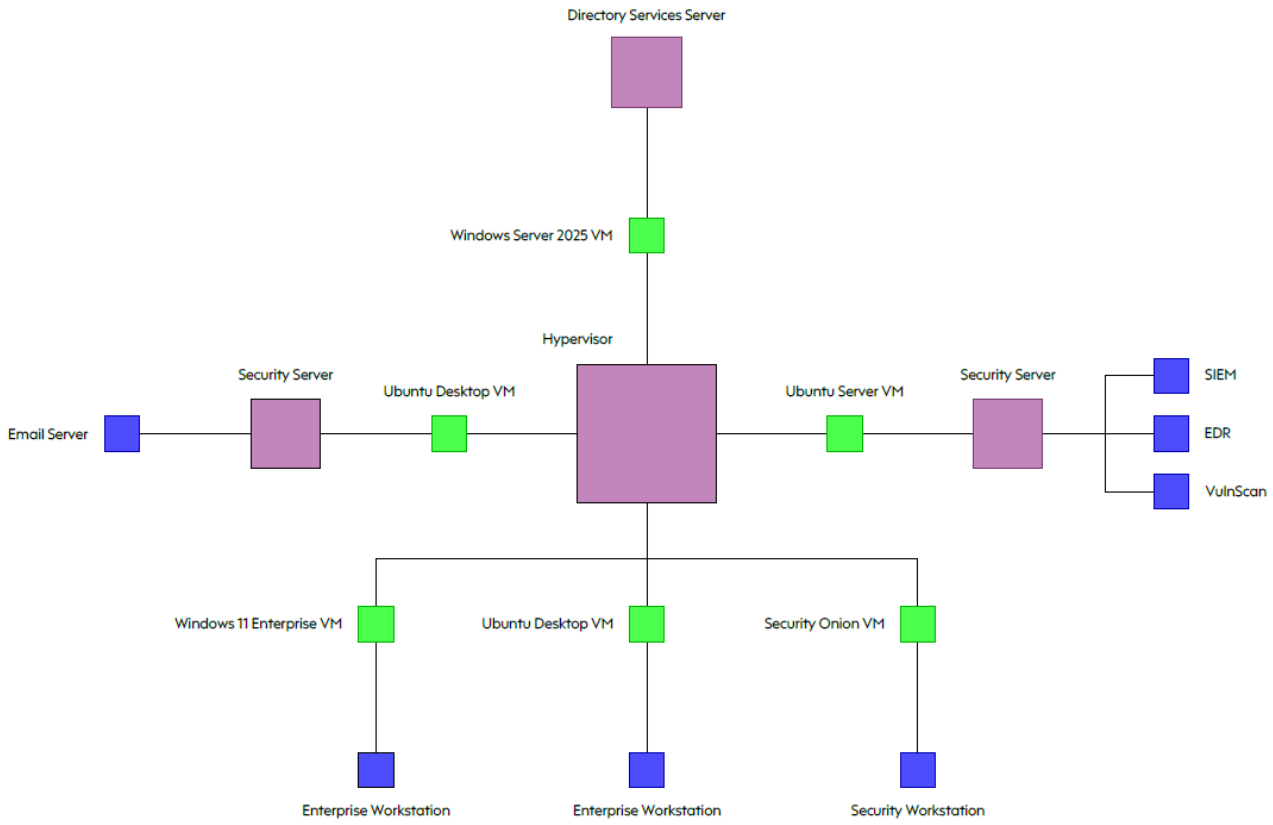
Cybersecurity-Homelab-Building-The-Environment Documentation:

This guide provides step-by-step instructions with screenshots for key steps. Optional explanations are highlighted in red for clarity and can be skipped.

Below is the network architecture for the small business lab:



Element	*Purpose*	*Software/OS*	*Specs*	*Storage (minimum)*
Hypervisor.	*Runs and manages multiple VMs*	*VirtualBox on Windows 11 host.*	*-*	
Enterprise Workstation 1	*Employee workstation simulation*	*Windows 11 Enterprise VM*	*2 CPU / 4096 MB*	*80 GBs*
Enterprise Workstation 2	*Linux workstation simulation (developer/employee)*	*Ubuntu 22.04 Desktop VM*	*1 CPU / 2048 MB*	*80 GBs*
Security Workstation	*Security analysis and monitoring*	*Security Onion VM*	*1 CPU / 2048 MB*	*55 GBs*
Security Server 1	*Email environment for*	*Ubuntu 22.04 Desktop VM*	*2 CPU / 4096 MB*	*80 GBs*

	phishing exercises (MailHog/Postfix)			
Security Server 2	*Central SIEM/EDR server, log collection, analysis, vulnerability scanning*	*Ubuntu Server 22.04 with Wazuh*	*1 CPU / 2048 MB*	*25 GBs*
Directory Services Server	*Central identity & network management (AD, DNS, DHCP, SSO)*	*Windows Server 2025*	*2 CPU / 4096 MB*	*50 GBs*

Step 1: Download ISOs For VMs:

Install the operating system ISOs by downloading them from the links provided below:

- Windows Server 2025: https://www.microsoft.com/en-us/evalcenter/evaluate-windows-server-2025
- Windows 11 Enterprise: https://www.microsoft.com/en-us/evalcenter/evaluate-windows-11-enterprise
- Ubuntu Desktop 22.04.5 LTS: https://releases.ubuntu.com/jammy/ubuntu-22.04.5-desktop-amd64.iso
- Ubuntu Server 22.04.5 LTS: https://releases.ubuntu.com/jammy/ubuntu-22.04.5-server-amd64.iso
- Security Onion: https://github.com/Security-Onion-Solutions/securityonion/releases

These ISO files contain the operating systems designated for installation on the virtual machines within the business network.

Step 2: Enable Virtualisation On PC:

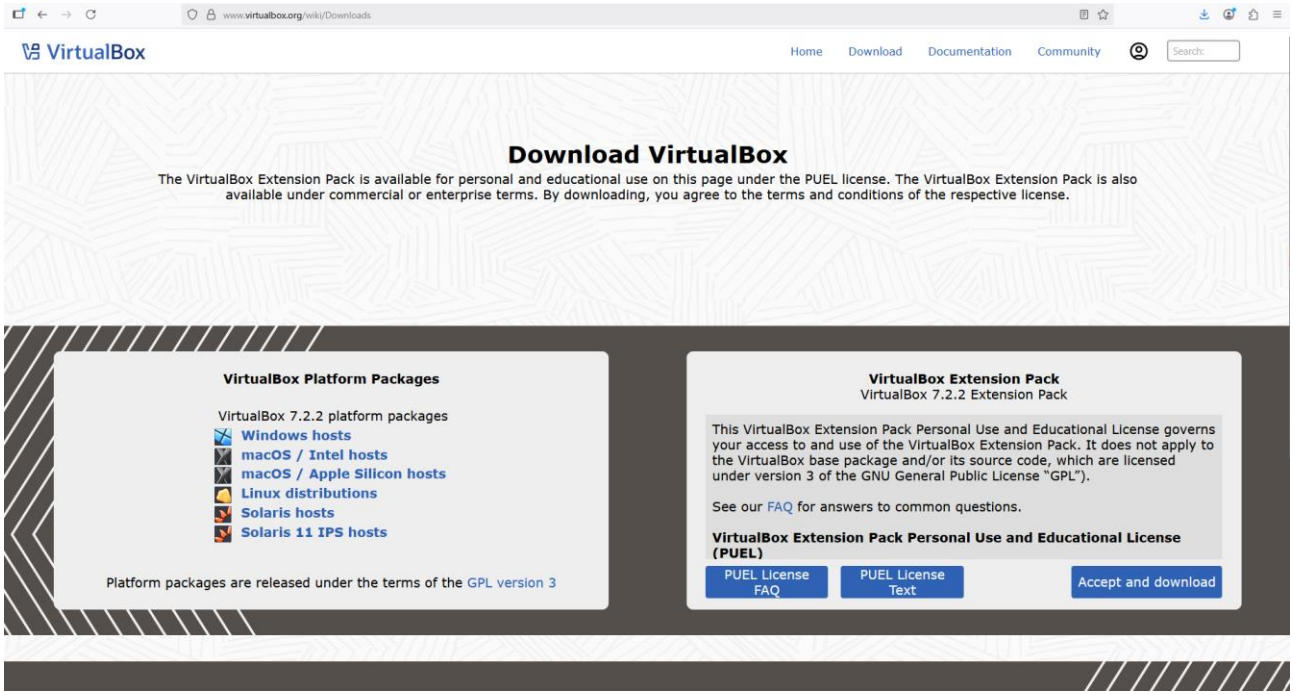
1. Restart your PC and enter the BIOS/UEFI using the manufacturer-specific hotkey.
2. Locate and enable Virtualization Technology (Intel VT-x / AMD-V).
3. Save changes and reboot.

Virtualisation must be enabled to run virtual machines.

Step 3: Install Virtual Box:

1. Download VirtualBox for host OS: https://www.virtualbox.org/wiki/Downloads
2. Windows hosts: Ensure Microsoft Visual C++ 2019 Redistributable is installed: https://learn.microsoft.com/en-us/cpp/windows/latest-supported-vc-redist?view=msvc-170
3. Install VirtualBox using default settings.

VirtualBox is a type-2 hypervisor that allows you to run virtual machines on your computer.



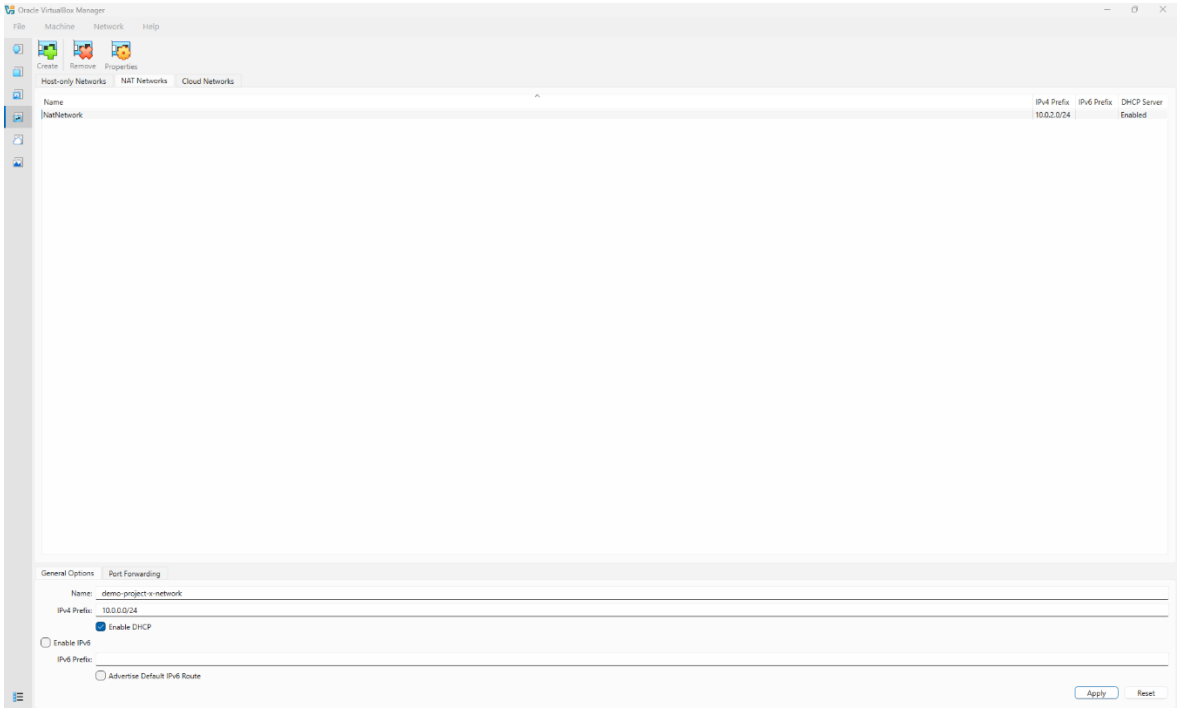
Step 4: Provision NAT Network On Vbox:

1. Switch VirtualBox preferences to Expert Mode to access the network menu.
2. Click the Network icon → NAT Networks → Add new network.
3. Name the network, set IPv4 Prefix to 10.0.0.0/24, enable DHCP, and click Apply.

NAT Network: Allows VMs internet access while isolating them from the host network. Compromised VMs cannot reach your real network.

IPv4 Prefix /24: Provides 254 usable IPs (10.0.0.1–10.0.0.254) for lab VMs.

DHCP: Automatically assigns IPs to VMs, avoiding manual configuration.



Step 5: Provisioning Windows Server 2025 VM (for Directory Services Server):

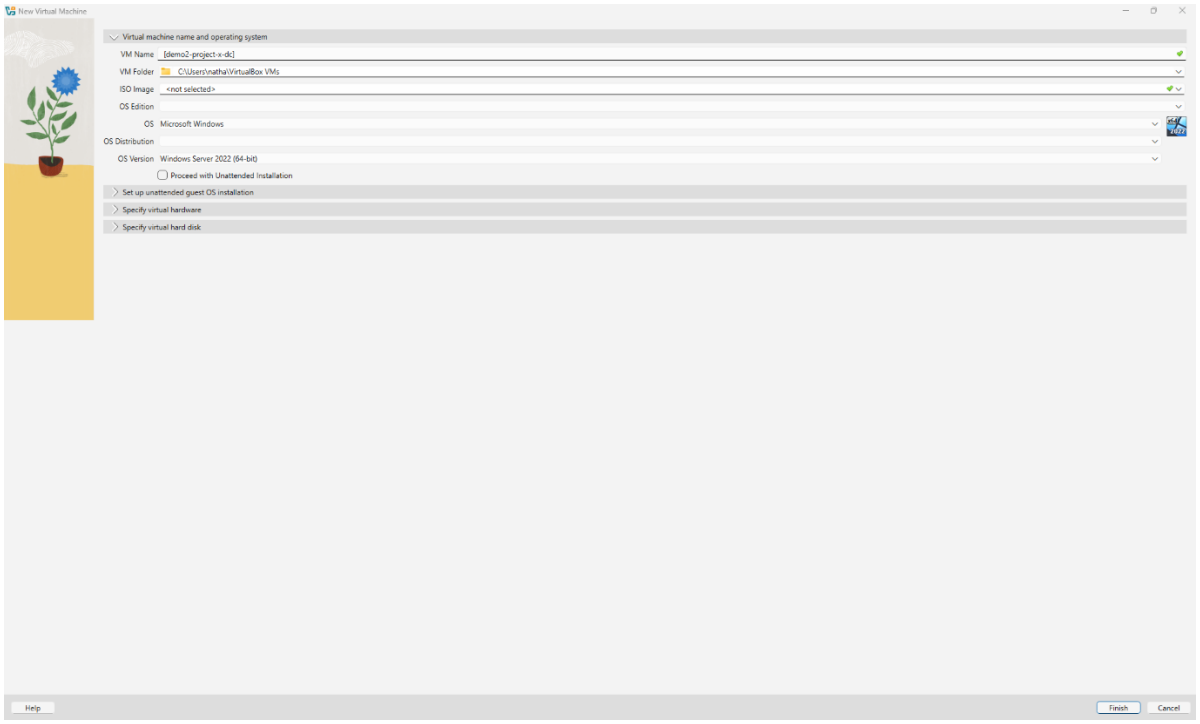
1. Create VM in VirtualBox:

- Name VM, Type: Microsoft Windows, Version: Windows 2022 (64-bit).

