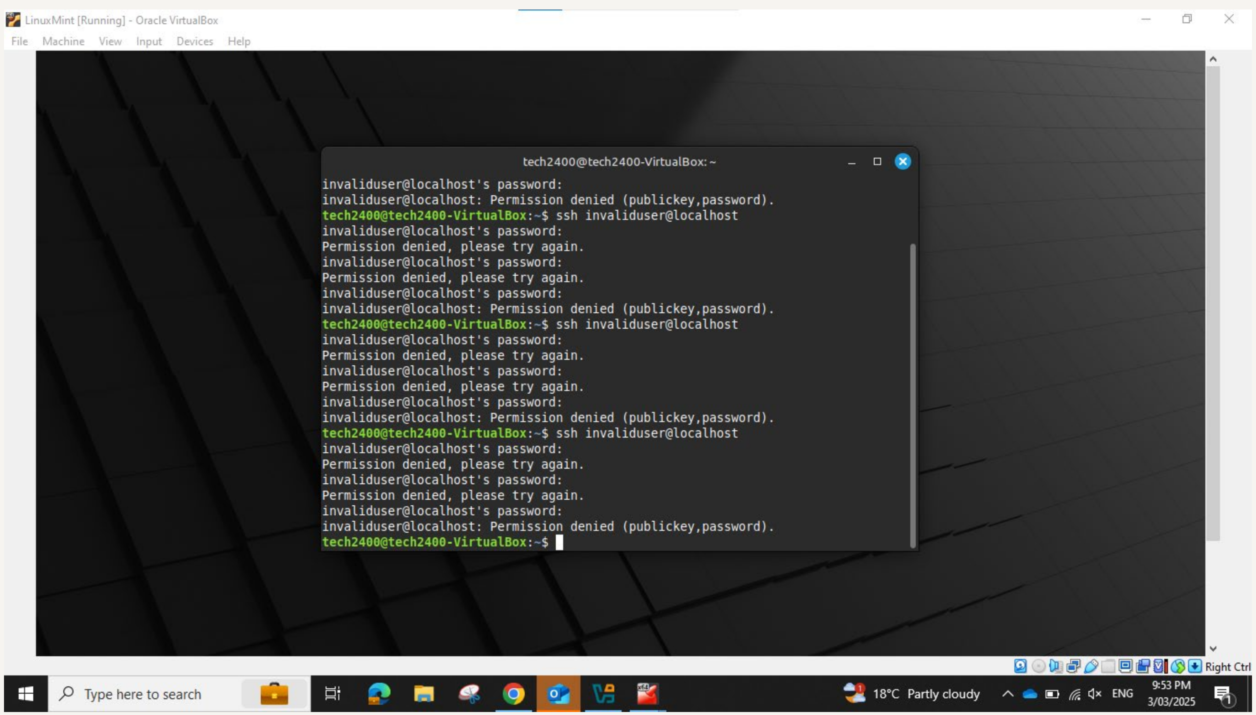
- Restart SSH for the changes to take effect
- sudo systemctl restart ssh



Activity:
Generate SSH Logs for Analysis

Step 3: Simulate unauthorised login attempts

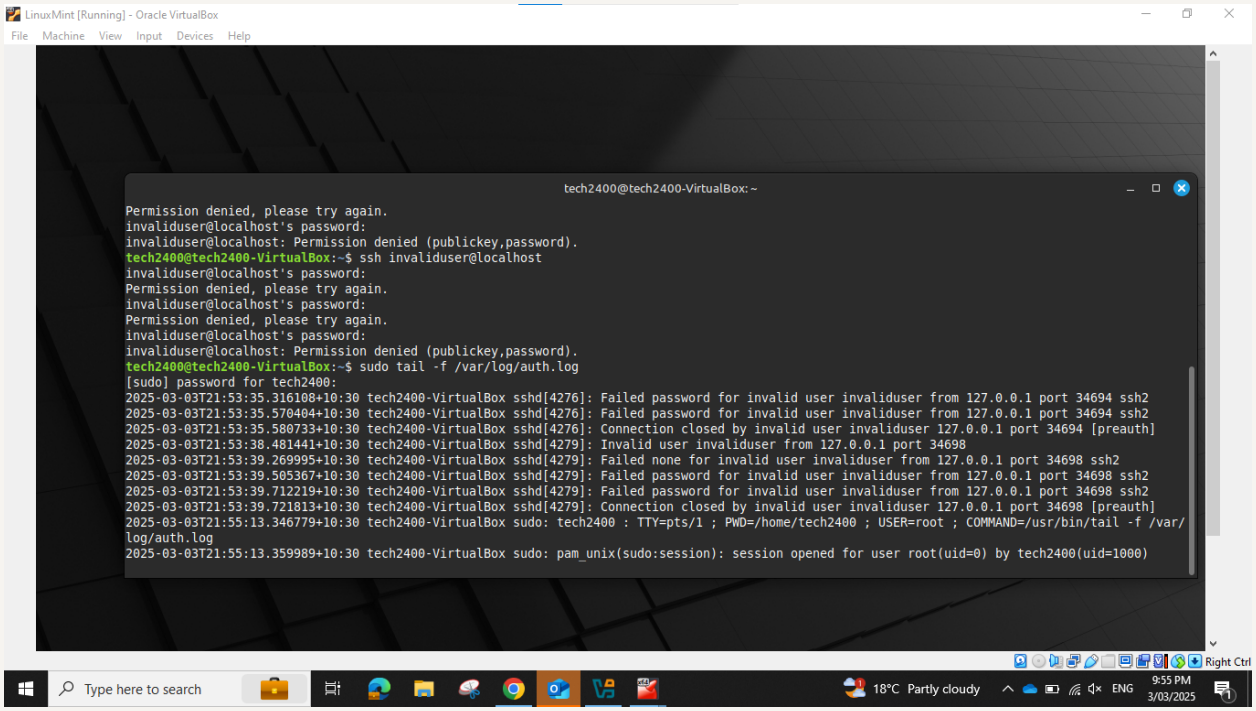
- Use the ssh command with invalid credentials several times
- ssh invaliduser@localhost



Activity:
Generate SSH Logs for Analysis

Step 3: Simulate unauthorised login attempts

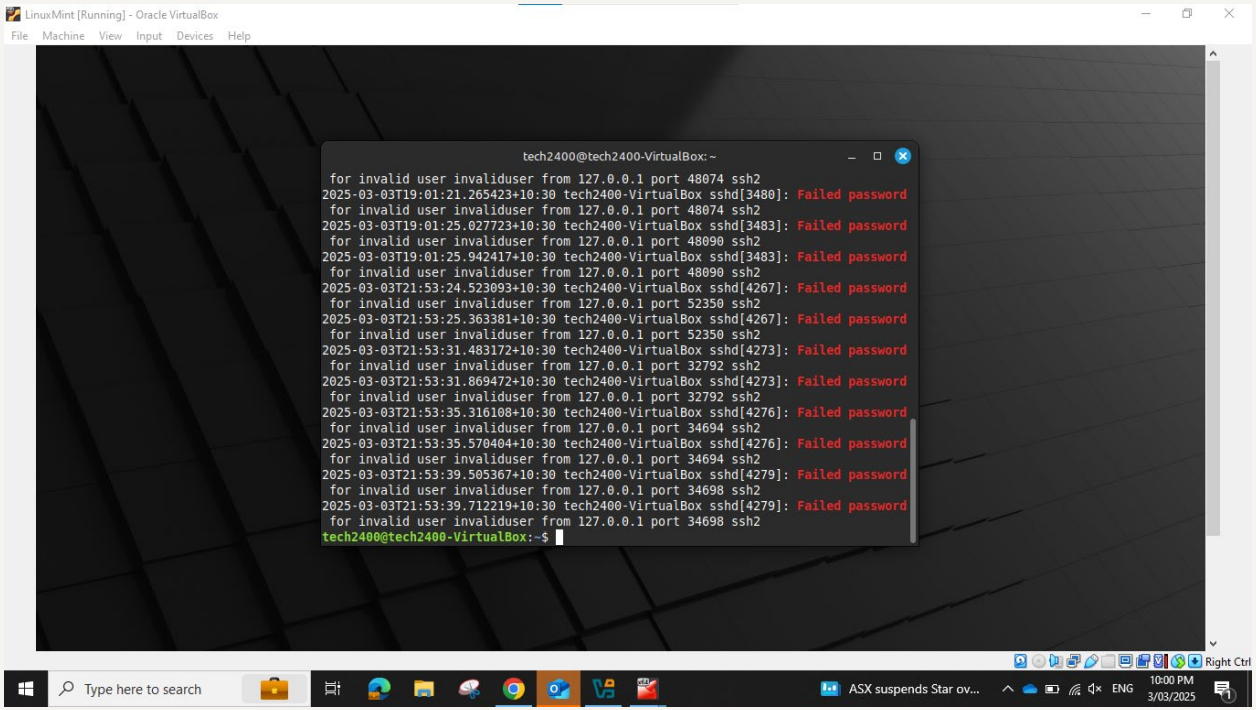
- View the generated logs: sudo tail –f /var/log/auth.log



Activity:
Generate SSH Logs for Analysis

Step 3: Simulate unauthorised login attempts

- Search logs for failed login attempts



Activity: Ingesting Logs into ELK

Goal: To process and store logs in Elasticsearch for analysis
Note: Logstash is the data processing pipeline that will pull in logs, process them, and forward them to Elasticsearch

Step 1: Create a Logstash configuration file

- Run `sudo nano /etc/logstash/conf.d/ssh_logs.conf`

Activity: Ingesting Logs into ELK

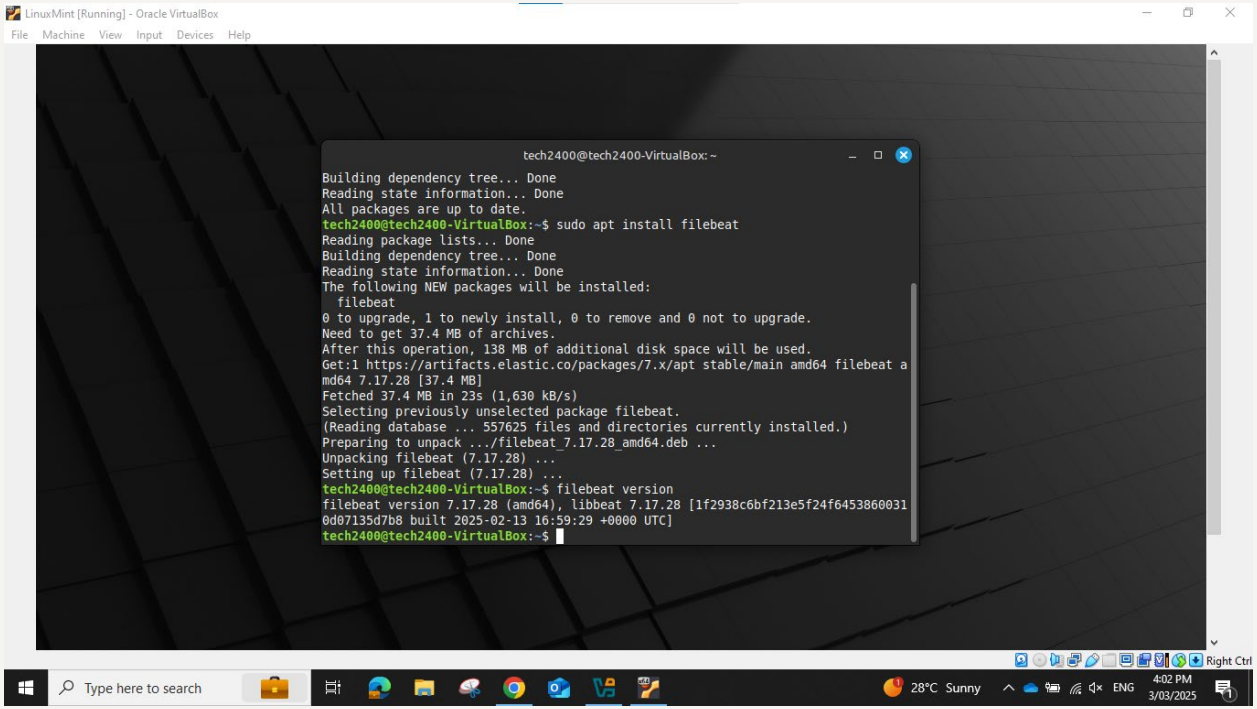
Step 1: Create a Logstash configuration file