- Memory: 4096 MB, CPUs: 2, Hard Disk: 50 GB.

- Assign VM to the NAT network created earlier.

Set up a virtual machine environment with sufficient resources to run Windows Server 2025 and lab services reliably.



2. Mount ISO & Install OS:

- Boot from Windows Server 2025 ISO.

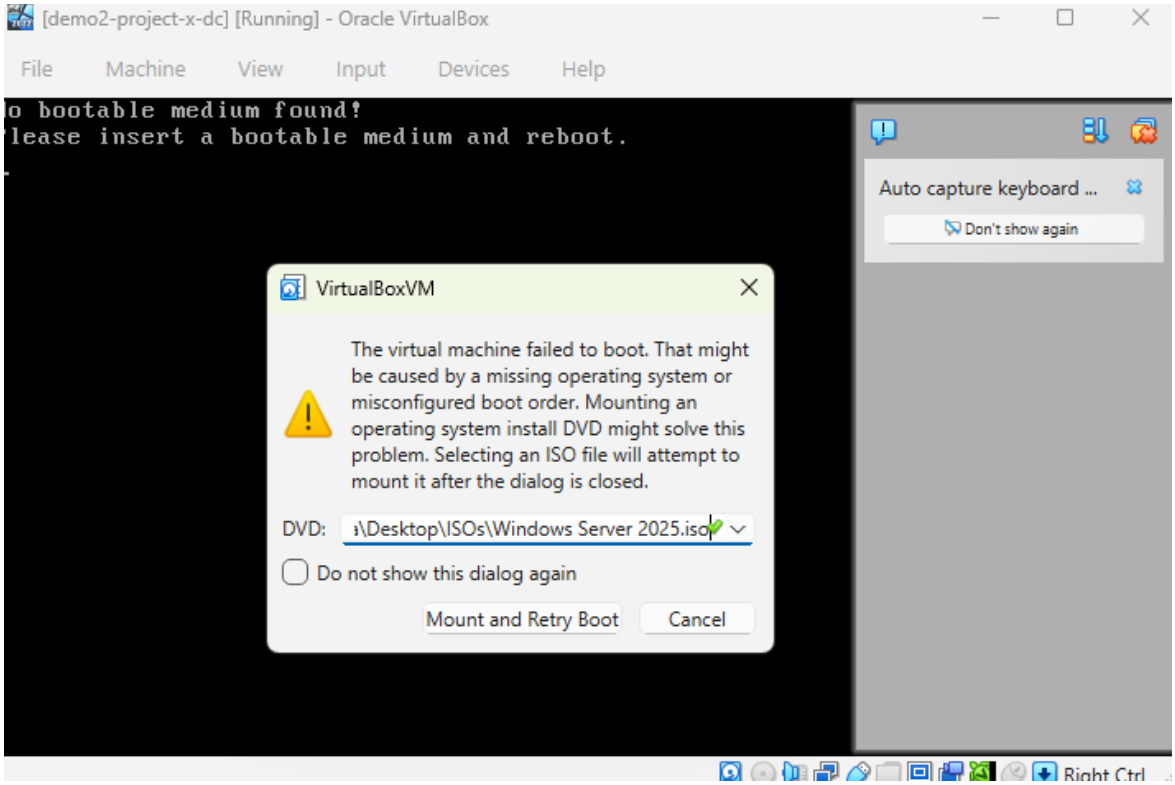
- Select language and region.

- Install Windows Server 2025 Standard Evaluation (Desktop Experience).

- Create a new partition (default settings) for system stability.

- Set Administrator password.

Install the server OS to provide a base system for Active Directory, DHCP, DNS, and other lab services. Creating a partition separates system files from the OS for stability and recovery.



3. Convenience & Usability Configurations

- Disable auto logoff: Settings → Accounts → Sign-in options → “Require sign-in” set to Never.

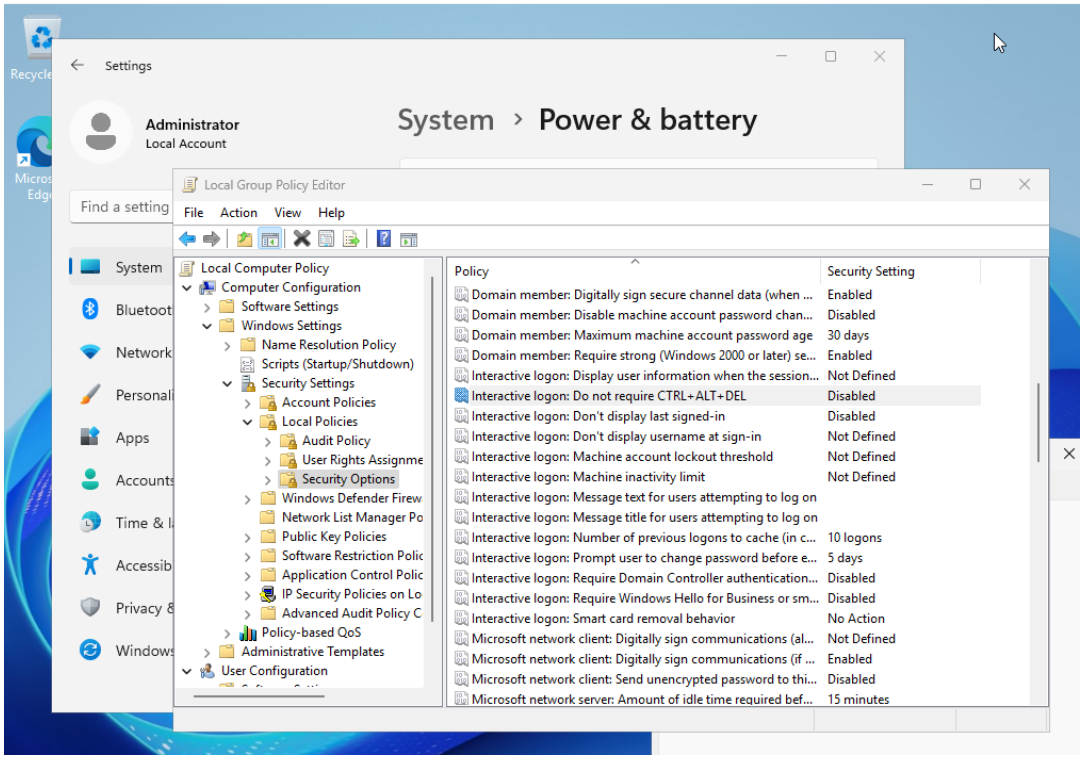
- Disable Ctrl+Alt+Del requirement:

- Win + R → gpedit.msc → Computer Configuration → Windows Settings → Security Settings → Local Policies → Security Options → “Interactive logon: Do not require Ctrl+Alt+Del” → Enabled.

- Install VirtualBox Guest Additions: Devices → Insert Guest Additions CD → Run VBoxWindowsAdditions.exe → Reboot.

- Enable Full Screen, Shared Clipboard (bidirectional), Drag & Drop (bidirectional).

Disabling auto logoff and Ctrl+Alt+Del simplifies uninterrupted access and login, while installing Guest Additions enables full screen, shared clipboard, and drag-and-drop functionality for improved efficiency.

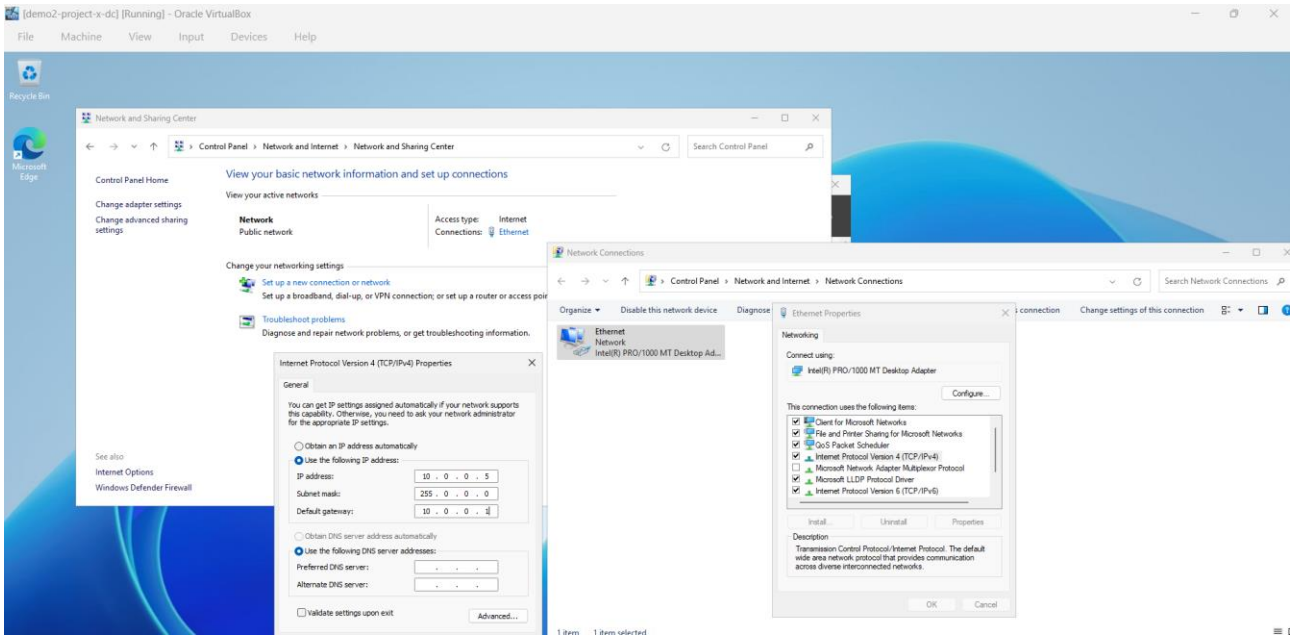


4. Assign Static IP

- Control Panel → Network & Internet → Network & Sharing Center → Change adapter settings → Ethernet → IPv4 properties.

- Set IP: 10.0.0.5, Subnet Mask: 255.255.255.0, Gateway: 10.0.0.1.

Ensures consistent network addressing so that other VMs and services can reliably locate the directory server.

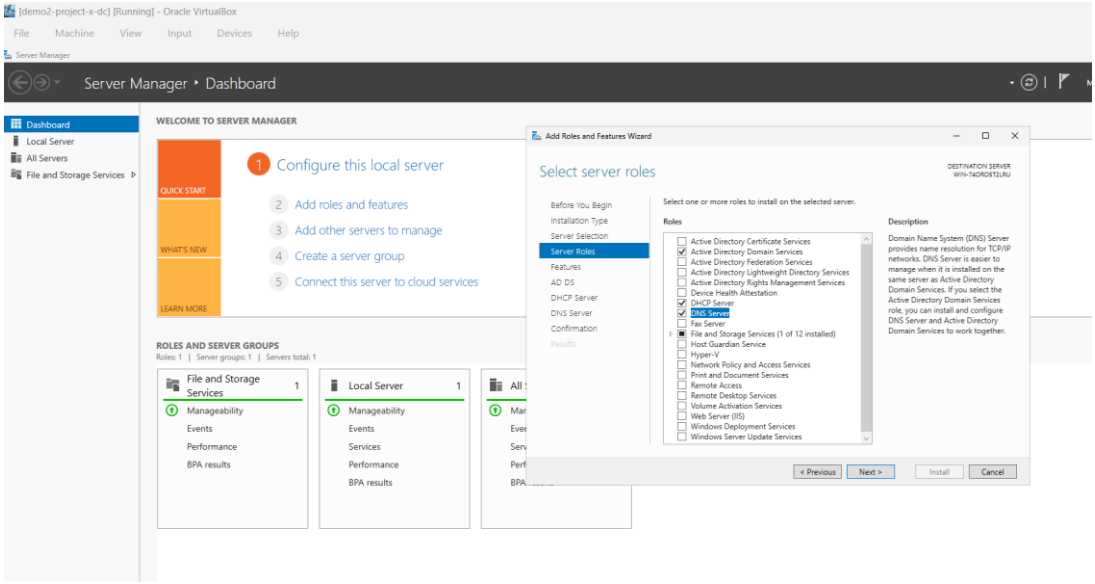


5. Install Server Roles

- Server Manager → Add Roles & Features → Role-based installation.

- Install: Active Directory Domain Services (AD DS), DHCP, DNS (all defaults).

Deploy core network services (AD DS, DHCP, DNS) needed for lab operations, user authentication, and automated IP management.



6. Promote to Domain Controller

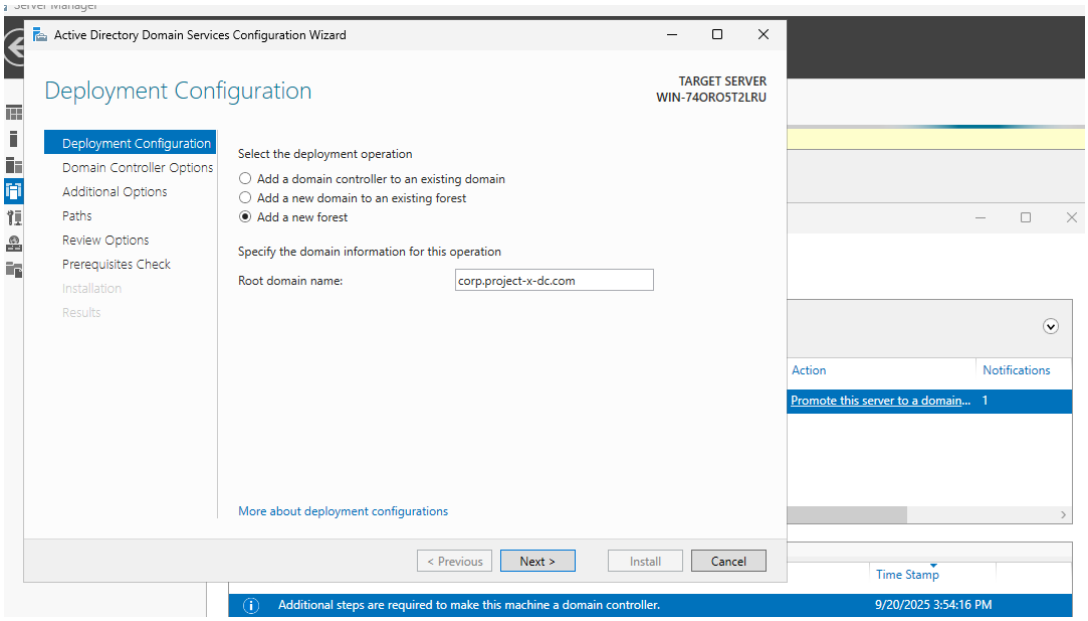
- Server Manager → AD DS → Promote server to Domain Controller.

- Create new forest with root domain name (e.g., corp.project-x-dc.com).

- Set DSRM password, leave defaults for NetBIOS, paths, DNS options.

- Complete installation and restart VM.

Enables the server to host and manage the Active Directory domain, providing centralized identity, authentication, and policy management for lab VMs.