- sudo nano /etc/logstash/conf.d/ssh_logs.conf

Activity: Ingesting Logs into ELK

Step 2: Verify Filebeat installation

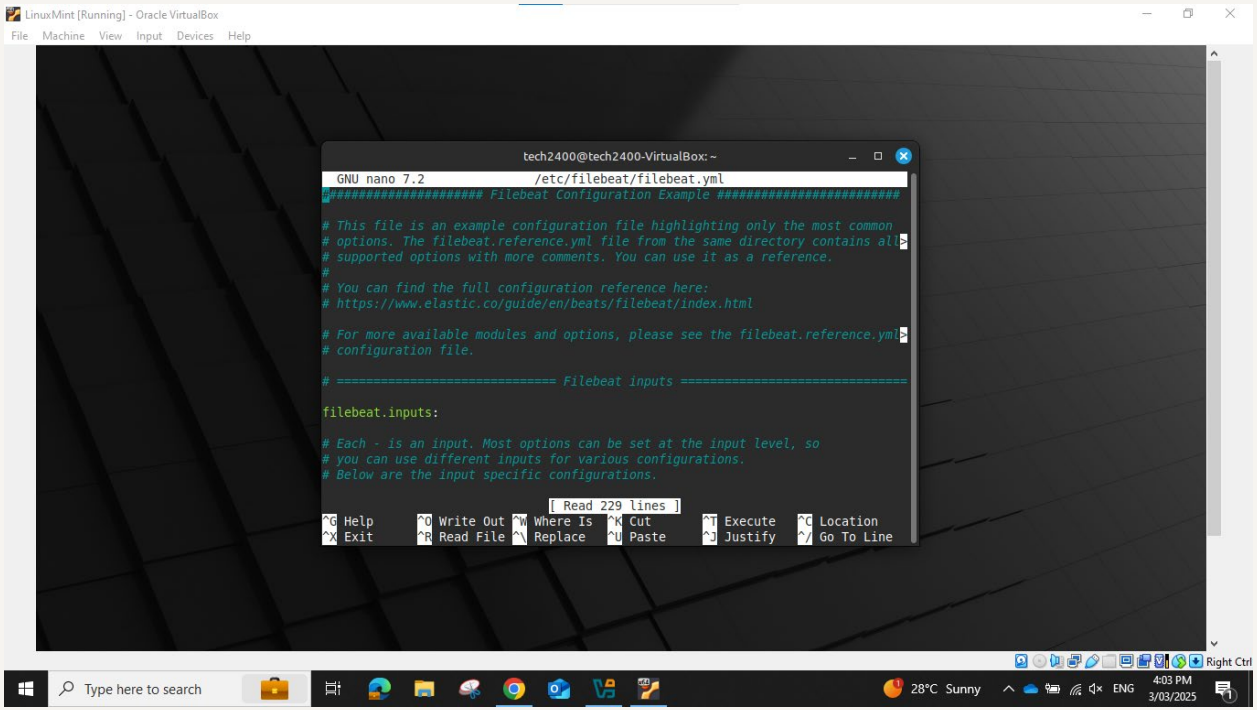
- Run filebeat version



Activity: Ingesting Logs into ELK

Step 3: Configure Filebeat

- Open Filebeat configuration file by running
- sudo nano /etc/filebeat/filebeat.yml

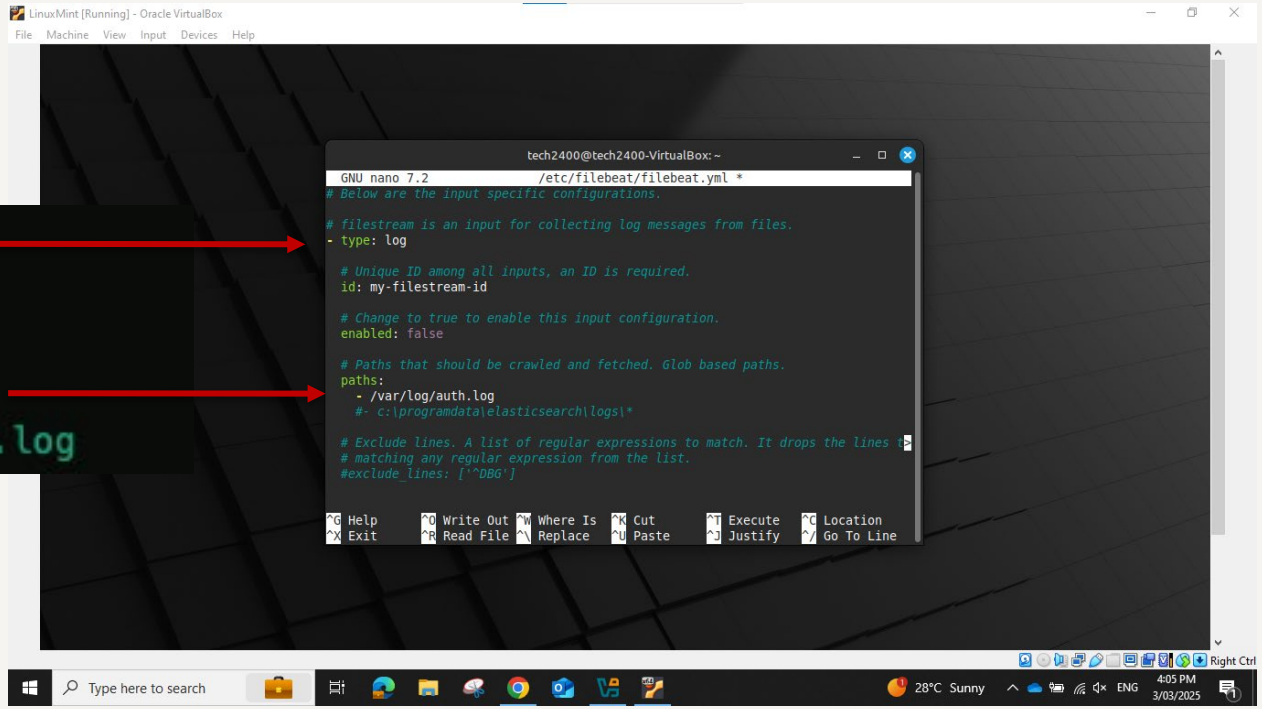


Activity: Ingesting Logs into ELK

Step 3: Configure Filebeat

- Navigate through the file to add the following configuration:

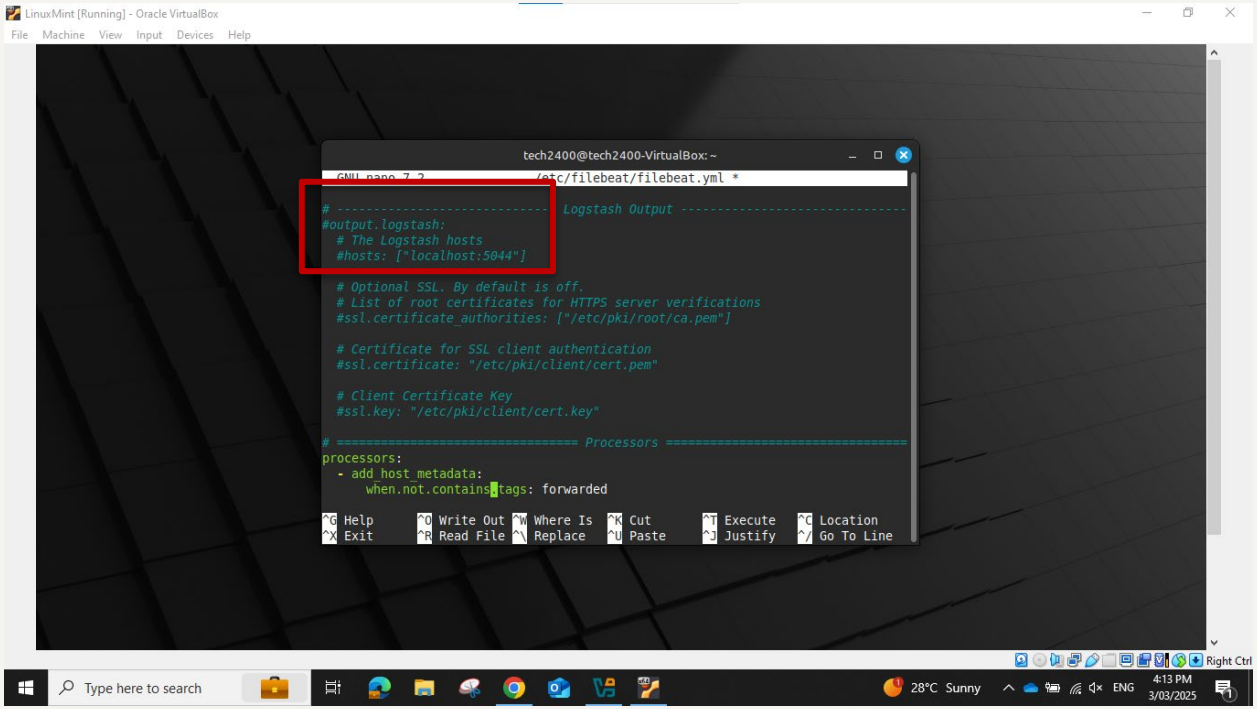
```
filebeat.inputs:
  - type: log
    paths:
      - /var/log/auth.log
```



Activity: Ingesting Logs into ELK

Step 3: Configure Filebeat

- Look for output.logstash and make sure that hosts: ["localhost:5044"]