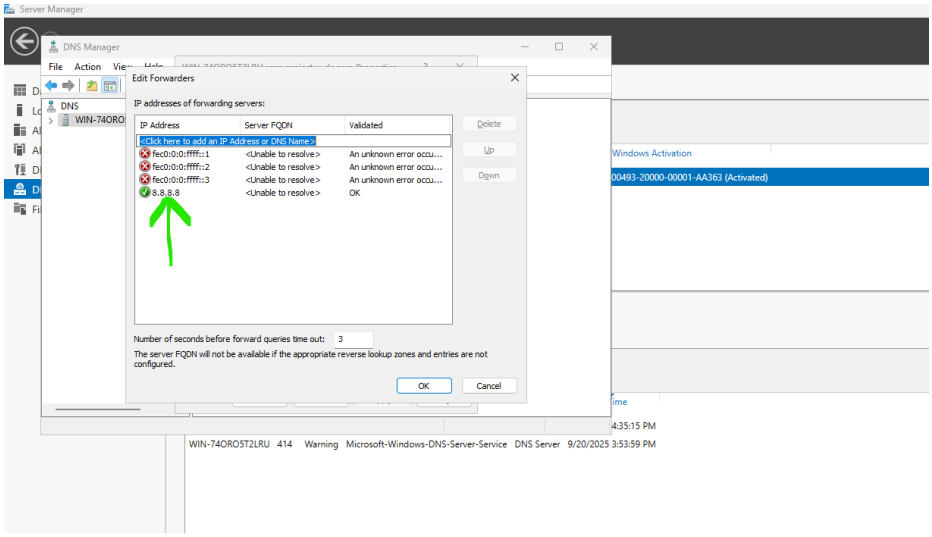
7. Configure DNS

- Server Manager → DNS → Forward Lookup Zone → Configure internal domain.

- Forwarders → Add external DNS (e.g., 8.8.8.8) for internet name resolution.

- Test connectivity: ping google.com and nslookup [domain].

DNS: Resolves internal domain names and forwards external requests.



8. Configure DHCP

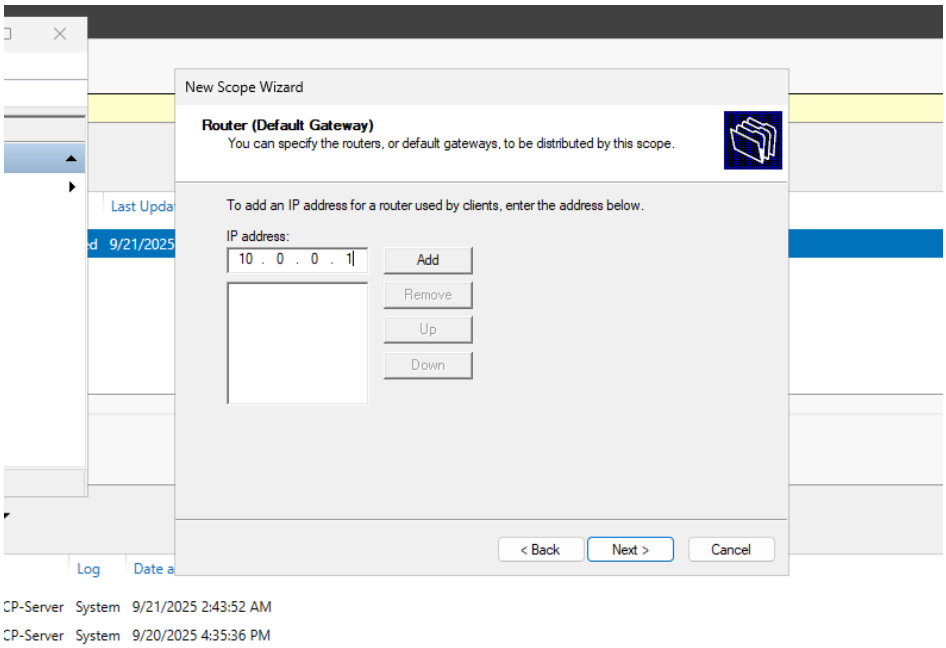
- DHCP → IPv4 → New Scope.

- Set Scope Name, IP Range: 10.0.0.100–10.0.0.200, Subnet: /24.

- Use defaults for exclusions, lease duration (8 days), and other options.

- Set default gateway: 10.0.0.1, complete DHCP configuration.

DHCP: Dynamically assigns IPs to lab VMs within a defined range, preventing conflicts with static addresses



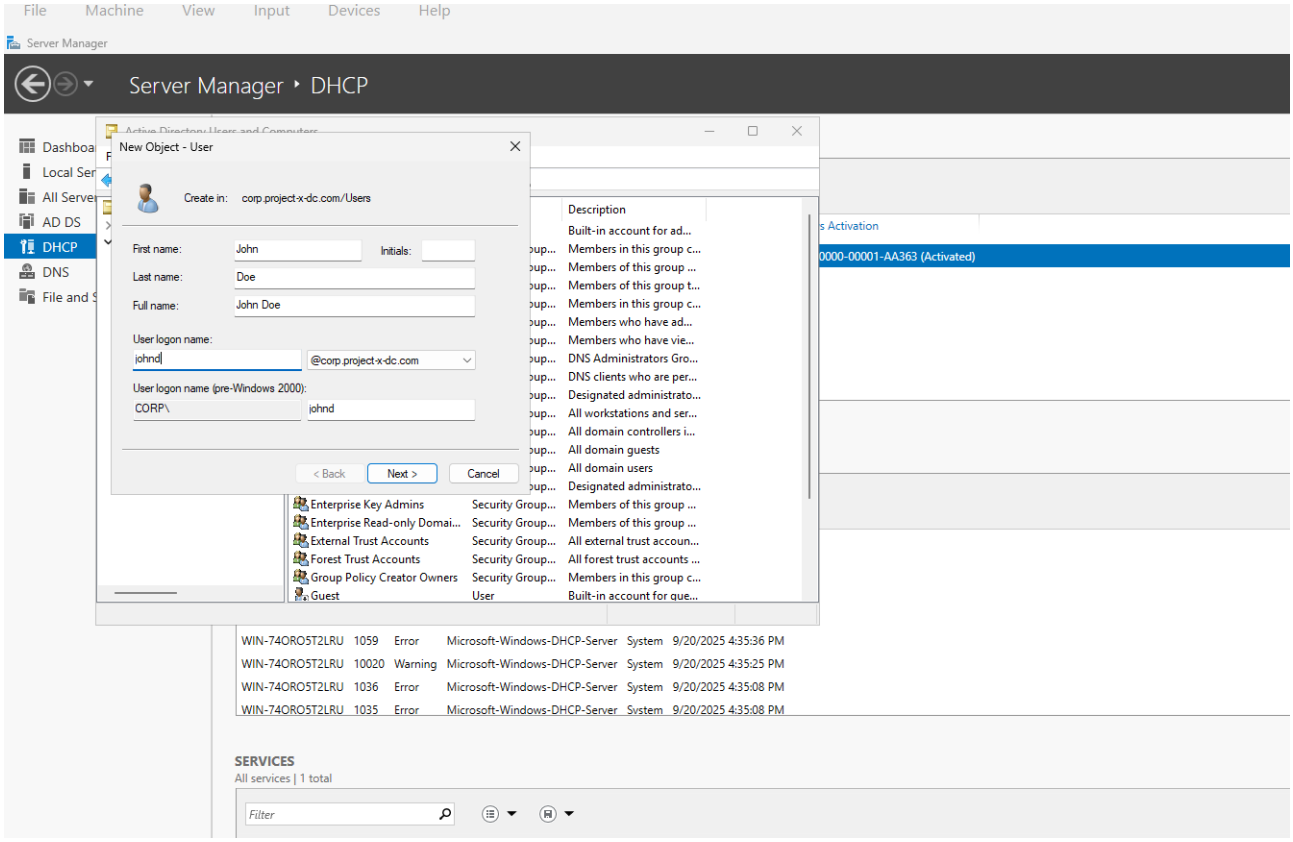
10. Create AD Users

- Server Manager → Tools → Active Directory Users and Computers → Users folder → New User.

- Example users: John Doe (Windows 11 workstation) and Jane Doe (Linux workstation).

- Set password, optionally disable password changes for simplicity in lab environment.

Provides user accounts for workstations to authenticate to the domain and participate in lab exercises.



11. Take Snapshot

- VM → Machine → Take Snapshot → Name & Save.

Saves the VM's current state, allowing rollback in case of errors or mistakes, ensuring safe experimentation and repeatable lab setups.

Step 6: Provisioning Windows 11 Enterprise (Enterprise Workstation 1):

1. Create VM

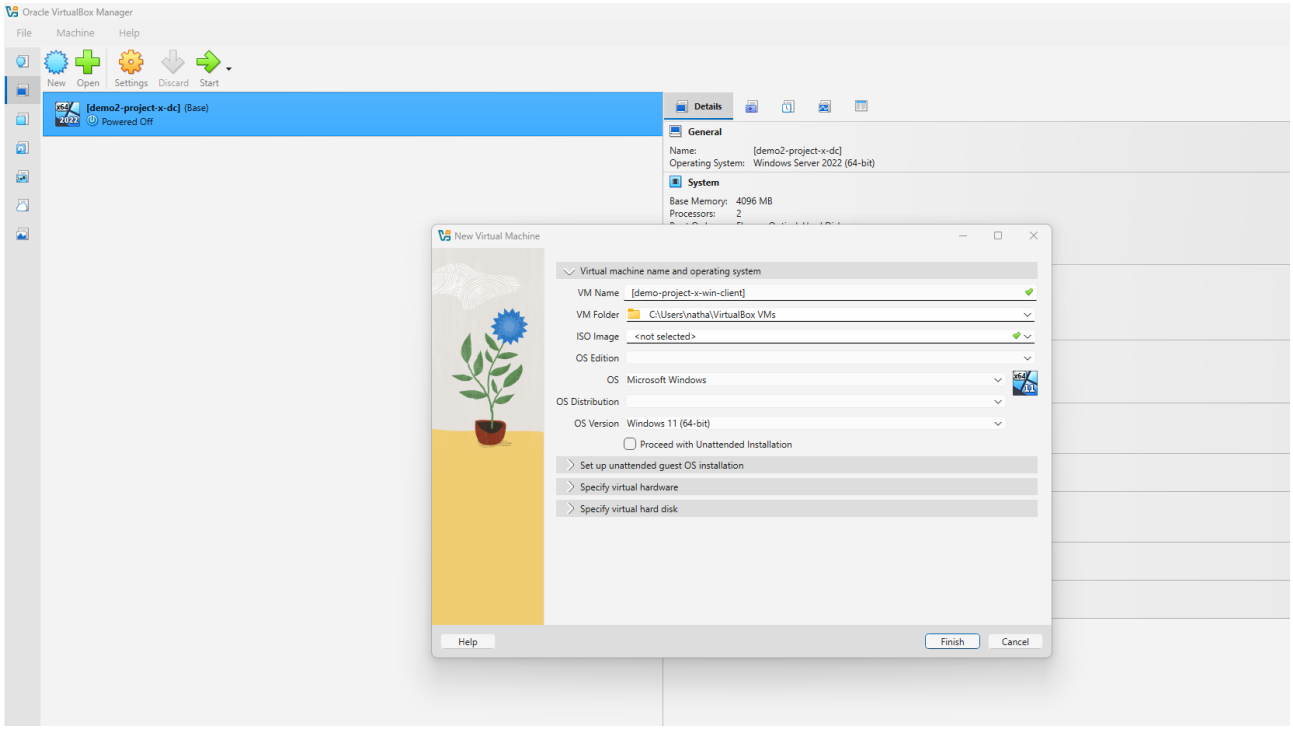
- Name VM, type Microsoft Windows, version Windows 11 (64-bit).

- Memory: 4096 MB, Processors: 2.

- Virtual Disk: 80 GB, default settings.

- Connect VM to NAT network.

Baseline hardware to run Windows 11 smoothly in the lab.



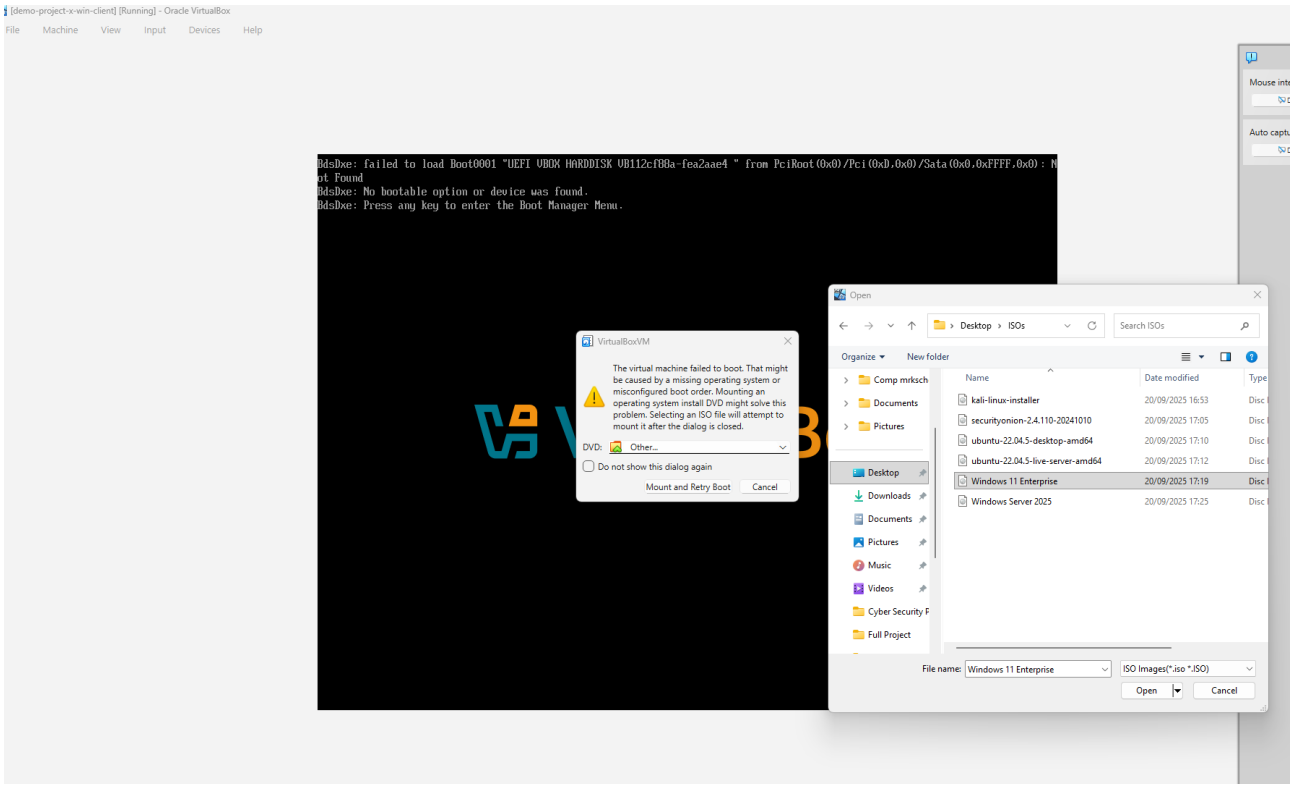
2. Mount ISO & Install OS

- Boot from Windows 11 Enterprise ISO.

- Select language, region, install Windows 11, create new partition (default settings).

- Set local Administrator password.

Clean installation ensures stable VM setup.



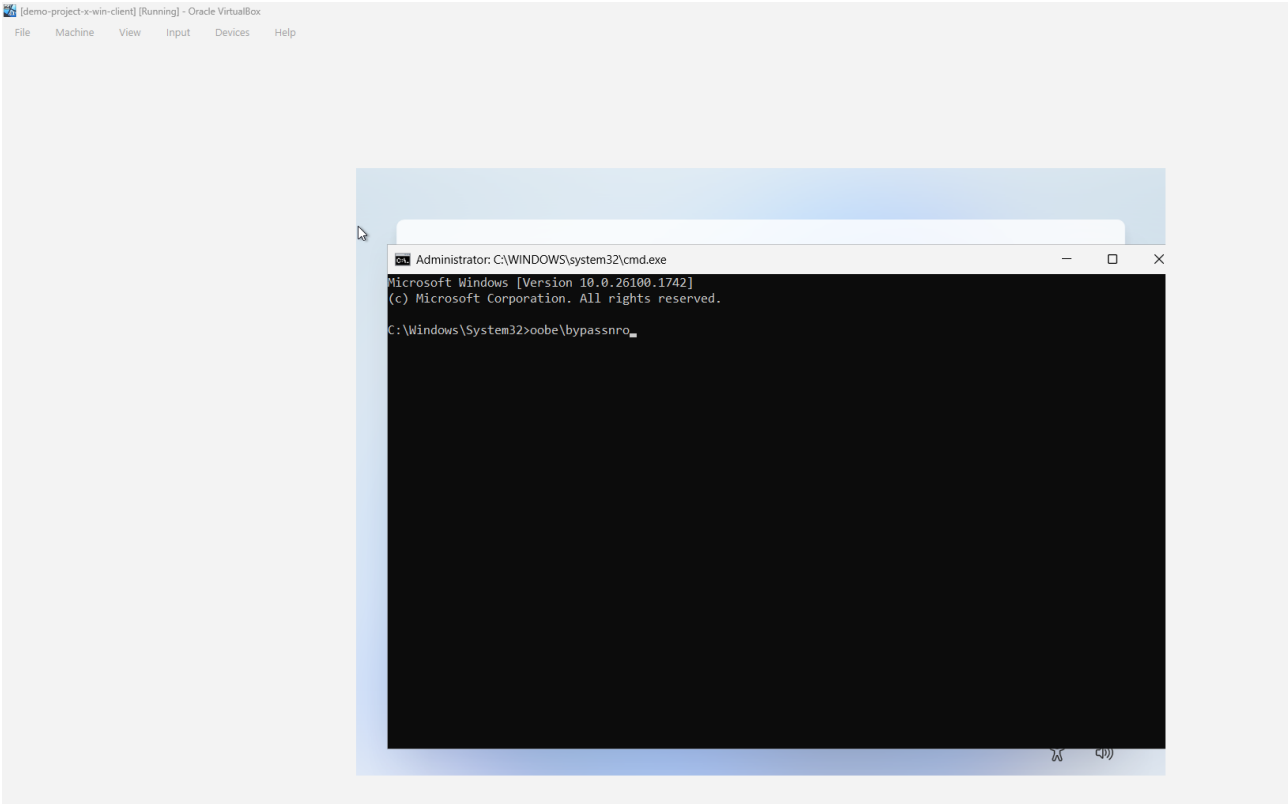
3. Convenience & Usability

- Bypass online account: Switch network to Host-only, open Command Prompt (Shift+F10), run oobe\bypassnro, complete OOBE wizard offline.

- Disable auto logoff & Ctrl+Alt+Del requirement: Settings → Accounts → Sign-in options → Require sign-in → Never; gpedit.msc → Local Policies → Security Options → "Interactive logon: Do not require Ctrl+Alt+Del" → Enabled.

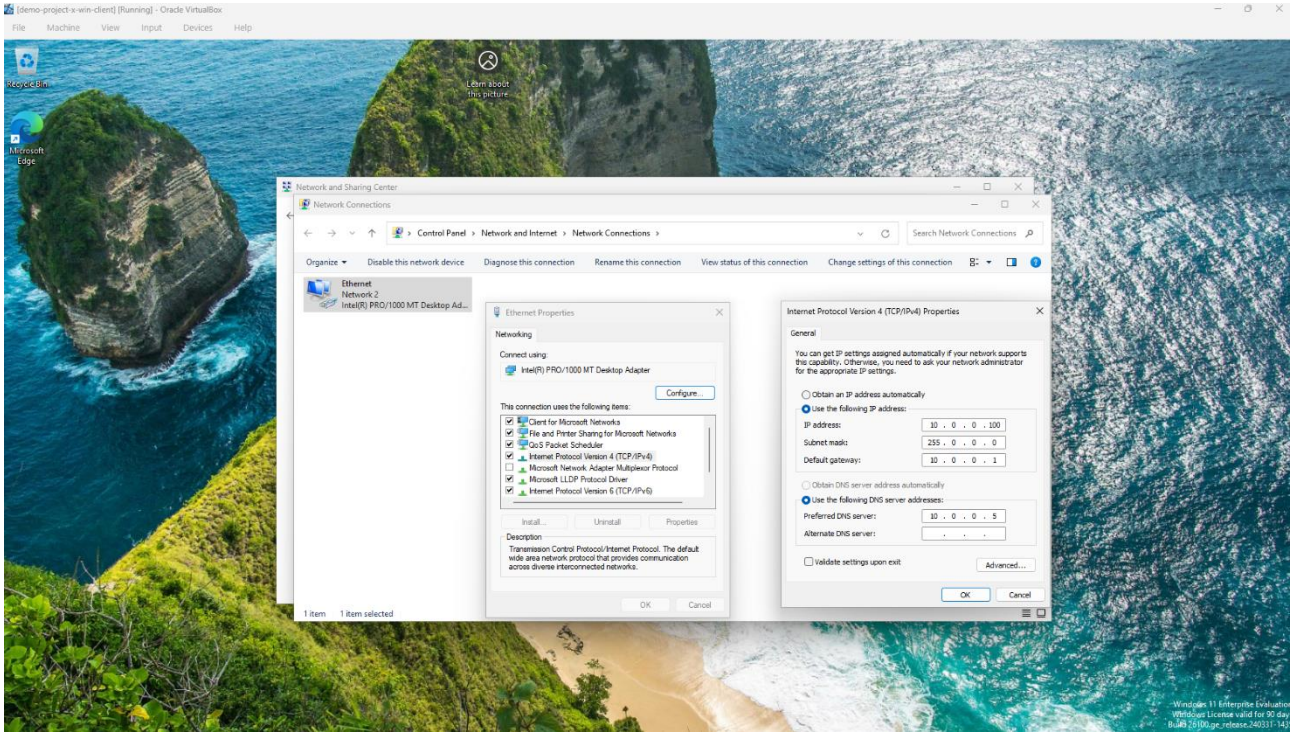
- Install VirtualBox Guest Additions: Devices → Insert Guest Additions CD → Run VBoxWindowsAdditions.exe → Reboot.

- Enable full screen, shared clipboard (bidirectional), drag & drop (bidirectional).
Simplifies setup and improves interaction with host for efficiency.



4. Assign Static IP

- IPv4: 10.0.0.100, Subnet: 255.255.255.0, Gateway: 10.0.0.1, DNS: 10.0.0.5
Ensures predictable connectivity with domain controller and lab resources.



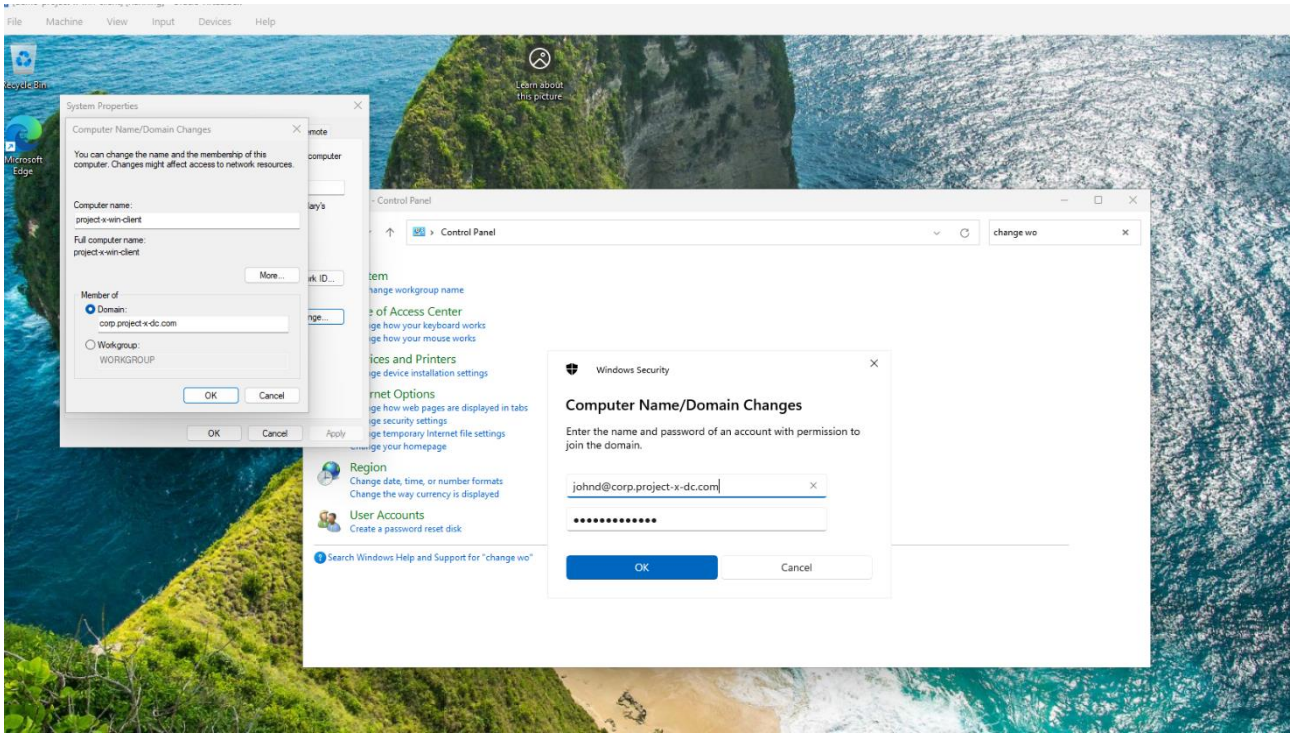
5. Join Domain & Configure Computer

- Rename VM (e.g., project-x-win-client).

- Join domain: corp.project-x-dc.com using AD user johnd.

- Reconnect to NAT network and restart.

Adds VM to managed network, granting access to domain resources and applying AD policies.



6. Snapshot

- Take VM snapshot after setup.

Allows rollback in case of misconfiguration or testing errors.

Step 7: Provisioning Ubuntu VM (Enterprise Workstation 2):

1. Create VM

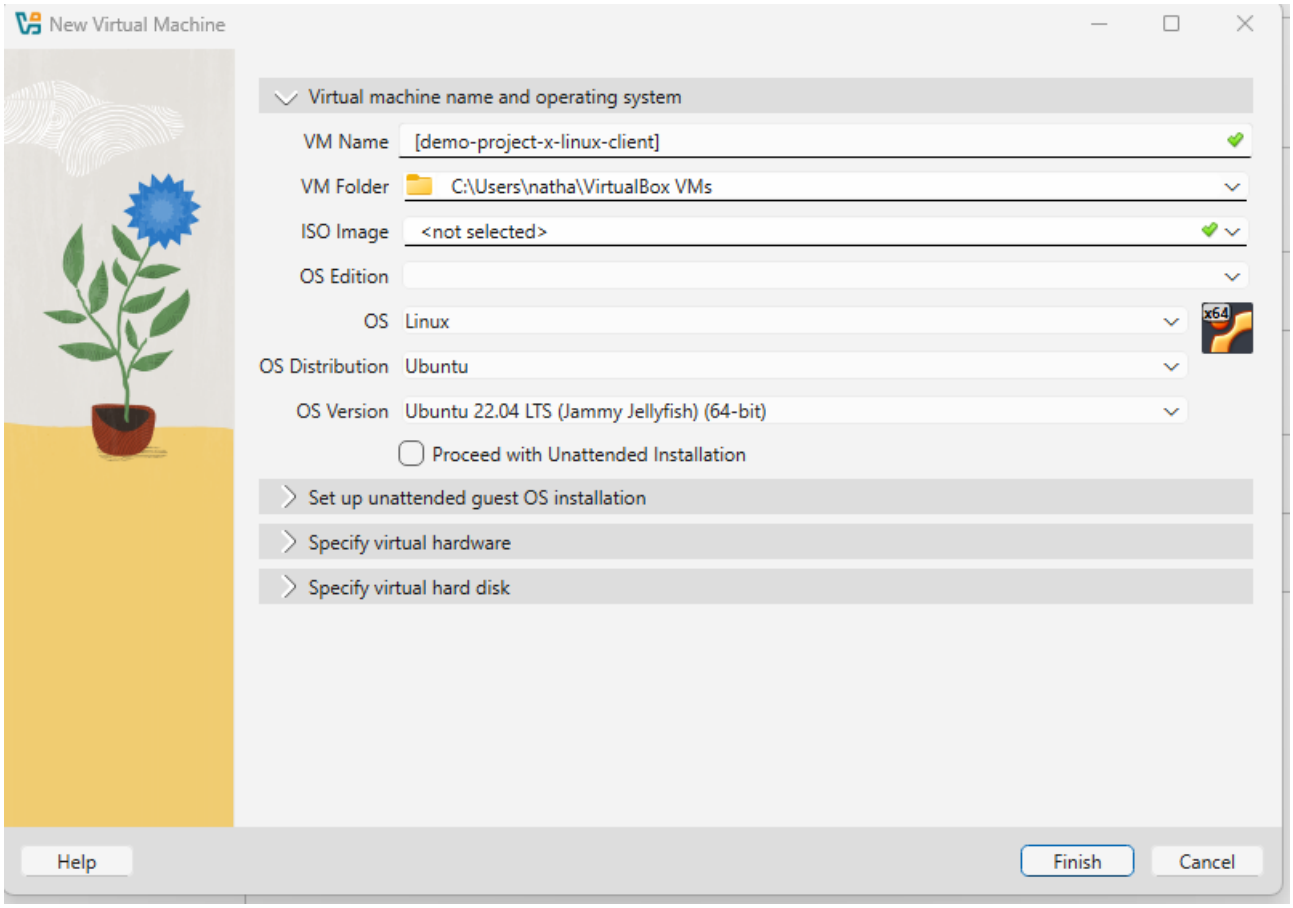
- Name VM, type Linux, version Ubuntu 22.04 LTS (64-bit).

- Memory: 2048 MB, CPU: 1.

- Virtual Disk: 80 GB, default settings.

- Connect to NAT network.

Baseline hardware to run Ubuntu for lab tasks.