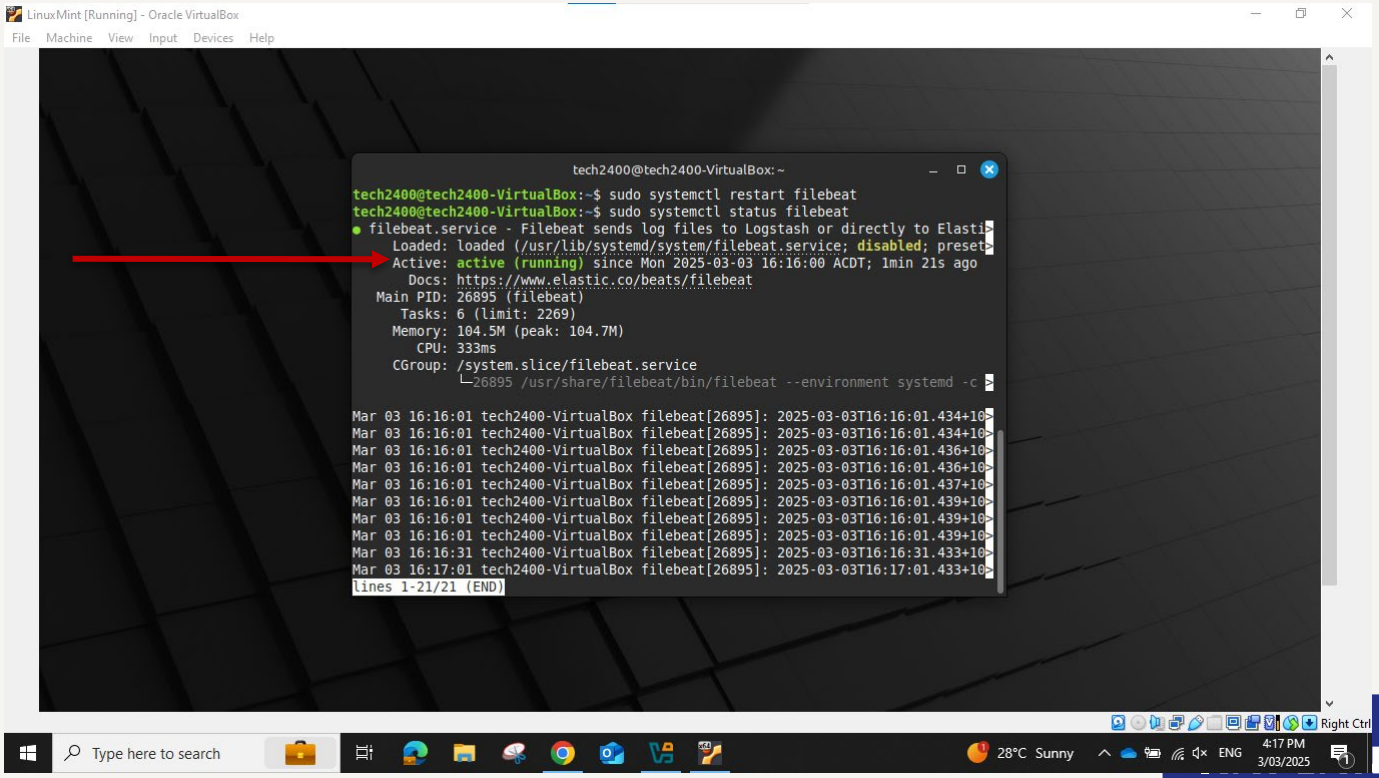


Activity: Ingesting Logs into ELK

Step 3: Configure Filebeat

- Save the file (Ctrl+O) and exit (Ctrl+X)
- Restart Filebeat: sudo systemctl restart filebeat
- Verify status: sudo systemctl restart filebeat

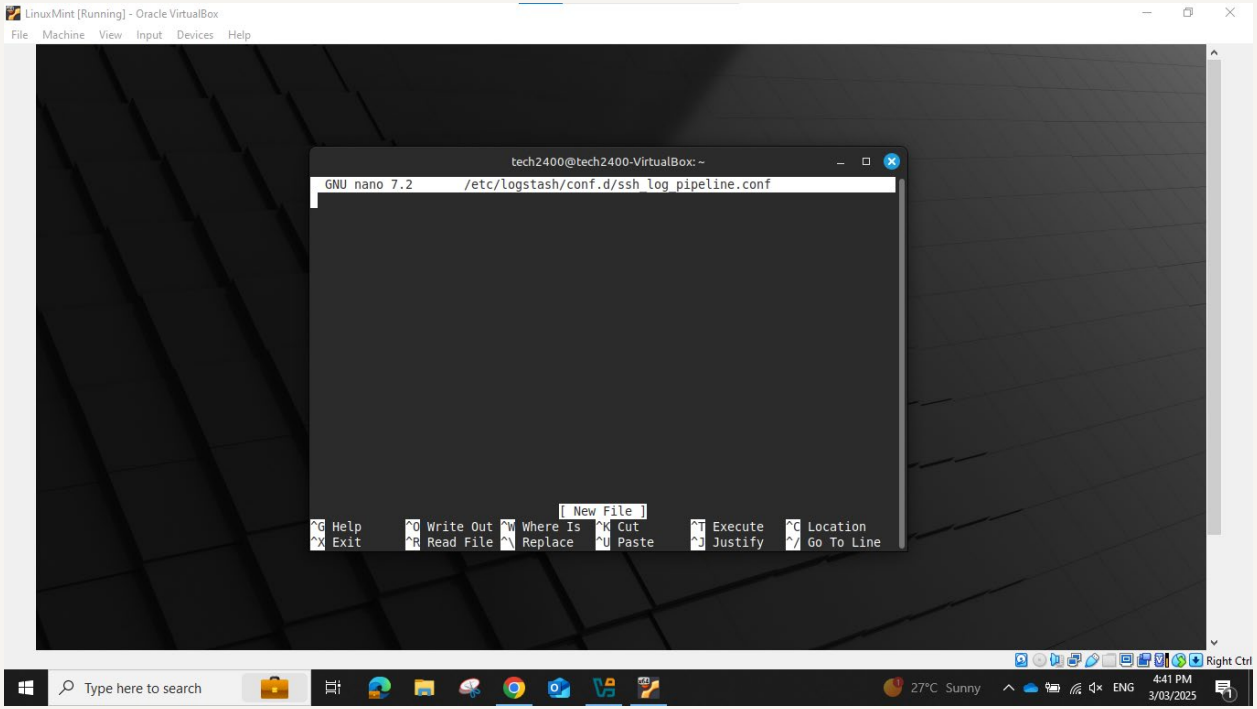
Indicates Filebeat is running



Activity: Ingesting Logs into ELK

Step 4: Configure Logstash to parse SSH logs

- Edit the Logstash pipeline configuration:
- sudo nano /etc/logstash/conf.d/ssh_log_pipeline.conf



Activity: Ingesting Logs into ELK

Step 4: Configure Logstash to parse SSH logs

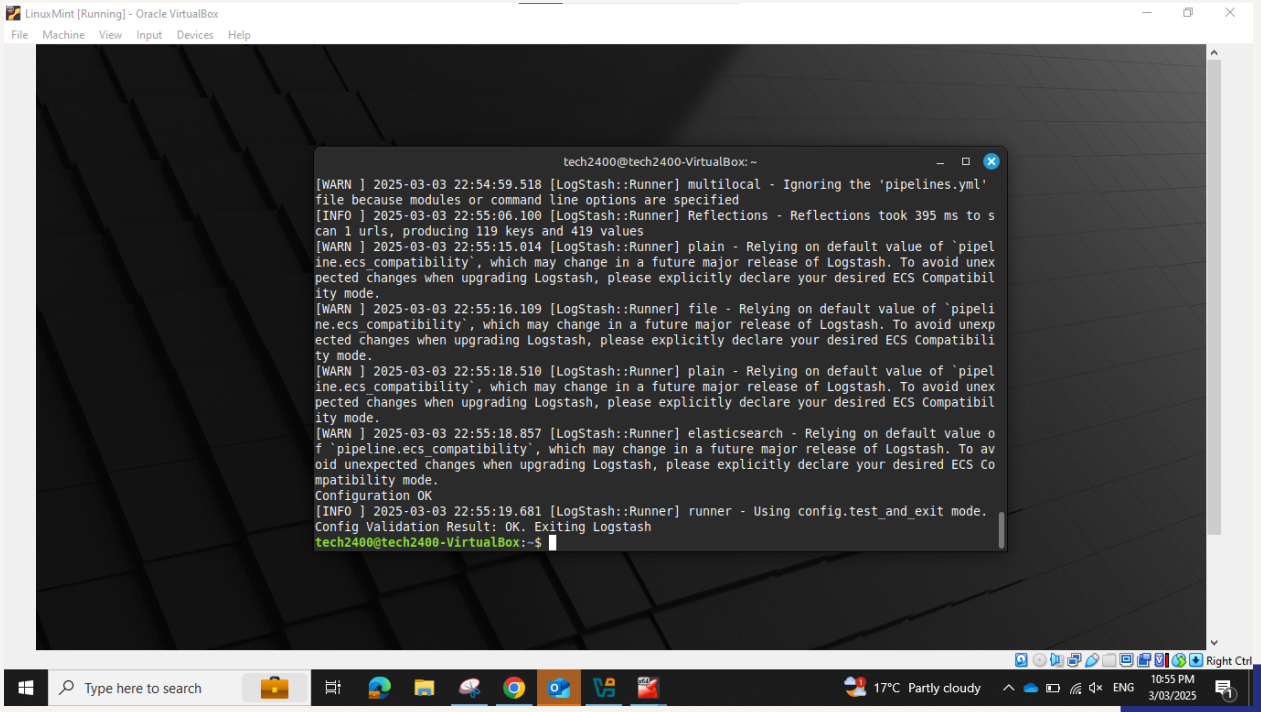
- Copy the input, filter, and output configuration from the screenshot
- Or, use generative AI to generate the configuration
- Copy and then paste (Ctrl+Shift+V) the configuration to enable SSH log collection.
- Save (Ctrl+O)
- Enter to accept file name
- Close (Ctrl+X)

```
input {
  file {
    path => "/var/log/auth.log"  # Path to the SSH log file
    start_position => "beginning"
    sincedb_path => "/dev/null"  # This ensures it reads the
      file from the start every time
  }
}

filter {
  # Optional: Parse logs (you can use grok for more complex
    parsing)
  grok {
    match => { "message" => "%{SYSLOGTIMESTAMP:timestamp}
      %{HOSTNAME:hostname} %{DATA:program}[%{NUMBER:pid}]:
      %{GREEDYDATA:message}" }
  }

  # Optional: Add a date filter for parsing the timestamp
    properly
  date {
    match => ["timestamp", "MMM dd HH:mm:ss", "ISO8601"]
  }

  # Optional: You can use mutate to clean or modify fields as
    needed
  mutate {
    add_field => { "log_type" => "ssh" }
  }
}

output {
  elasticsearch {
    hosts => ["http://localhost:9200"]  # Elasticsearch URL
    index => "ssh_logs-%{+YYYY.MM.dd}"  # Index name pattern
      with date
    user => "elastic"  # Elasticsearch username (if required)
    password => "changeme"  # Elasticsearch password (if
      required)
  }

  stdout {
    codec => rubydebug  # To print to the console for debugging
  }
}
```