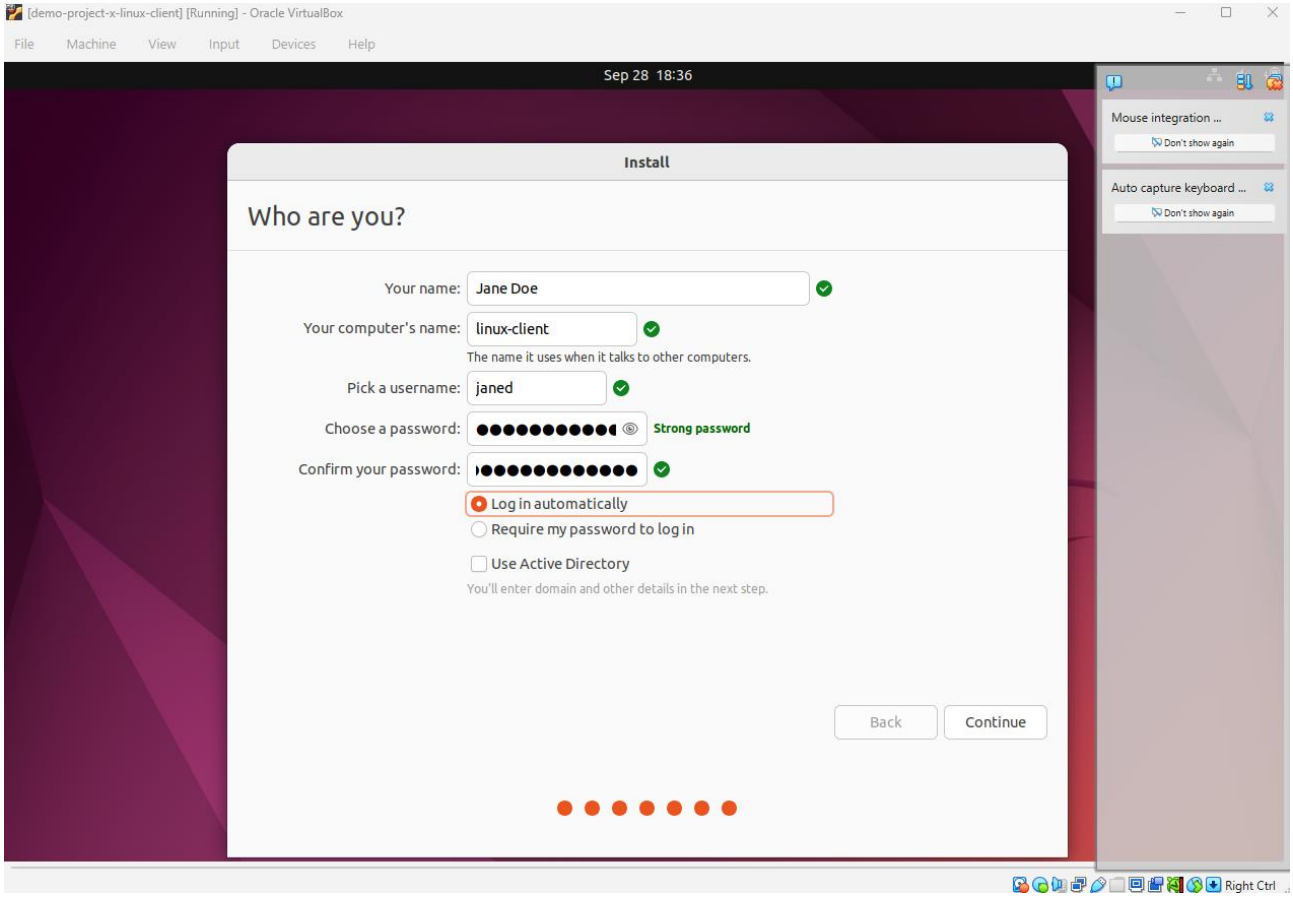
2. Mount ISO & Install OS

- Boot from Ubuntu 22.04.5 Desktop ISO, install with default options.

- Create partitions (default settings), set local user (Jane Doe).

- Skip online accounts, Ubuntu Pro, and location services. Disable screen blanking.

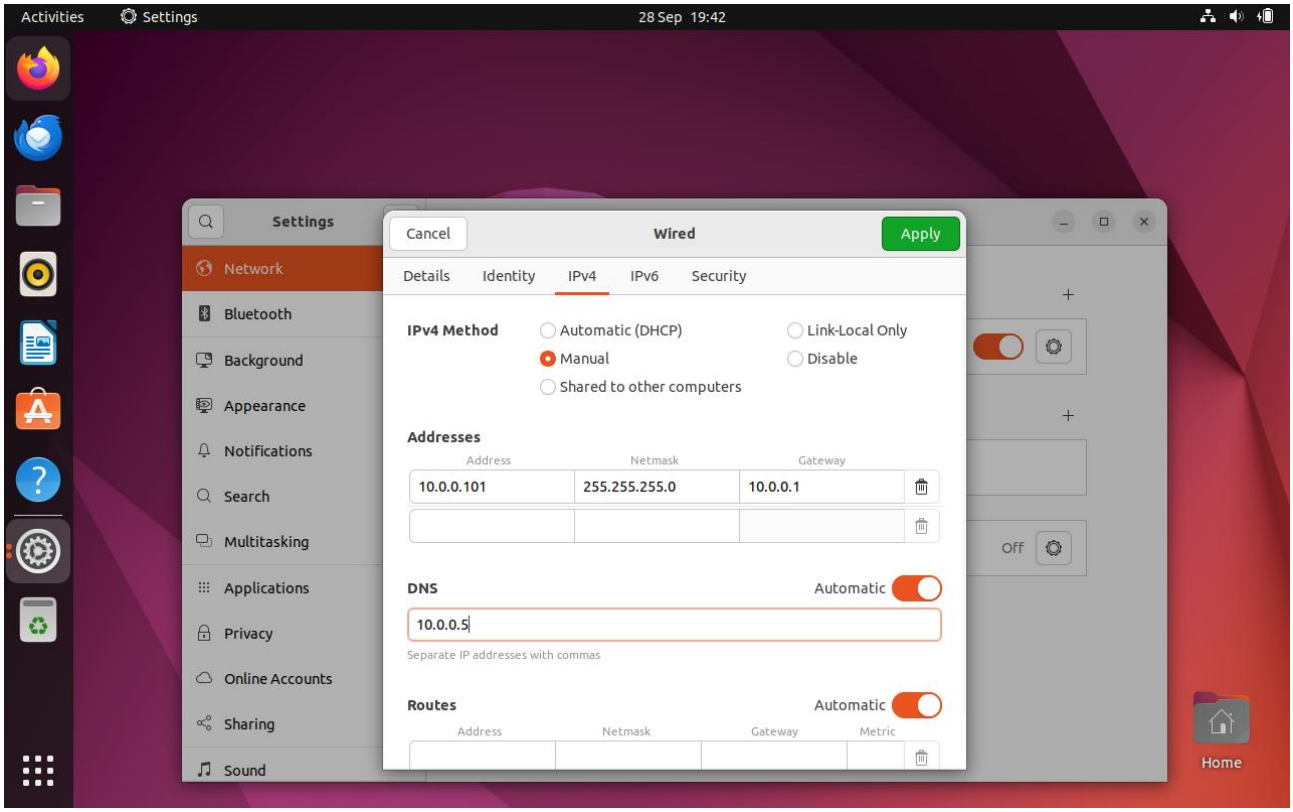
Clean installation ensures stability and local offline setup.



3. Assign Static IP

- IPv4: 10.0.0.101, Subnet: 255.255.255.0, Gateway: 10.0.0.1, DNS: 10.0.0.5

Ensures predictable network address and connectivity with domain controller.



4. Join Active Directory Domain

- Install required packages: winbind, libpam-winbind, libnss-winbind, krb5-config, samba modules.

- Configure Samba: /etc/samba/smb.conf with realm, workgroup, Kerberos, ADS security, Winbind settings.

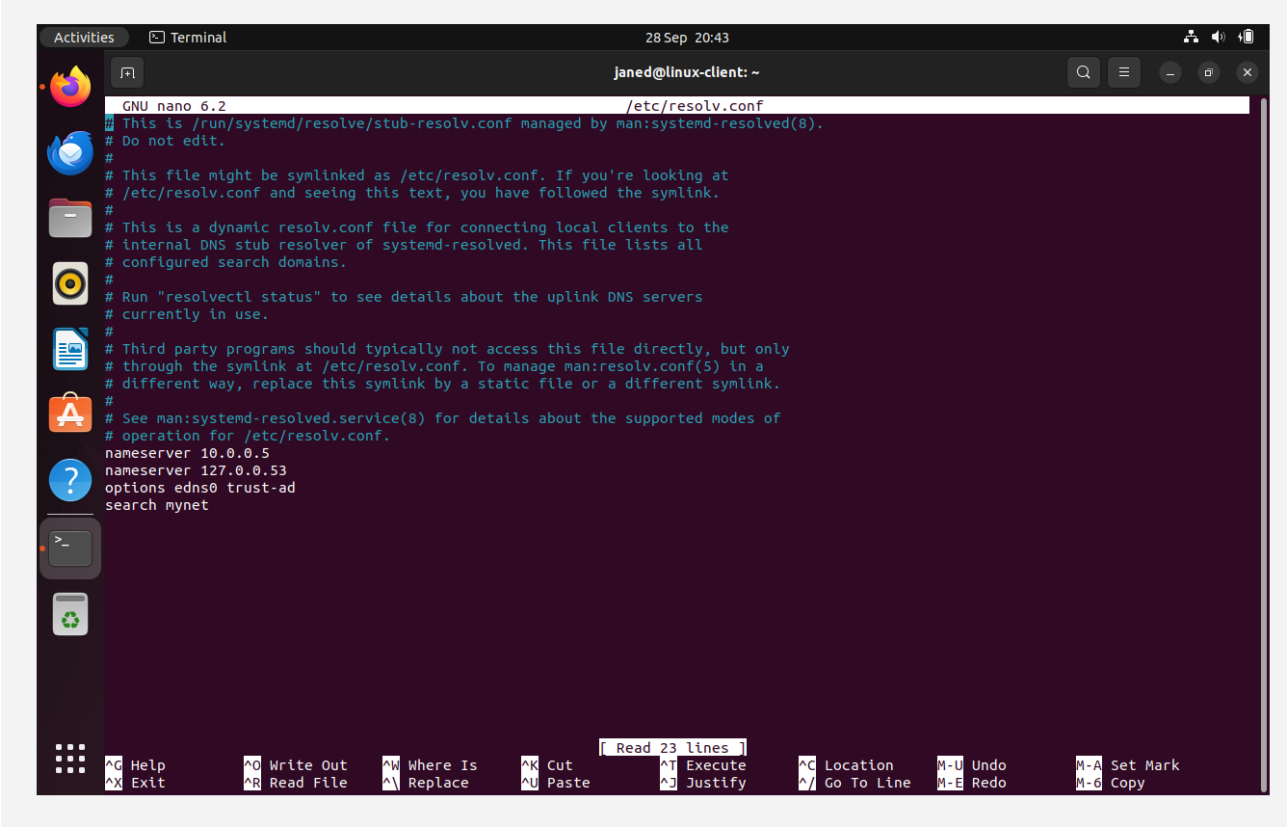
- Update /etc/nsswitch.conf to include Winbind in passwd and group.

- Enable home directory creation for domain users via sudo pam-auth-update.

- Update DNS to point to AD server (10.0.0.5).

- Join domain: sudo net ads join -U Administrator, restart Winbind: systemctl restart winbind.

Integrates Linux workstation with Active Directory for centralised authentication and domain resource access.



5. Convenience & Usability

- Set screen blanking to never via Ubuntu settings.

- Install VirtualBox Guest Additions (Devices → Insert Guest Additions CD → Run installer).

Improves usability by preventing screen lock and enabling full screen, shared clipboard, and drag/drop functionality between host and VM.

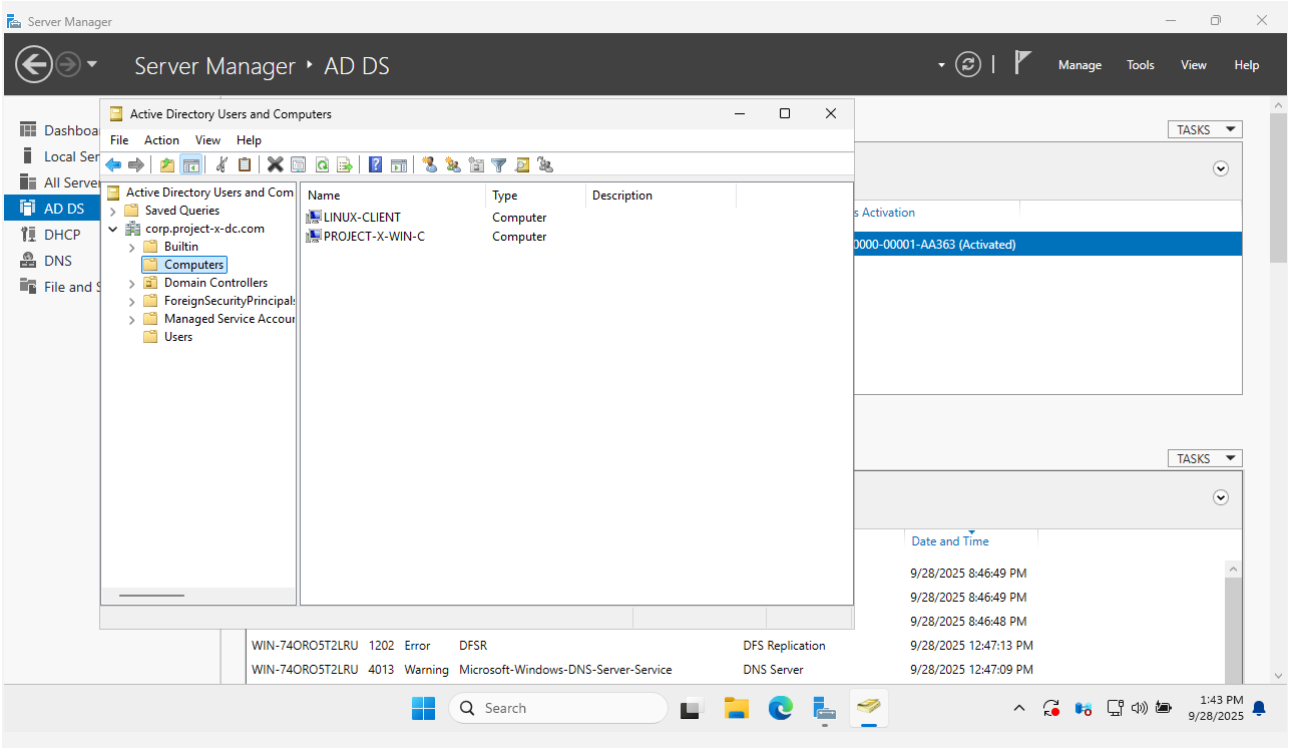
6. Test AD Connectivity & Login

- Verify users: wbinfo -u, check domain info: net ads info.

- Login as domain user: CORP+janed, creates /home/CORP/janed.

- Confirm connection on Domain Controller → Active Directory Users and Computers → Computers shows Linux client.

Confirms successful domain join, user authentication, and home directory creation.



7. Snapshot

- Take VM snapshot after setup.