Activity: Ingesting Logs into ELK

Step 4: Configure Logstash to parse SSH logs

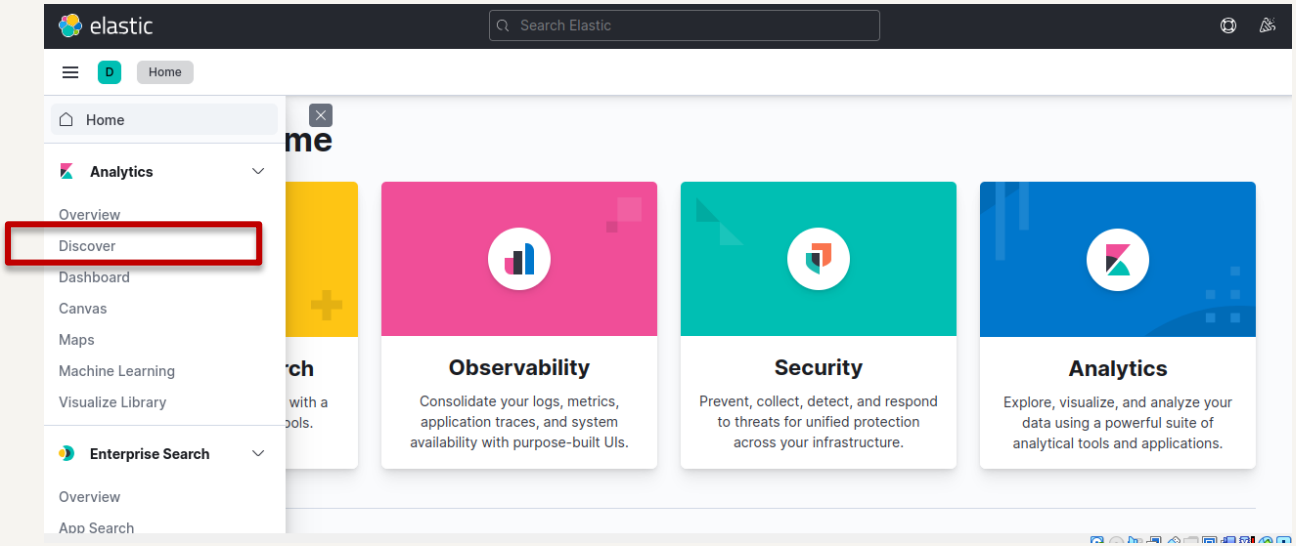
- Test the Logstash configuration
- sudo logstash –f /etc/logstash/conf.d/ssh_logs.conf --config.test_and_exit



Activity: Exploring the Logs in Kibana

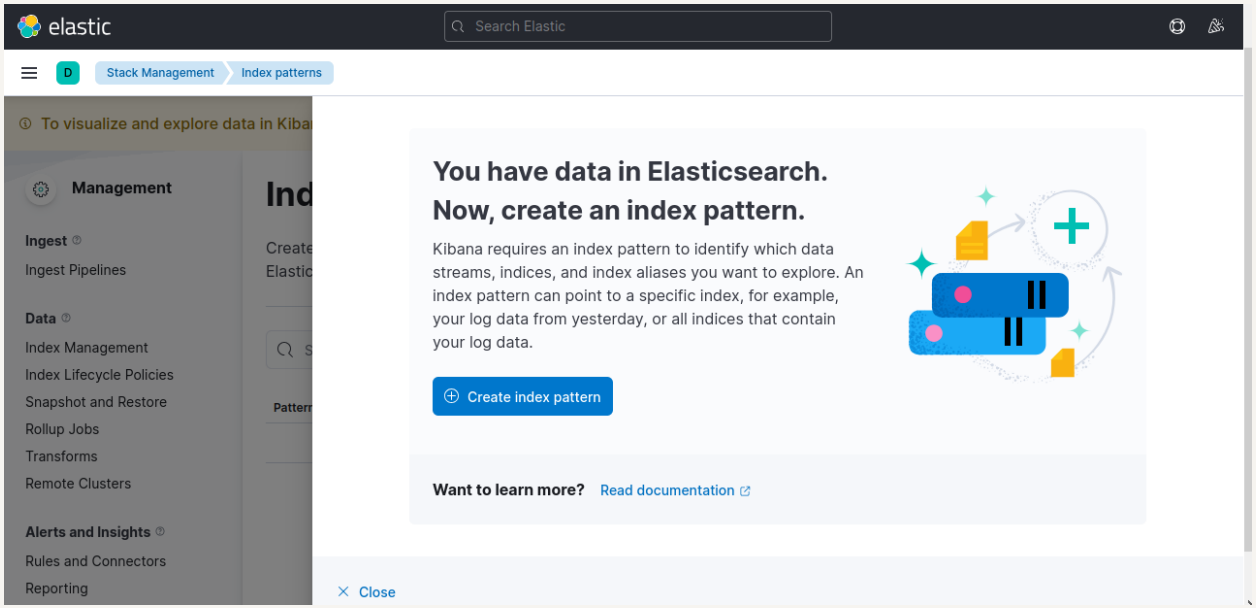
Goal: To search and analyse logs related to failed SSH login attempts.

- Open Kibana (http://localhost:5601) in a browser
- Navigate to Discover



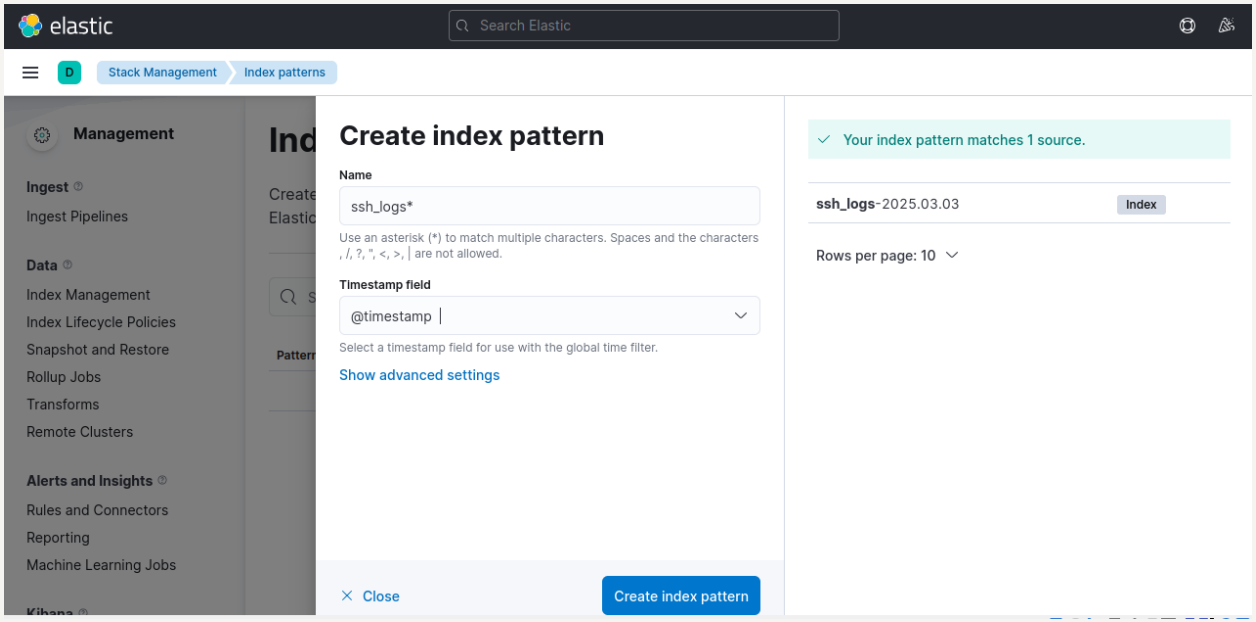
Activity: Exploring the Logs in Kibana

- Click on Create index pattern



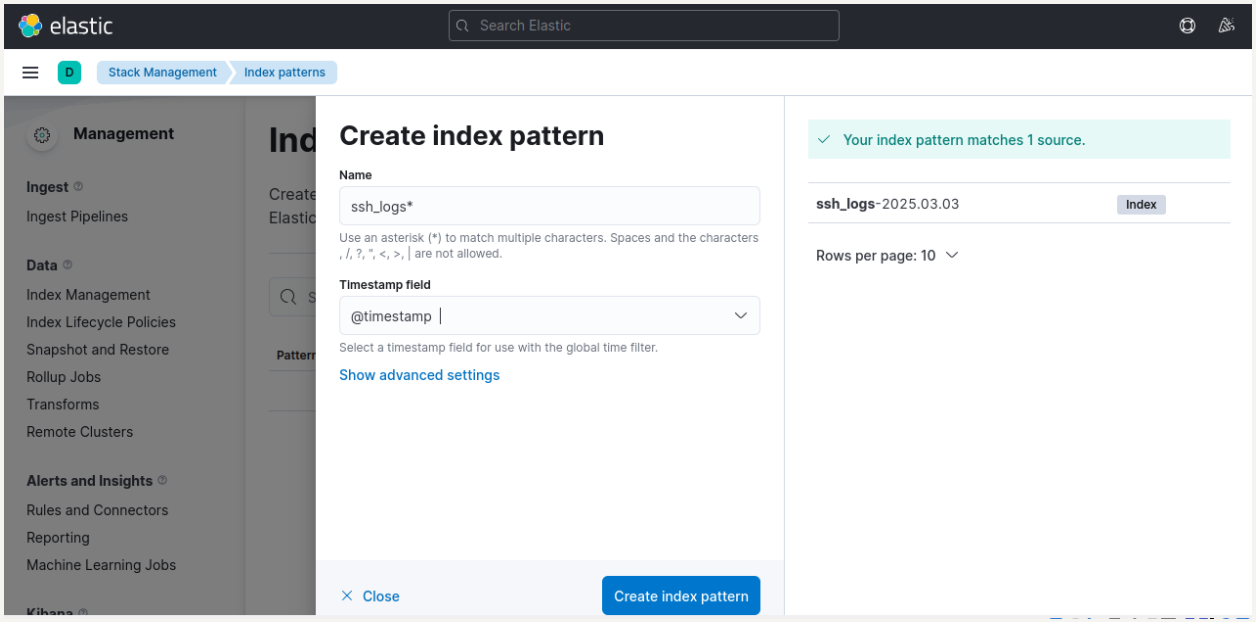
Activity: Exploring the Logs in Kibana

- Under Name, enter: ssh_logs*
- In the Timestamp field, select: @timestamp
- Click Create index pattern



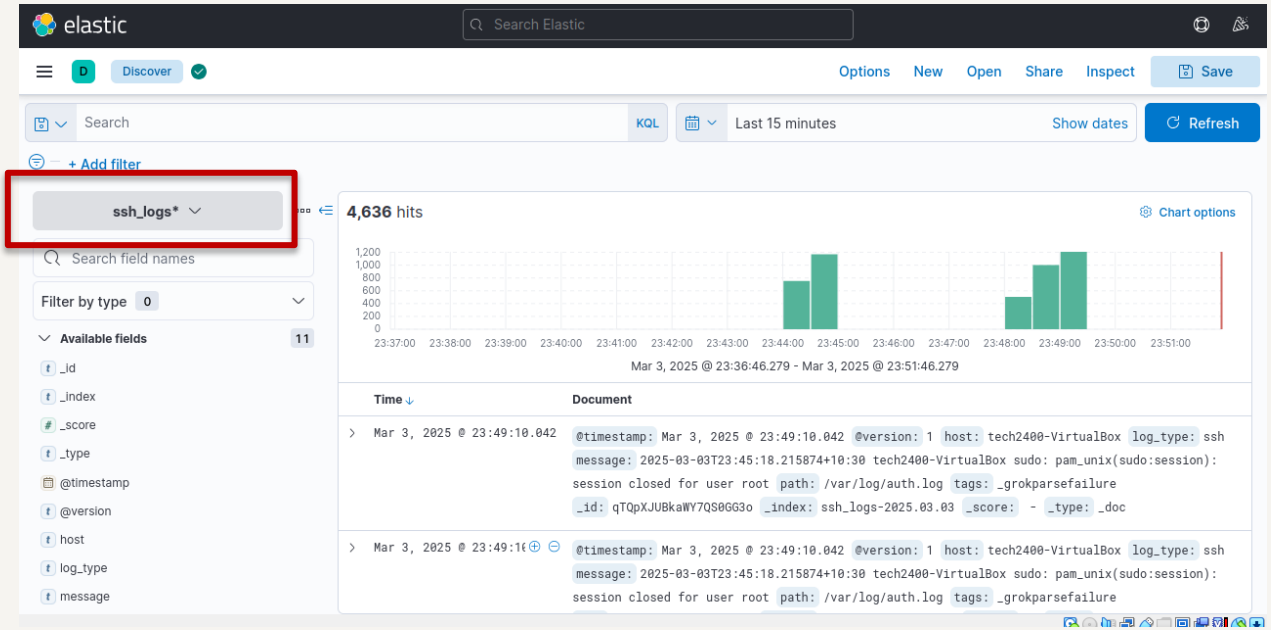
Activity: Exploring the Logs in Kibana

- Under Name, enter: ssh_logs*
- In the Timestamp field, select: @timestamp
- Click Create index pattern



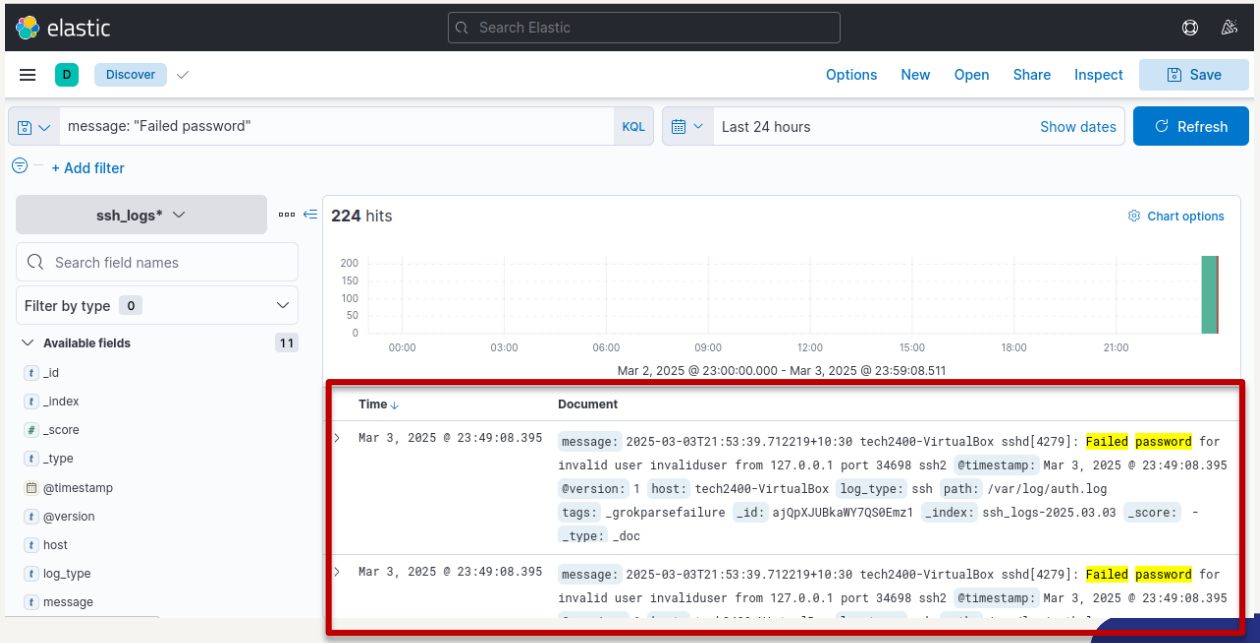
Activity: Exploring the Logs in Kibana

- Navigate back to Discover
- Make sure "ssh_logs-*" is selected from the dropdown



Activity: Exploring the Logs in Kibana

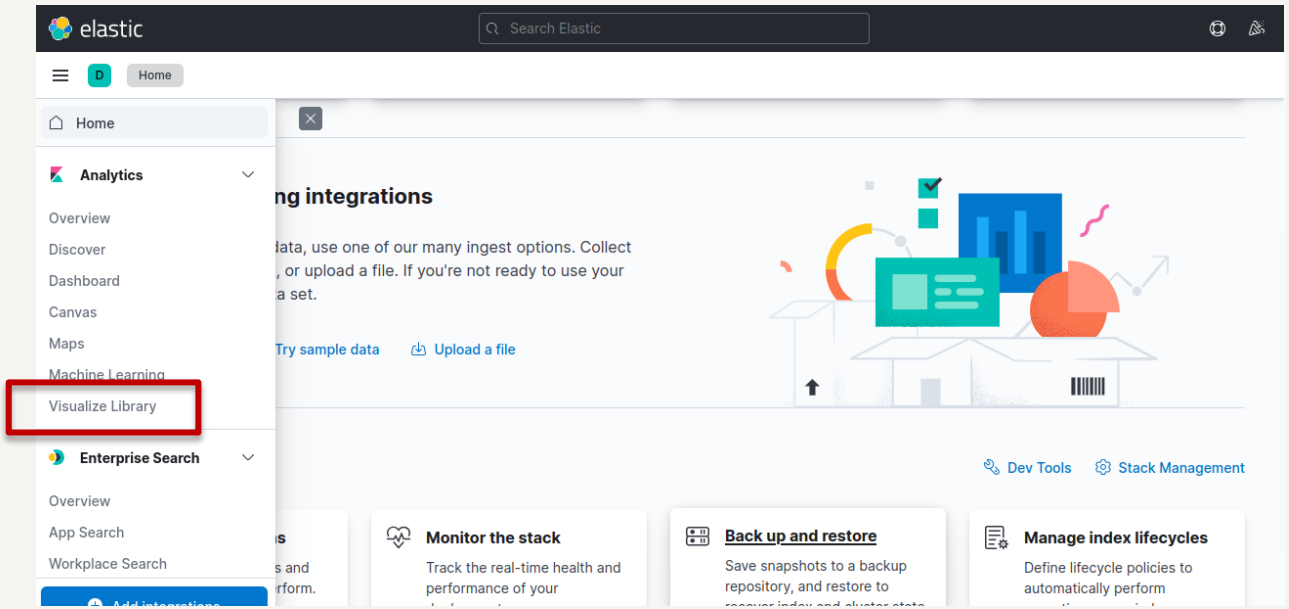
- Broaden the time range (e.g., Last 24 hours)
- Search for failed SSH attempts using
- message: "Failed password"
- Check on the logs to identify patterns such as frequent failed attempts from the same IP.



Activity:
Creating a Visualisation in Kibana

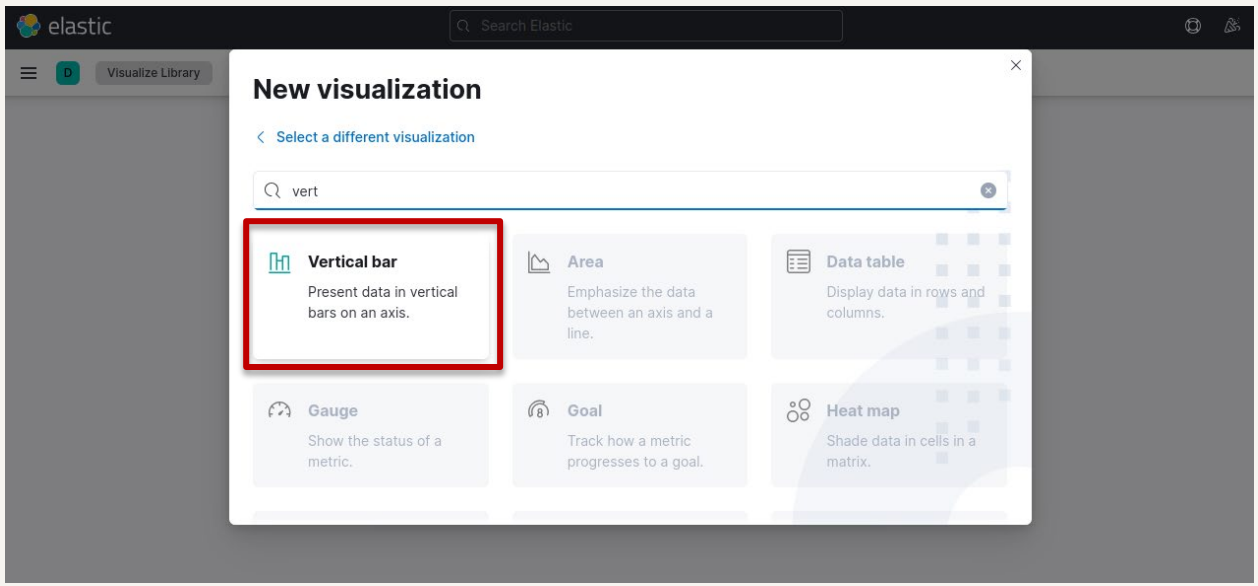
Goal: To visually analyse failed login attempts and detect brute force patterns.