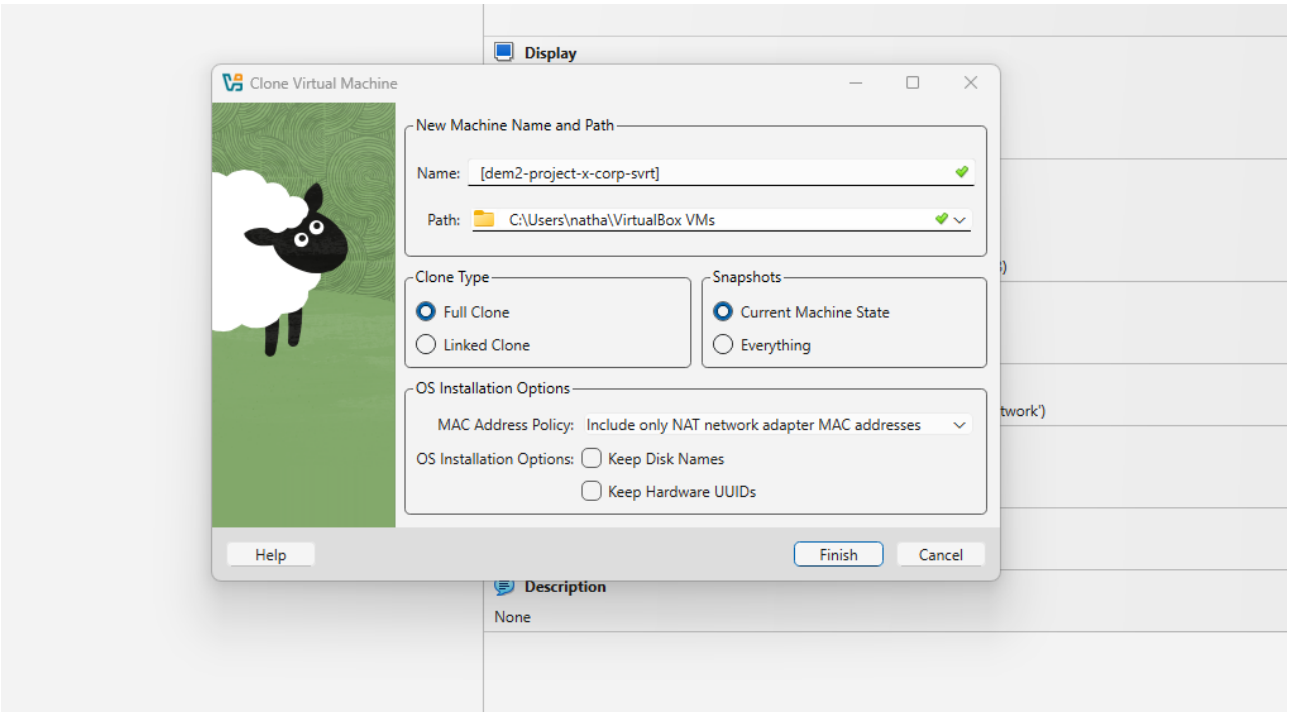
Allows rollback in case of misconfiguration or testing errors.

Step 8: Provisioning Ubuntu VM (Security Server 1/Corporate Server):

1. Clone VM

- Clone Ubuntu Desktop VM from Step 7.14 (Enterprise Workstation 2).

Saves time and ensures consistency by reusing preconfigured OS and dependencies.



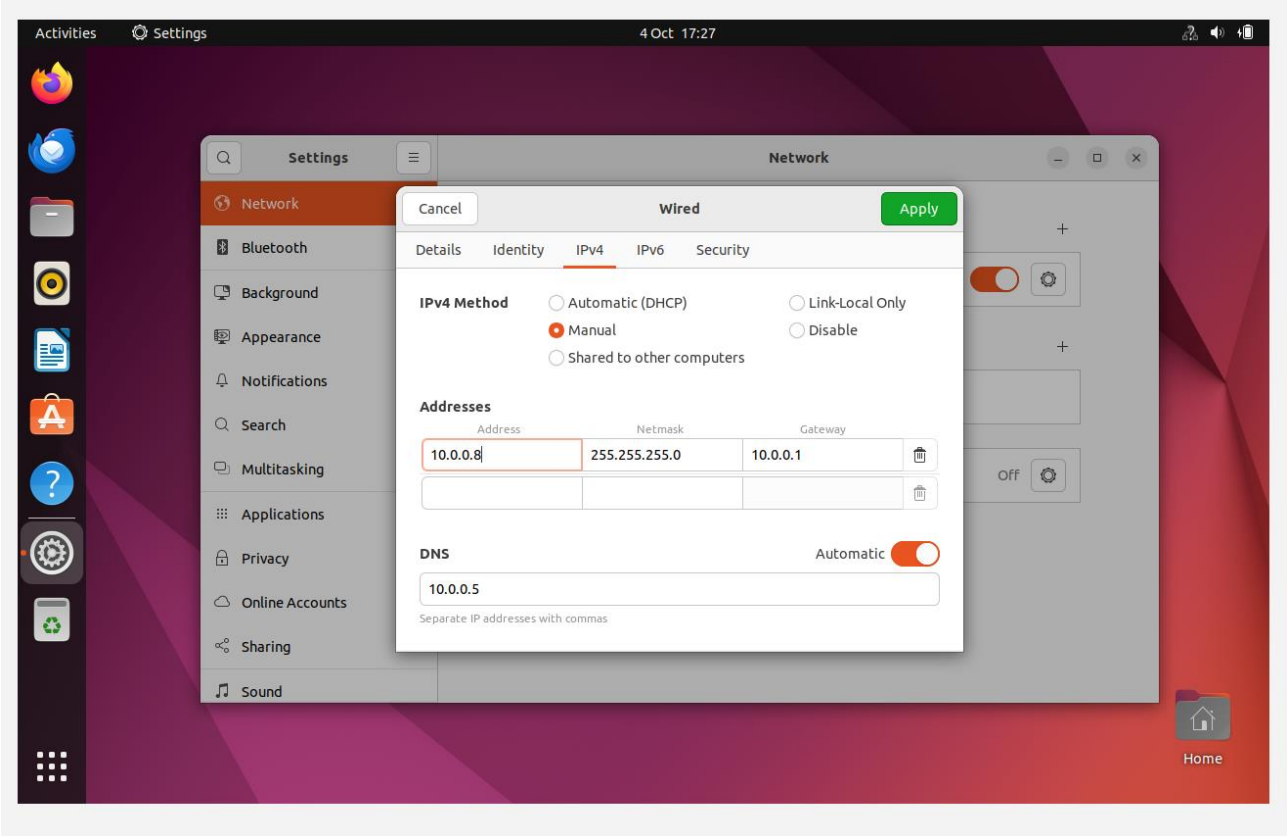
2. Configure VM

- Set static IP: 10.0.0.8 (outside DHCP range).

- Set hostname: corp-svr (sudo hostnamectl set-hostname corp-svr).

- Create administrative user: project-x-admin, add to sudo group.

Ensures predictable network identity, domain integration, and secure administration.



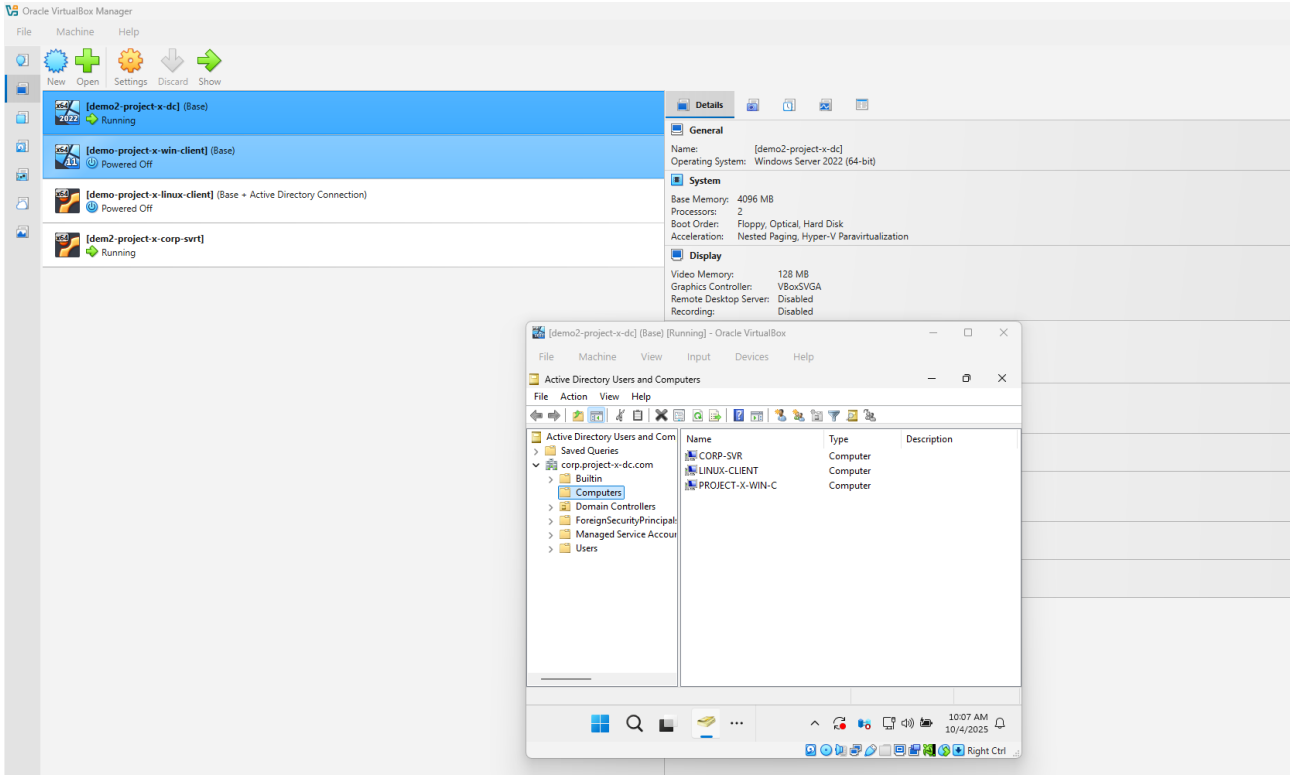
3. Join Active Directory Domain

- Ping DC to confirm connectivity.

- Join domain: sudo net ads join -U Administrator.

- Log in to generate home directory: CORP+Administrator.

Integrates server with AD for authentication, group policies, and domain services.



4. Convenience & Usability

- Skip Ubuntu setup wizards, set screen blank to never.

- Install VirtualBox Guest Additions (Devices → Insert Guest Additions CD → Run installer).

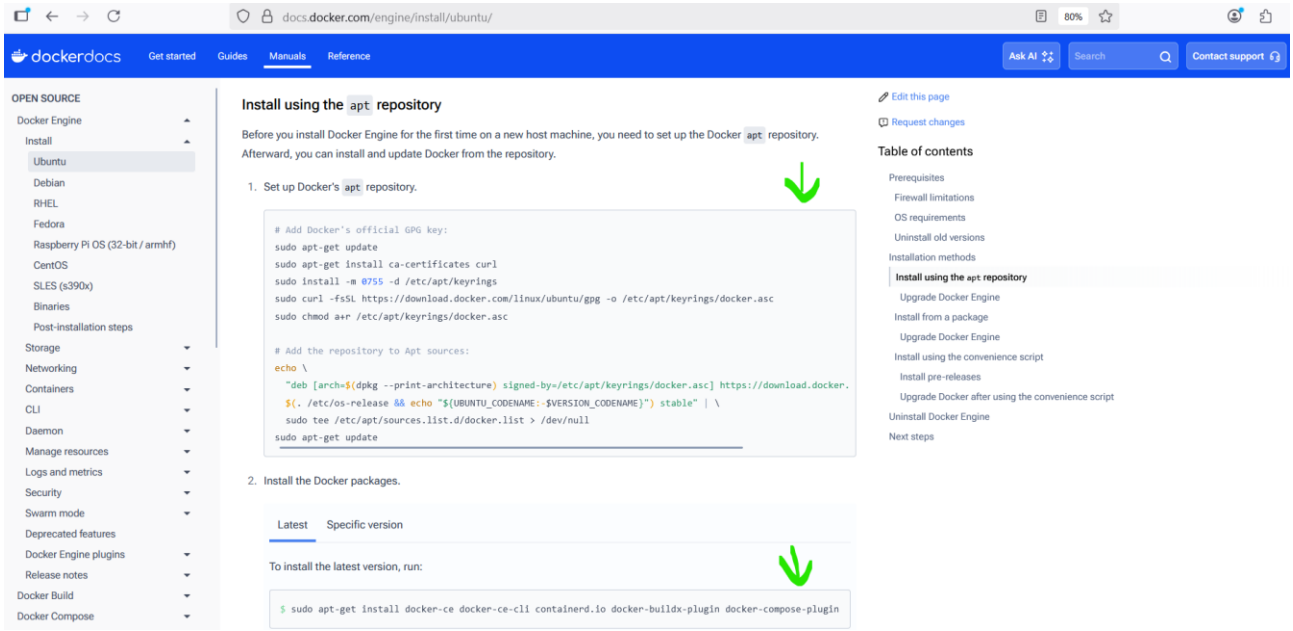
Improves usability by preventing screen lock and enabling full screen, shared clipboard, and drag/drop functionality between host and VM.

5. Install Docker

- Install via apt repository (per official instructions).
https://docs.docker.com/engine/install/ubuntu

- Test installation: sudo docker run hello-world, sudo docker ps -a.

Enables containerised services (email, DNS, FTP) in isolated environments for secure administration.



6. Snapshot

- Take VM snapshot after configuration.

Allows rollback in case of misconfiguration or testing errors.

Step 9: Install Mailhog on Security Server 1:

1. Set Up MailHog

- On Security Server 1 VM:

```
cd /home
sudo mkdir mailhog
cd mailhog
sudo nano docker-compose.yml
```

- Paste Docker Compose config:

```
version: "3"
services:
  mailhog:
    image: mailhog/mailhog
    container_name: mailhog
    ports:
      - "1025:1025"
      - "8025:8025"
```

- Start container: sudo docker compose up -d

Runs a lightweight, isolated SMTP server to safely capture and inspect emails.

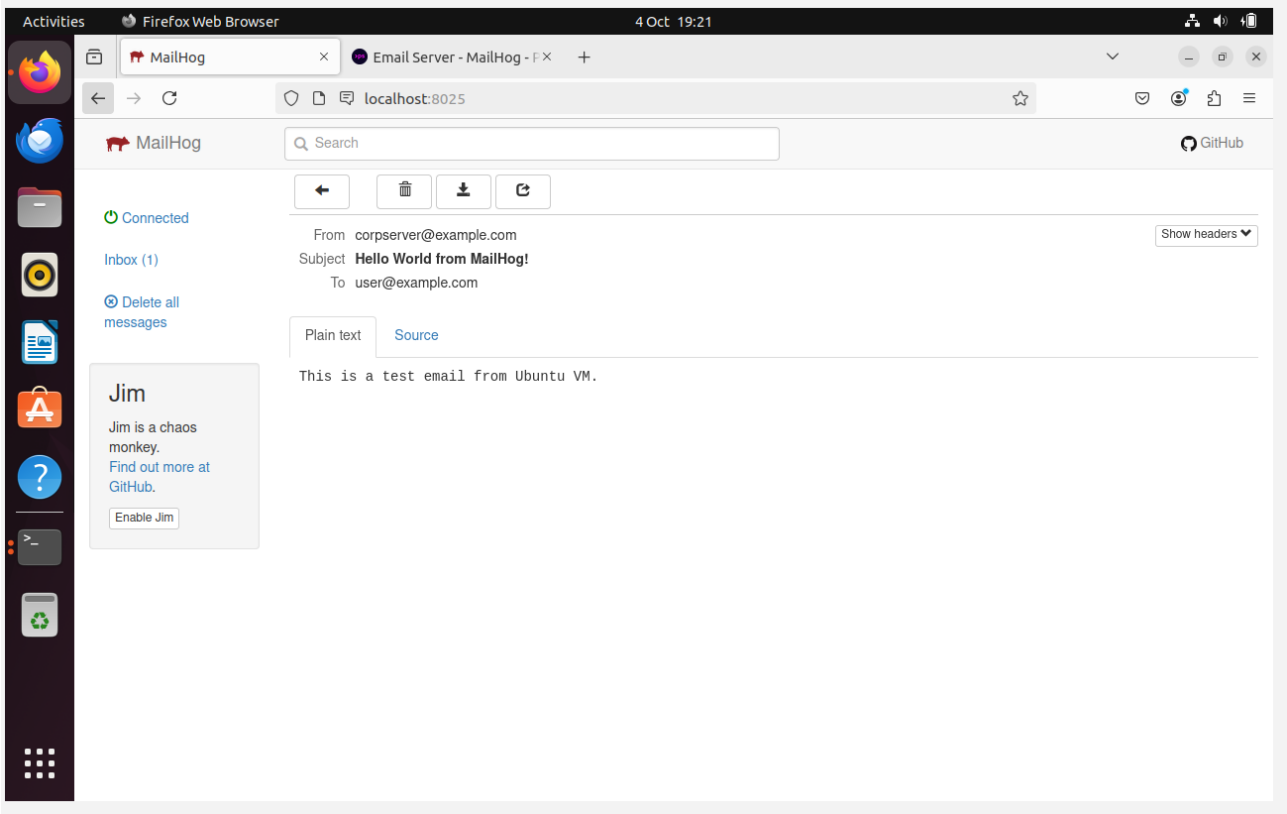
2. Test MailHog Locally

- Create test email script test_message.py, make executable and run:

sudo chmod +x test_message.py
sudo python3 test_message.py

- Open dashboard: http://localhost:8025

Verifies container captures emails correctly within the lab.



3. Poll Emails from Linux Client

- Create polling script email_poller.sh, install dependencies:

sudo apt update
sudo apt install curl jq -y
sudo chmod +x email_poller.sh
./email_poller.sh &

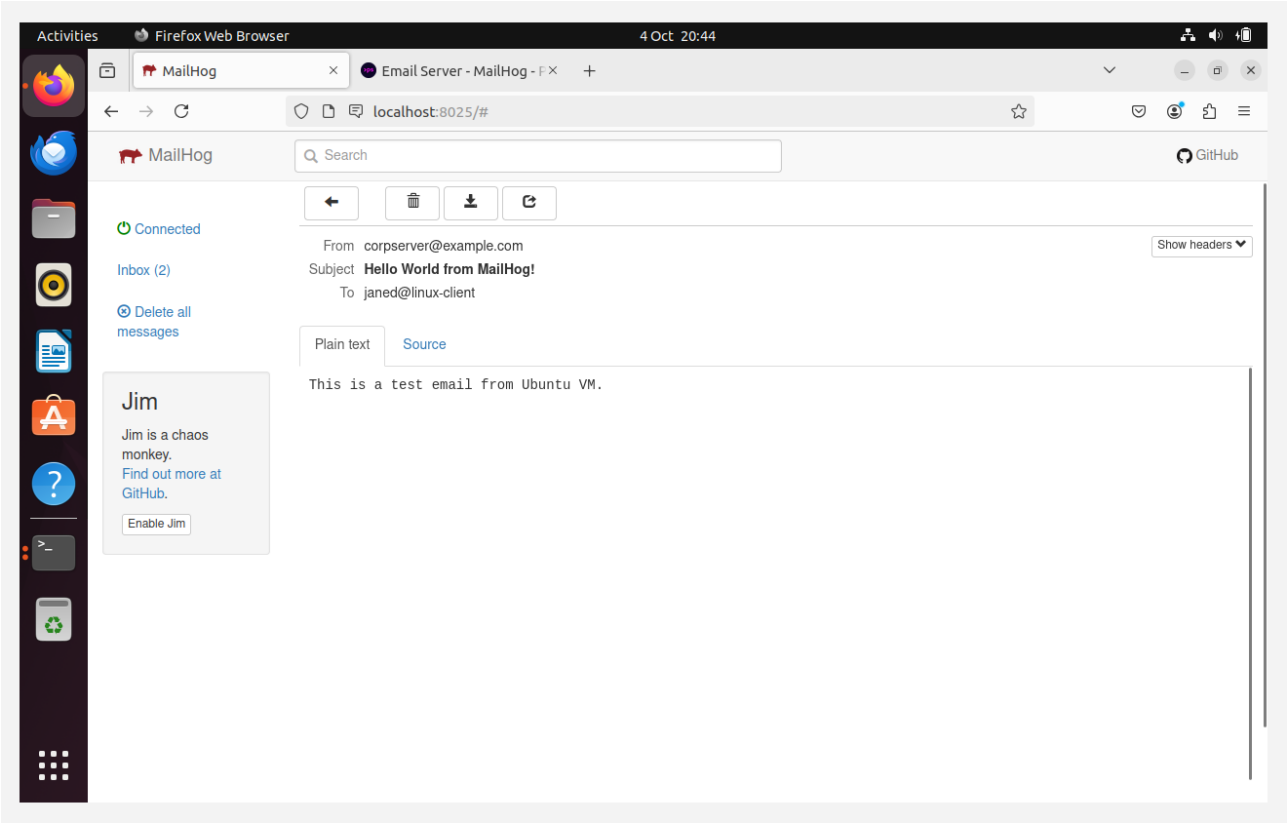
- Script continuously monitors MailHog and displays new messages.
Ensures Linux client can safely receive and display emails from MailHog.

4. End-to-End Test

- Update test_message.py to target Linux client user and run.

- Verify dashboard and client output show new email.

Confirms cross-VM email delivery within the lab.



5. Take Snapshot

- Snapshot Security Server 1 VM in VirtualBox.

Preserves MailHog setup for quick rollback or restoration.

Step 10: Provisioning Security Onion VM (Security Workstation):

1. Create Security Onion VM

- Name: demo-project-x-sec-work

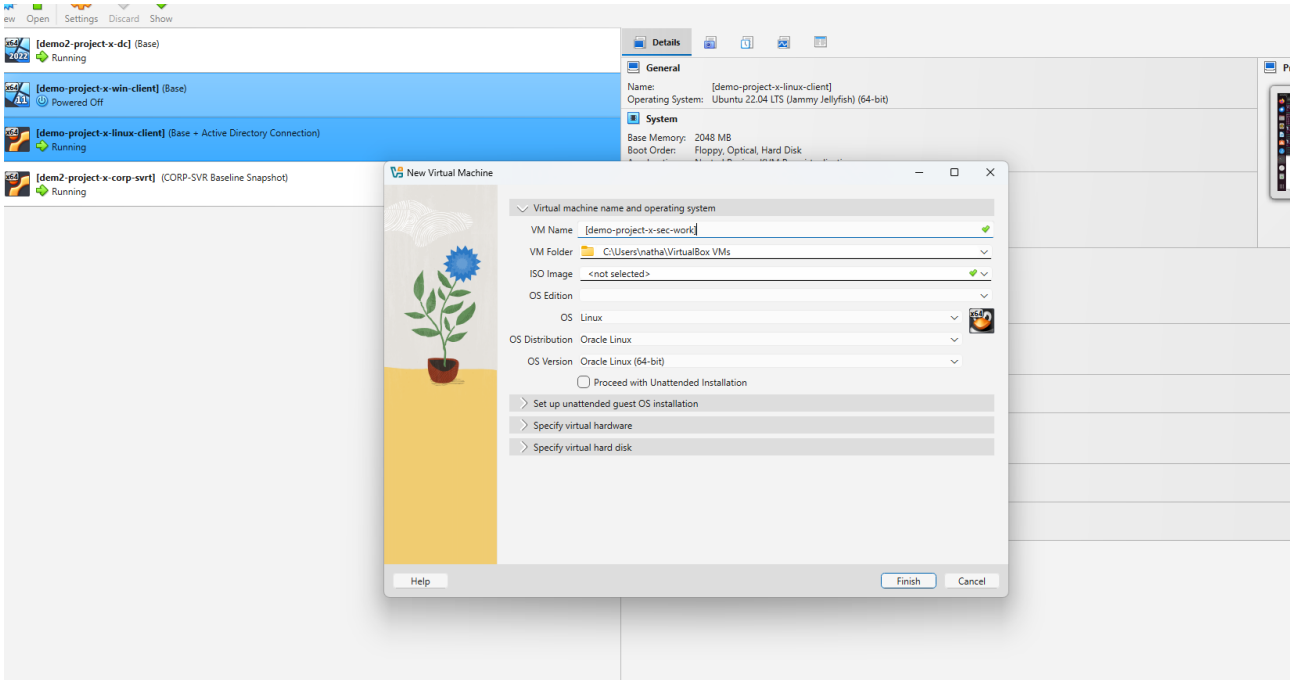
- Type: Linux, Version: Oracle Linux (64-bit)

- Memory: 2048 MB, CPU: 1

- Disk: 55 GB

- Connect to NAT network

Baseline hardware to run Security Onion for network monitoring.



2. Mount ISO & Install OS

- Boot from Security Onion 2.4.110 Desktop ISO

- Localhost login: project-x-sec-work (set password)

- Hostname: project-x-sec-work

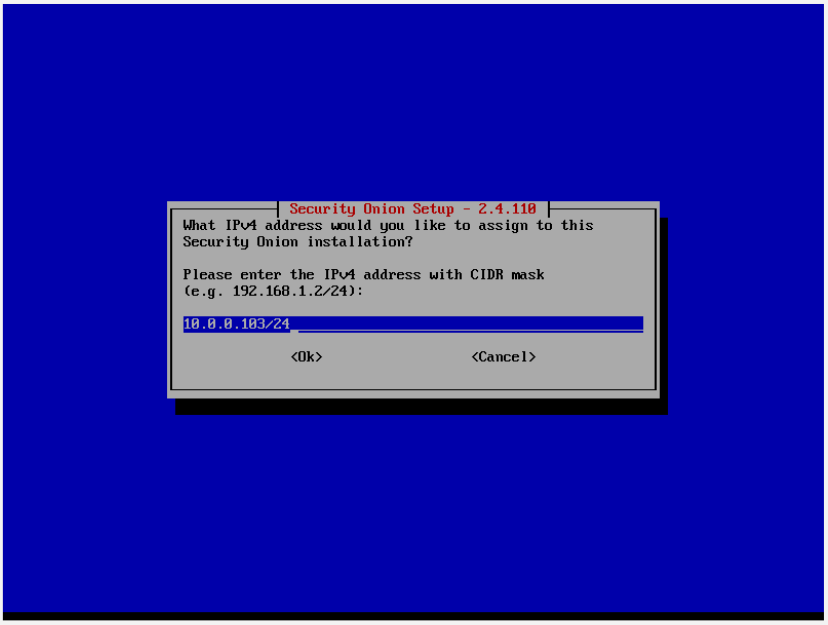
- NIC: default, Static IP: 10.0.0.103/24

- Gateway: 10.0.0.1, DNS: default, search domain: corp.project-x-dc.com

- Graphical interface: enabled

- Reboot VM

Provides a predictable IP and domain configuration for monitoring and ensures GUI access for management.



3. Root Password

- Open terminal: sudo passwd root → set new password

Enables administrative access for Security Onion configuration and management.

4. Take Snapshot

- Snapshot VM in VirtualBox

Preserves a working baseline for the Security Onion VM for quick rollback or restoration.

Step 11: Provisioning Ubuntu Server VM (Security Server 2):

1. Create Security Server 2 VM

- Clone Linux Client VM → rename demo-project-x-sec-box.

- Update hostname: sudo nano /etc/hostname → replace with sec-box, reboot.

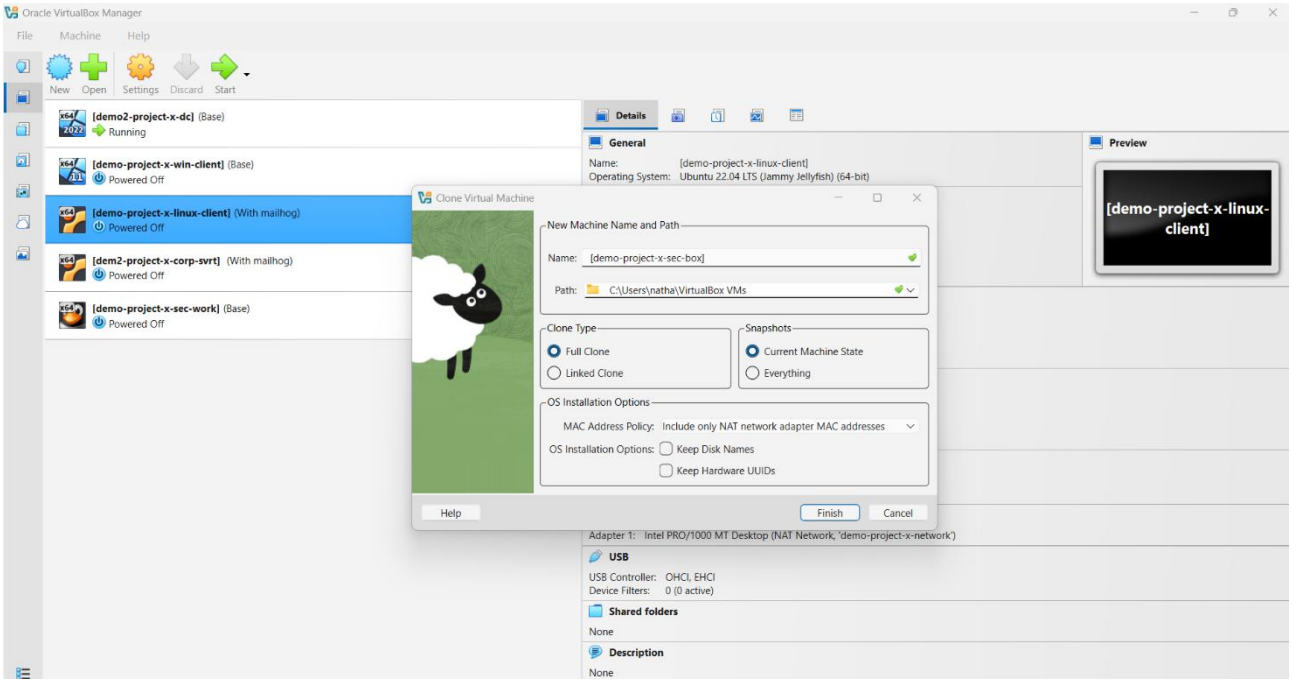
- Create new user:

sudo adduser sec-user (set password)

sudo usermod -aG sudo sec-user

Switch to sec-user: su sec-user → verify sudo with sudo whoami.

Prepares a dedicated Ubuntu Server VM for hosting Wazuh with admin privileges.