- Navigate to Visualize Library



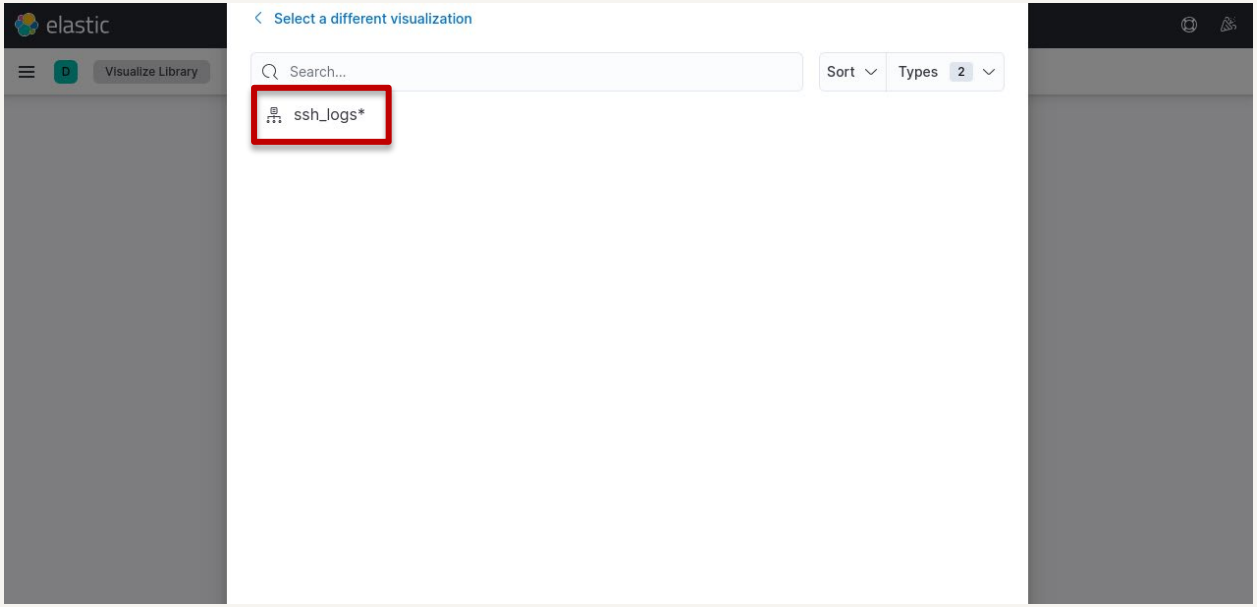
Activity:
Creating a Visualisation in Kibana

- Click Create new visualization > Aggregation based
- Filter: Vertical Bar



Activity:
Creating a Visualisation in Kibana

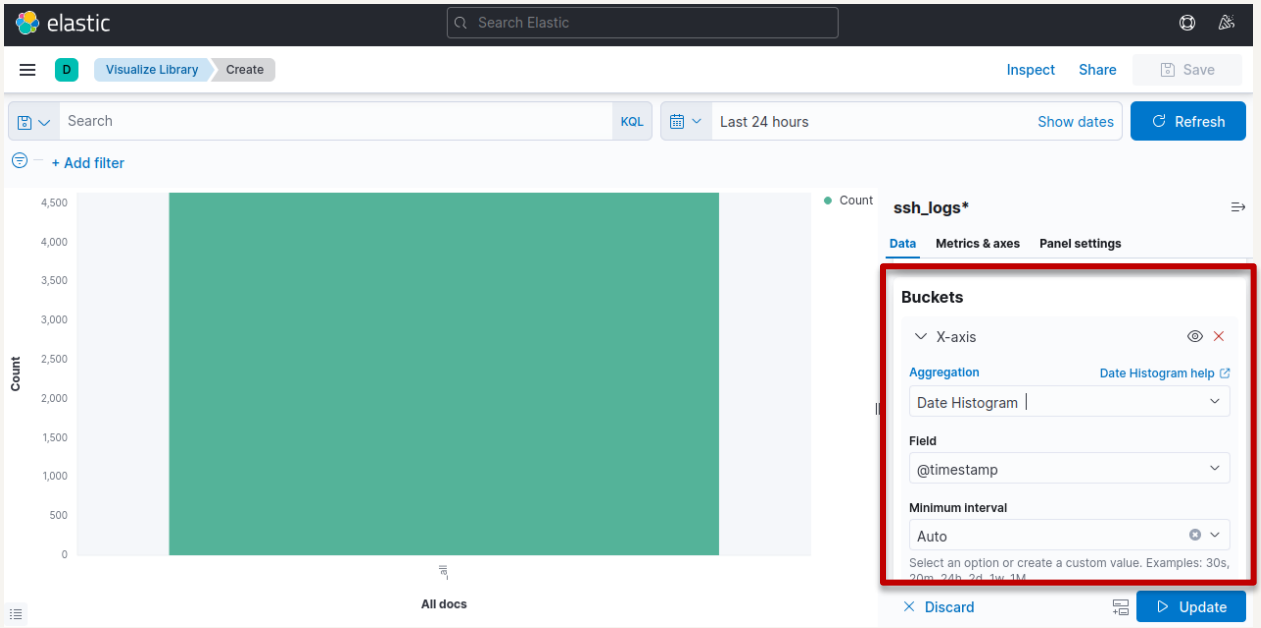
- Select ssh_logs* as the data source



Activity:
Creating a Visualisation in Kibana

- Under Buckets, click Add
- Select X-axis
- Under Aggregation, select Date Histogram
- Set Field to @timestamp and Minimum interval to Auto
- Click Update

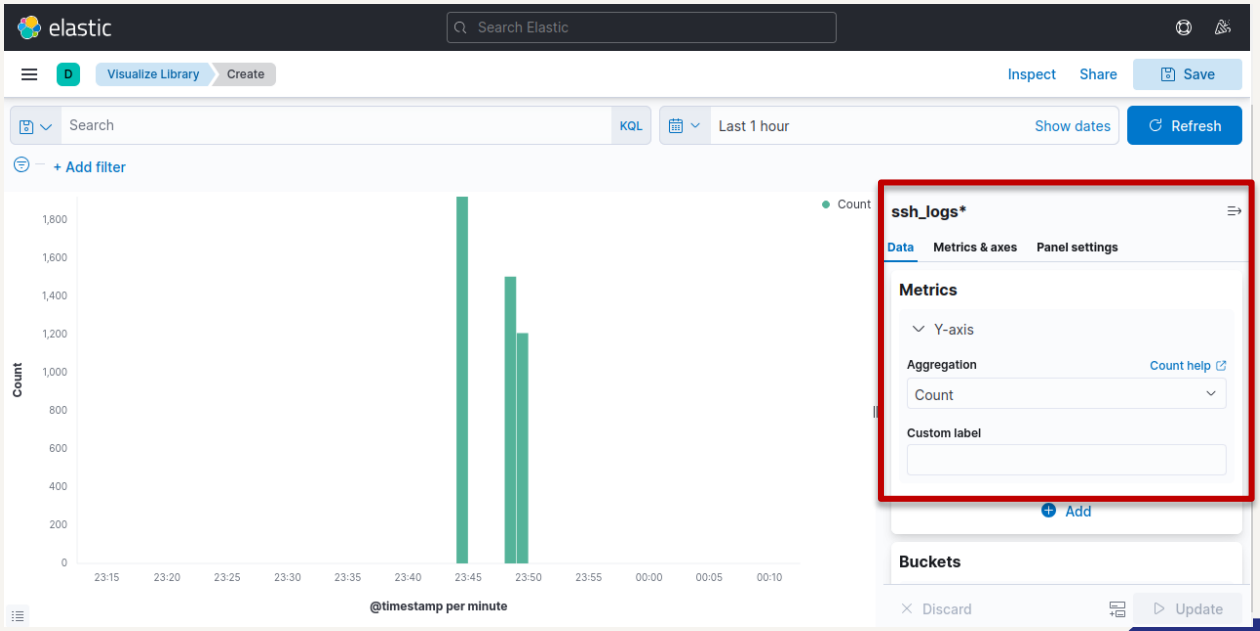
Note: By default, Y-Axis is already set to count



Activity:
Creating a Visualisation in Kibana

- The bar chart has been updated

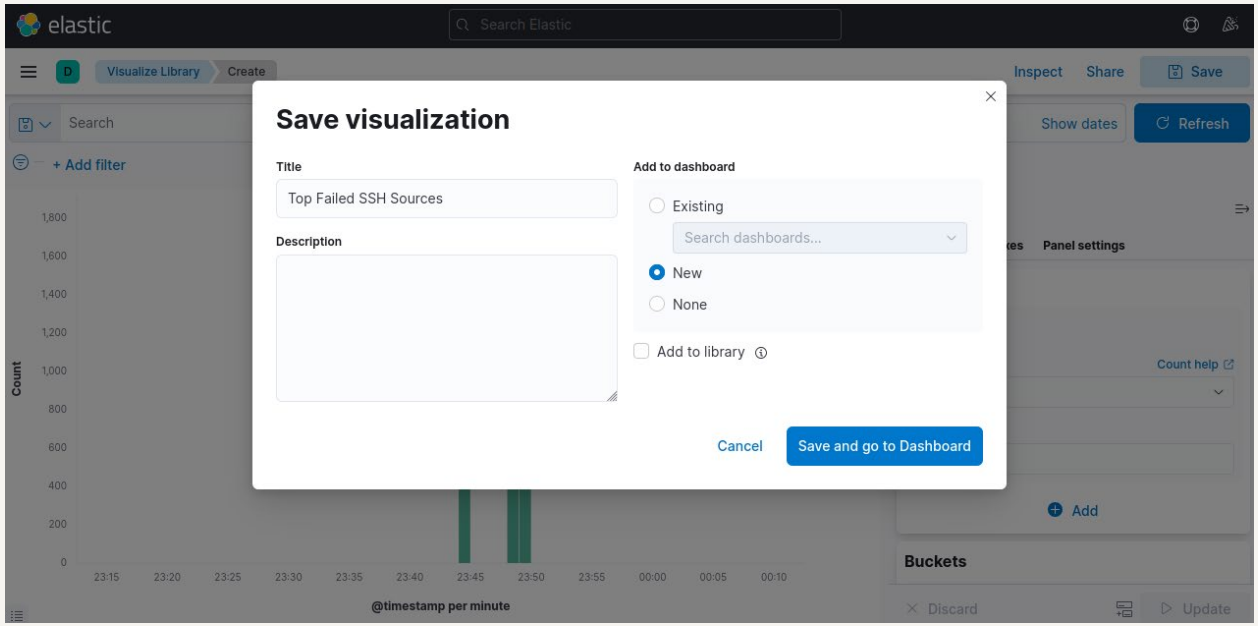
Note: By default, Y-Axis is already set to count, which counts the number of logs



KAPLAN BUSINESS SCHOOL AUSTRALIA

Activity:
Creating a Visualisation in Kibana

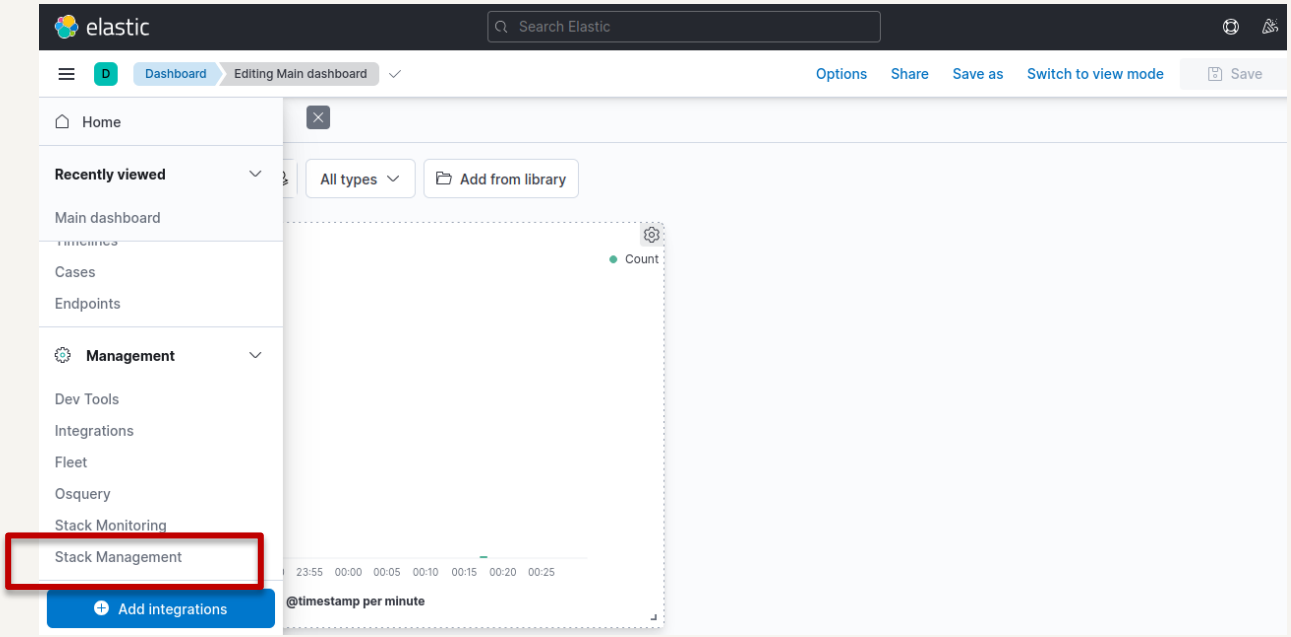
- Click on the Save button at the top right corner
- Save the visualisation and select the dashboard to which it will be saved (in this case, select New)



Activity: Setting up an Alert

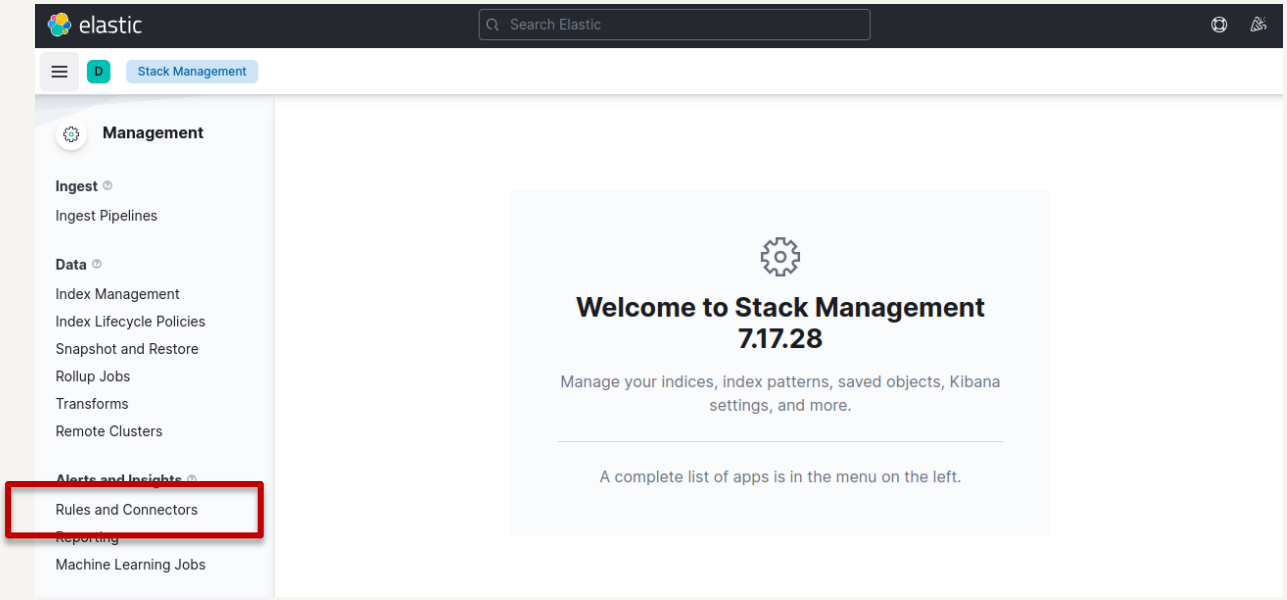
Goal: To get notified when a potential brute-force attack occurs.

- In Kibana, go to Stack Management



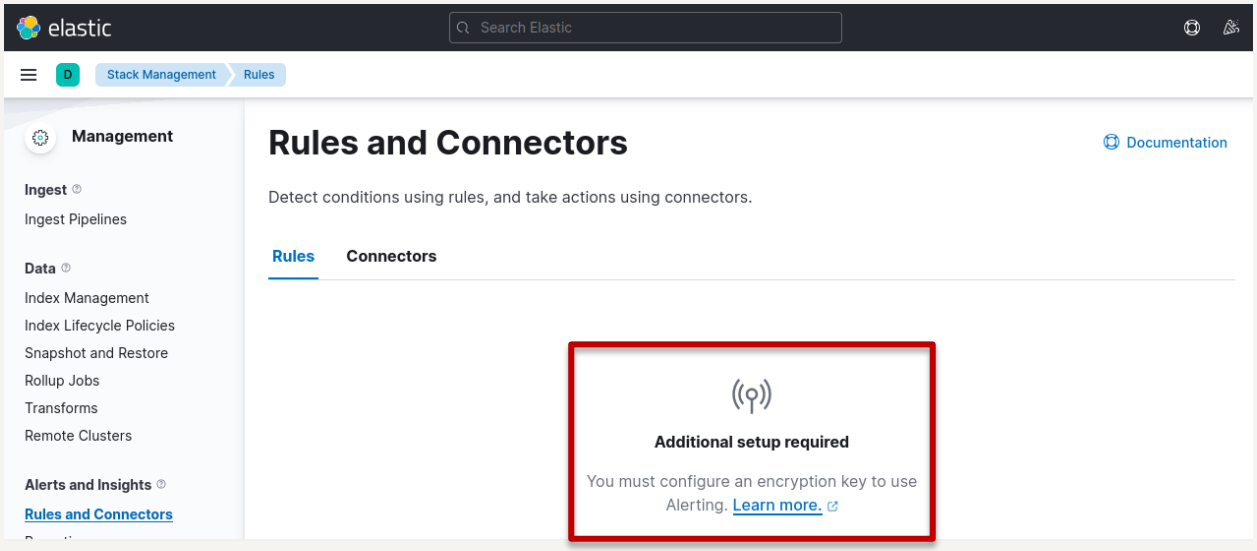
Activity: Setting up an Alert

- Navigate to Rules and Connectors



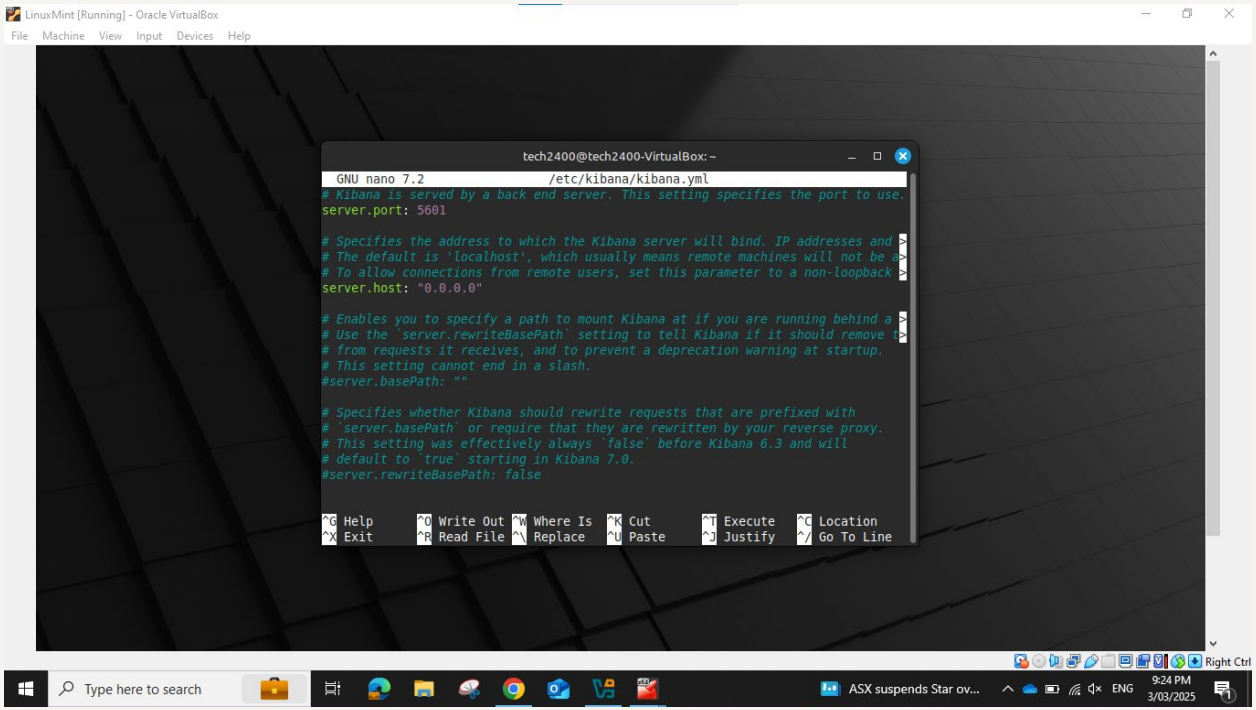
Activity: Setting up an Alert

- Additional setup will be required because Kibana needs to securely handle sensitive information when triggering alerts



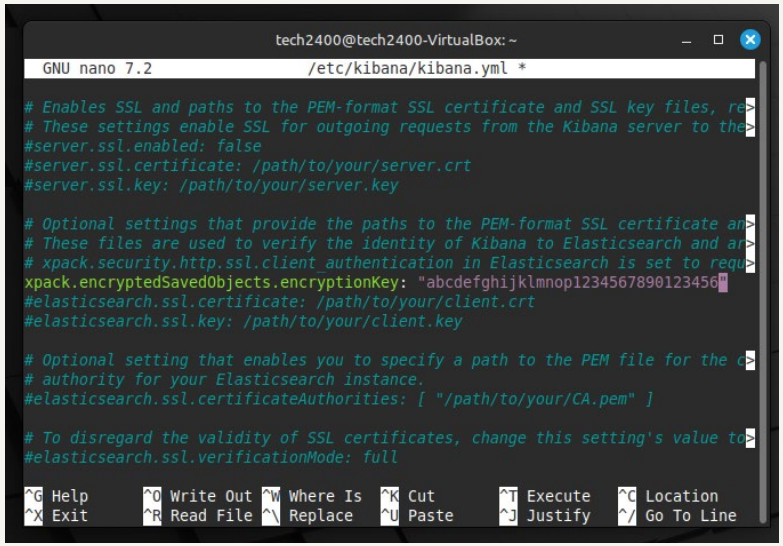
Activity: Setting up an Alert

- Close your browser and open a Terminal
- Run sudo nano /etc/kibana/kibana.yml



Activity: Setting up an Alert

- Add the Encryption Key setting (anywhere in the file):
- Xpack.encryptedSavedObjects.encryptionKey: "your_encryption_key"
- Save (Ctrl+O), Enter to accept filename, Close (Ctrl+X)