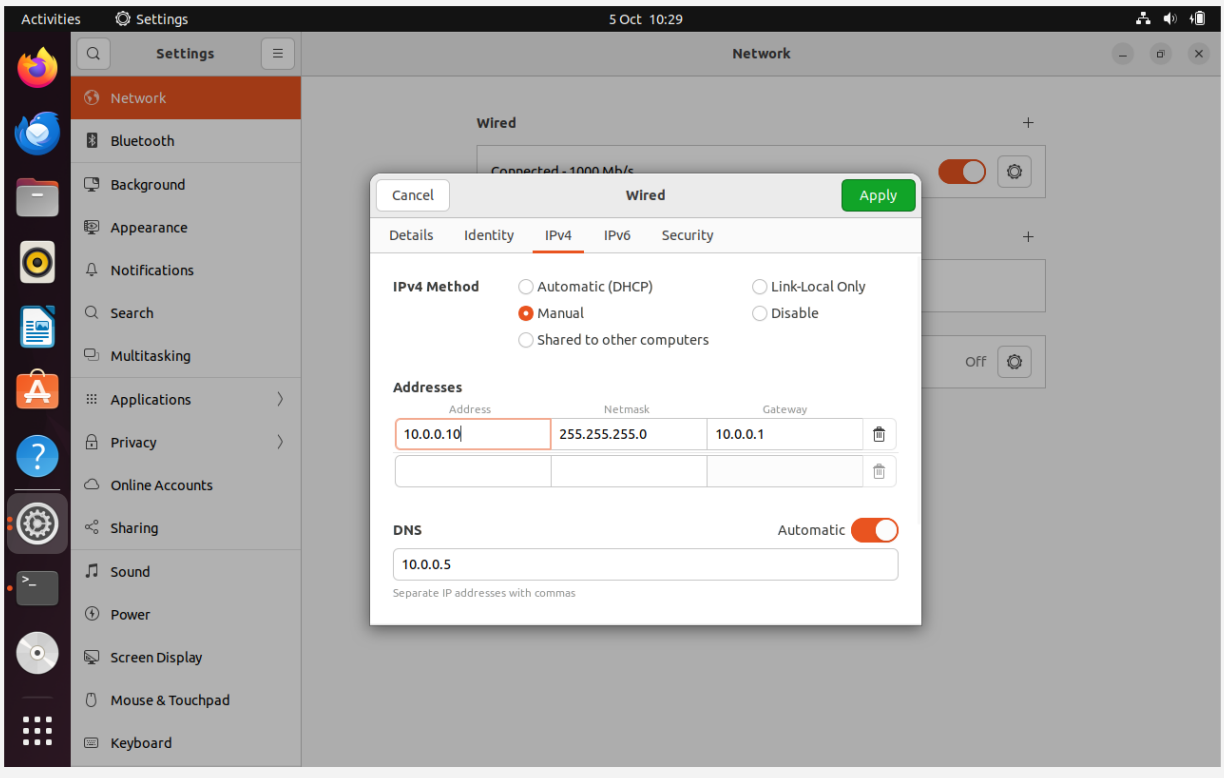


2. Configure Network

- Set static IP: 10.0.0.10.

- Test connectivity: ping 10.0.0.5 and ping corp.project-x-dc.com.

Ensures server has predictable addressing and communication with AD.



3. Domain Integration

- On Domain Controller: create new AD user + new domain admin group.

- On Security Server 2:

    - Restart winbind: sudo systemctl restart winbind.

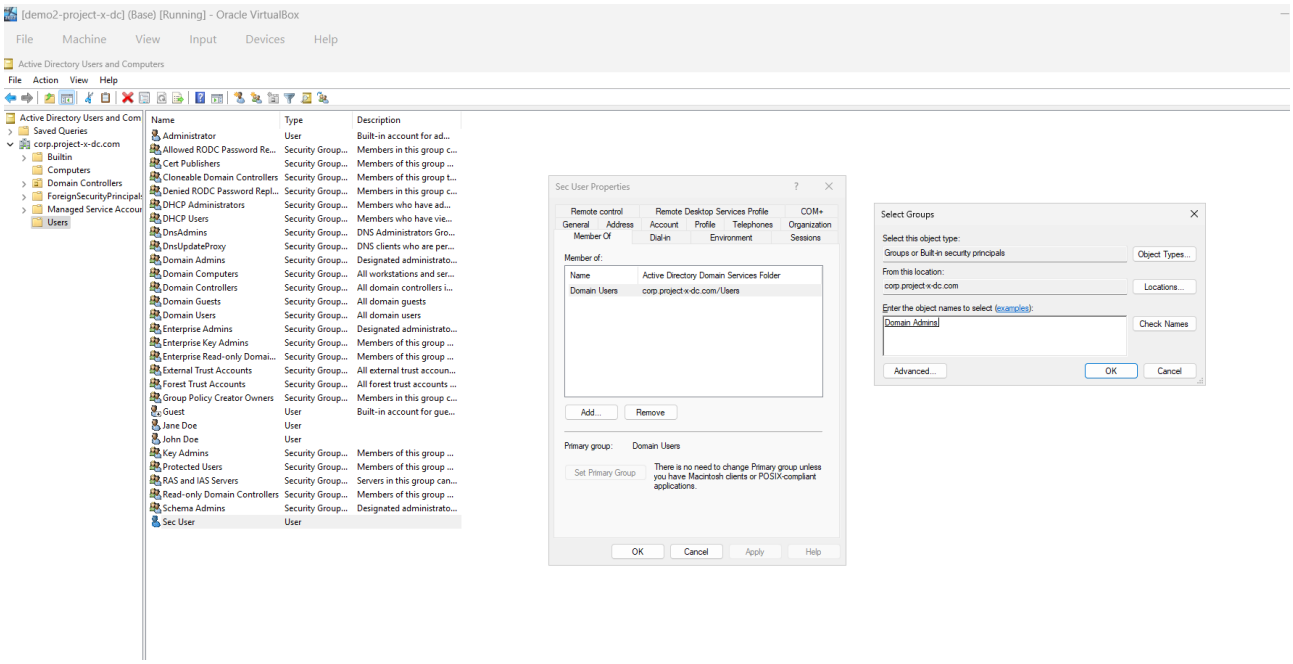
    - Join domain: sudo net ads join -U Administrator.

    - Restart winbind again.

    - Verify: wbinfo -u → should display new AD user.

- Test login: sudo login, run id to confirm domain membership.

Adds Security Server 2 to Active Directory, enabling central authentication and management.



4. Take Snapshot

- Snapshot VM in VirtualBox.

Saves a stable baseline after successful AD integration.

<u>Step 12: Installing And Setting Up Wazuh On Security Server 2 VM:</u>

1. Prepare Security Server 2 VM

- Update resources: 4096MB RAM, 2 CPUs, 80GB disk.

- Start VM, log in as sec-user.

Ensures server has enough capacity for Wazuh SIEM/XDR platform.

2. Install Wazuh

- Install curl: sudo apt install curl (already present on clone).

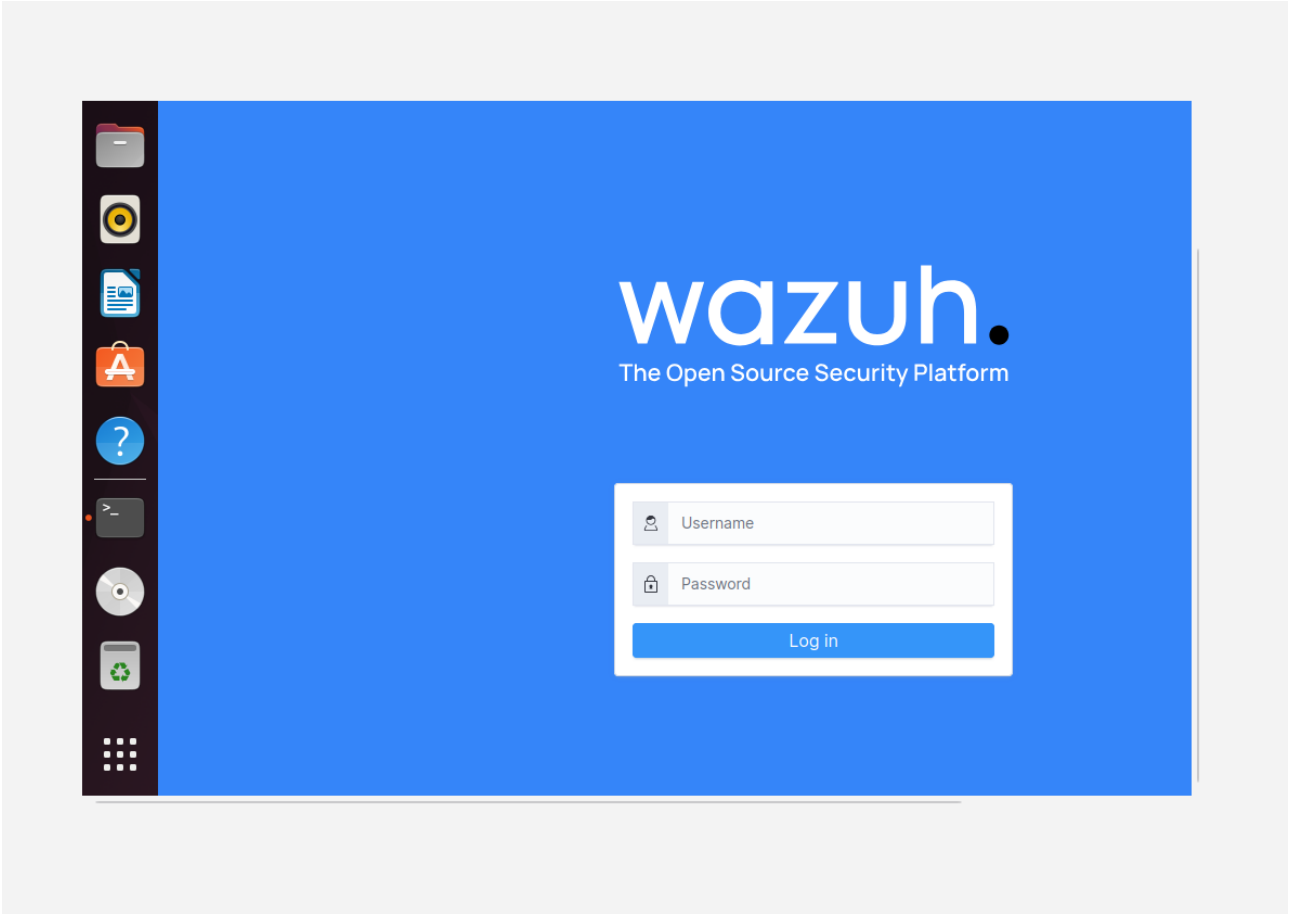
- Run installer:

curl -sO https://packages.wazuh.com/4.9/wazuh-install.sh && sudo bash ./wazuh-install.sh -a -i

- Save credentials: nano wazuh-password.txt.

- Access Wazuh UI at http://localhost, log in with generated user/password.

Installs Wazuh manager and web interface.



3. Deploy Agents

- Windows Workstation:

    - Wazuh → Server Management → Endpoints → Deploy Agent → Windows.

    - Server: 10.0.0.10, Agent: project-x-win-client.

- Run generated PowerShell commands on Windows client (as admin).

- Start service: NET START Wazuh.

- Confirm in Wazuh dashboard.

- Take snapshot of Windows client.

- Linux Client:

- Deploy new agent → Linux DEB amd64.

- Server: 10.0.0.10, Agent: project-x-linux-client.

- Run generated command on Linux client terminal.

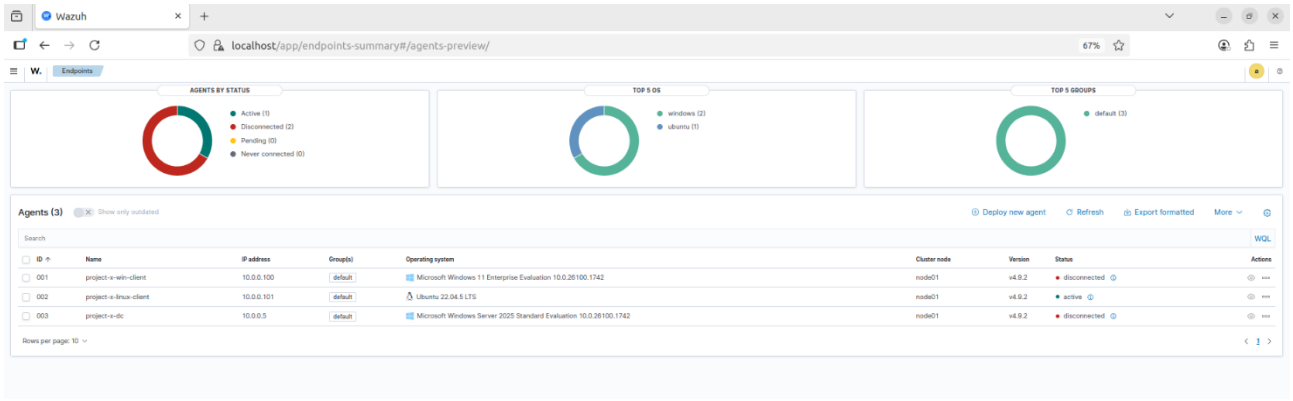
- Confirm in dashboard.

- Take snapshot of Linux client.

- Domain Controller:

- Repeat Windows agent steps on DC VM.

Connects all lab systems to Wazuh for centralised log collection.



4. Organise Agents

- Create groups in Wazuh: Windows and Linux.

- Assign agents to respective groups.

Groups allow applying OS-specific configurations.

5. Configure Agent Logs

- Windows (agent.conf):

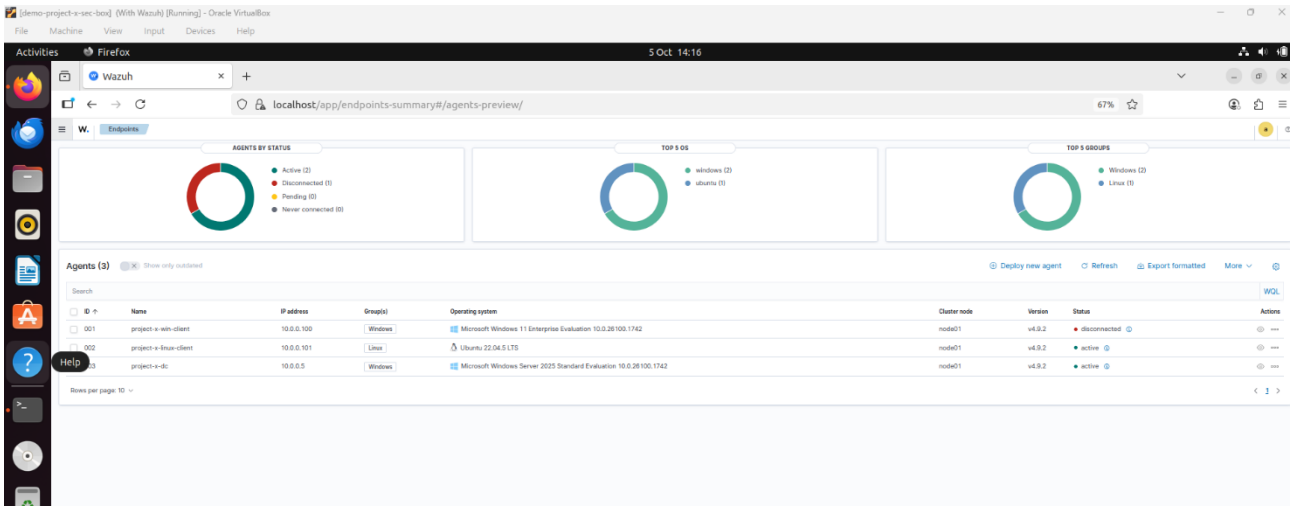
```
<agent_config>
        <!-- Shared agent configuration here -->
        <localfile>
                <location>Security</location>
                <log_format>eventchannel</log_format>
        </localfile>
        <localfile>
```

```
            <location>Application</location>
            <log_format>eventchannel</log_format>
        </localfile>
</agent_config>
```

- Linux (agent.conf):

```
<agent_config>
        <localfile>
            <log_format>syslog</log_format>
            <location>/var/log/auth.log</location>
        </localfile>
        <localfile>
            <log_format>syslog</log_format>
            <location>/var/log/secure</location>
        </localfile>
        <localfile>
            <log_format>audit</log_format>
            <location>/var/log/audit/audit.log</location>
        </localfile>
</agent_config>
```

Ensures native logs (eventchannel, syslog, audit) are collected per OS.



6. Verify & Snapshot

- Confirm logs appear in Wazuh dashboard.

- Take snapshots of Security Server 2 VM and all agents.

Establishes a clean baseline with Wazuh fully deployed and agents reporting logs.