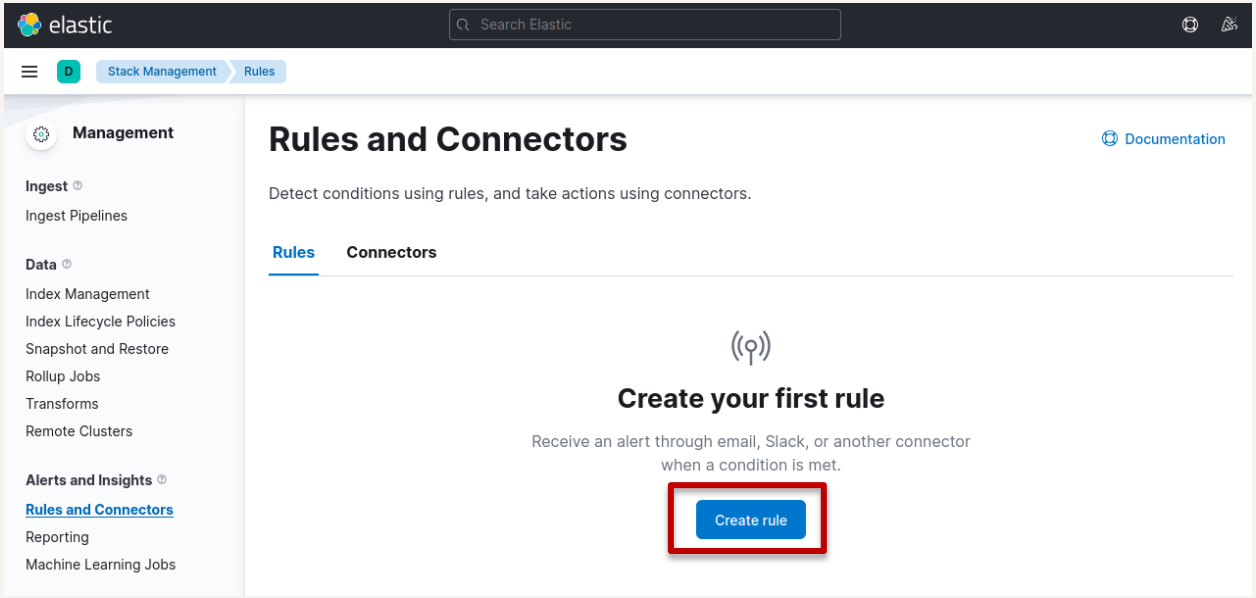


Note: A strong random string of at least 32 characters is recommended.

- Restart Kibana: sudo systemctl restart kibana

Activity: Setting up an Alert

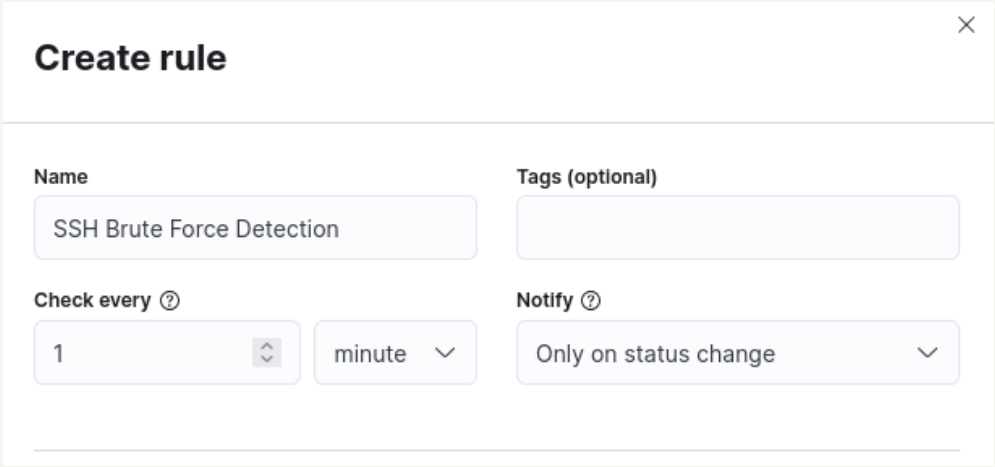
- Open Kibana on a browser (http://localhost:5601)
- Navigate back to Stack Management > Rules and Connectors
- Click on Create Rule



Activity: Setting up an Alert

Configure your rule:

- Name: SSH Brute Force Detection
- Check every: 1 minute
- Notify: Only on status change



Activity: Setting up an Alert

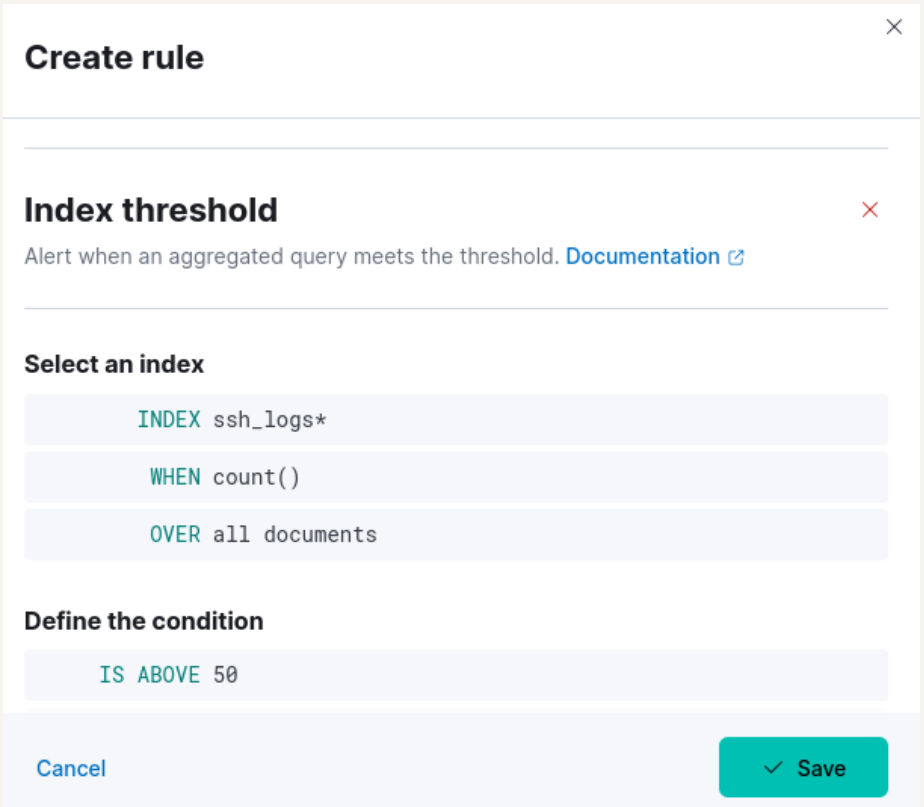
Configure your rule:

- Rule type: Index threshold

INDEX ssh_logs*
WHEN count()
OVER all documents

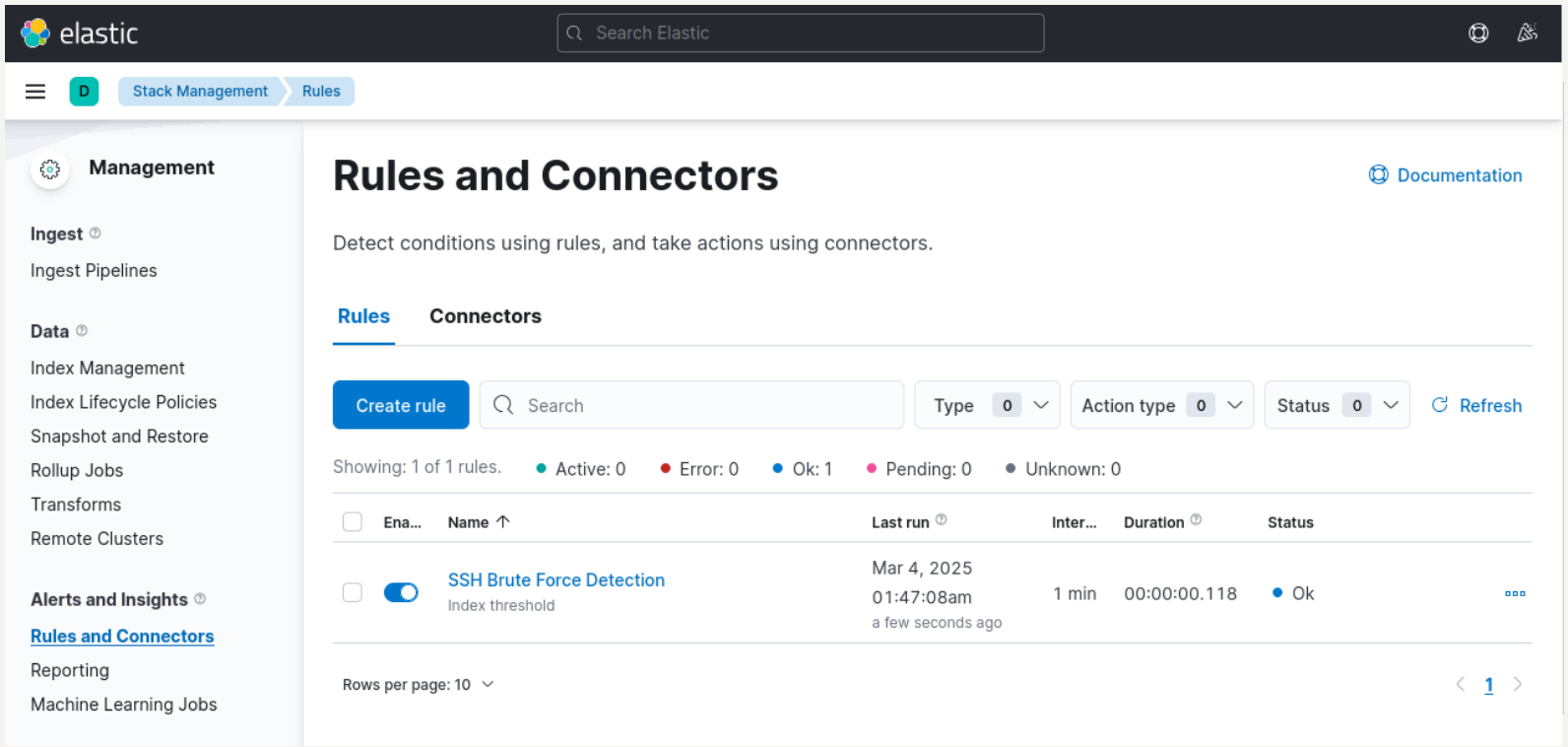
IS ABOVE 50

- Click Save



Activity: Setting up an Alert

- Access your rule through Stack Management > Rules and Connectors



Next Week

Week 10: Incident Response & Management (Part 2)

- Advanced Incident Response
- Basic Forensic Analysis Concepts
- Post-Incident Response & Lessons Learned
- Introduction to Snort on pfSense