

Server Management

- [Reboot a Server](#)
- [Shutdown a Server](#)
- [Bare Metal: Install Ubuntu 24.04](#)
- [DigitalOcean: Install Ubuntu 24.04](#)
- [VMWare: Install Ubuntu 24.04](#)

Reboot a Server

1. **Access Your Clusters**

- In the **left-hand menu**, click on **Clusters**.

2. **Select Your Desired Cluster**

- On the **Clusters Summary** page, locate and click on the cluster you wish to work with.
- You will see a list of all servers associated with that cluster.

3. **Choose the Server to Reboot**

- Scroll through the list and select the server you want to reboot.

4. **Initiate the Reboot Process**

- On the **right-hand side** of the selected server row, click on **Actions**.
- From the dropdown, choose **Reboot Server**.

5. **Confirm the Reboot**

- A confirmation prompt will appear. Click **Confirm** to proceed.
- The server will now begin the reboot process.

Following these steps will safely reboot your chosen server.

Shutdown a Server

1. **Access Your Clusters**

- In the **left-hand menu**, click on **Clusters**.

2. **Select Your Desired Cluster**

- On the **Clusters Summary** page, locate and click on the cluster you wish to work with.
- This will display a list of all servers associated with that cluster.

3. **Choose the Server to Shut Down**

- Scroll through the list and select the server you want to shut down.

4. **Initiate the Shutdown Process**

- On the **right-hand side** of the selected server row, click on **Actions**.
- From the dropdown, choose **Shutdown Server**.

5. **Confirm the Shutdown**

- A confirmation prompt will appear. Click **Confirm** to proceed.
- The server will now begin the shutdown process.

Following these steps will safely shut down your chosen server.

Bare Metal: Install Ubuntu 24.04

Below is a comprehensive, step-by-step guide for performing a clean installation of Ubuntu 24.04 as a headless server. This guide covers everything from preparation to post-installation configuration without a desktop environment. **Note:** A clean installation will erase existing data on the target drive. Be sure to back up any important data before proceeding.

1. Pre-Installation Preparations

a. Verify System Requirements

- **Processor:** 2 GHz dual-core or better
- **Memory (RAM):** Minimum 4 GB (more recommended for server workloads)
- **Disk Space:** Minimum 20 GB (adjust based on your intended server usage)
- **Network:** Ethernet or Wi-Fi adapter (wired is preferred for servers)
- **Console Access:** Either direct physical access or remote (via IPMI/KVM) since no GUI will be available

b. Backup Your Data

- **Warning:** A clean installation will **erase all data** on the chosen drive. Back up important files and configurations.
-

2. Download the Ubuntu 24.04 Server ISO

1. Visit the Official Ubuntu Website:

Navigate to [Ubuntu Server Downloads](#).

2. **Select Ubuntu 24.04 LTS:**

Click on the Ubuntu 24.04 LTS Server download button.

3. **Download the ISO:**

Save the ISO file to your computer.

4. **Verify the Download (Optional but Recommended):**

• **On Linux/macOS:**

```
sha256sum path/to/ubuntu-24.04-live-server-amd64.iso
```

• **On Windows:**

Use a tool such as [QuickHash](#) to verify the checksum provided on the Ubuntu website.

3. Create a Bootable USB Drive

a. Using Rufus (Windows)

1. **Download and Install Rufus:**

Visit rufus.ie and download the latest version.

2. **Insert a USB Drive:**

Ensure it's at least 2 GB (all data on the USB will be erased).

3. **Open Rufus and Configure:**

- **Device:** Select your USB drive.
- **Boot selection:** Click "SELECT" and choose the Ubuntu 24.04 Server ISO.
- **Partition scheme:**
 - Choose **GPT** for UEFI systems (most modern PCs).
 - Choose **MBR** if you need BIOS/Legacy support.
- **File System:** Typically "FAT32" (default).

4. **Start the Process:**

Click "START" and confirm any warnings. Rufus will now create your bootable USB drive.

b. Using balenaEtcher (macOS/Linux/Windows)

1. **Download Etcher:**

Go to [balenaEtcher](#) and download the version for your OS.

2. **Install and Run Etcher:**

- Select the Ubuntu 24.04 Server ISO.

- Choose your USB drive.
- Click “Flash!” and wait for the process to finish.

c. Using the `dd` Command (Linux/macOS)

⚠ **Warning:** The `dd` command is powerful. Ensure you’ve selected the correct drive to avoid data loss.

1. Identify the USB Drive:

```
lsblk
```

2. Write the ISO to the USB:

Replace `/dev/sdX` with your USB device (do not include a partition number, e.g., use `/dev/sdb` instead of `/dev/sdb1`):

```
sudo dd if=/path/to/ubuntu-24.04-live-server-amd64.iso of=/dev/sdX bs=4M status=progress  
conv=fdatasync
```

4. Booting from the USB Drive

1. **Insert the Bootable USB Drive** into the server hardware.
2. **Restart Your Server.**
3. **Enter BIOS/UEFI Settings:**
 - Common keys: `F2`, `F12`, `DEL`, or `ESC` (refer to your hardware’s documentation).
 - Set the boot order to prioritize booting from the USB drive.

4. **Save and Exit:**

Your server should now boot from the USB drive.

5. Starting the Ubuntu Server Installation

1. Boot Menu:

After booting from the USB, you’ll see a text-based installer menu. Since this is a server

installation, there is no graphical “Try” mode.

- **Action:** Select “Install Ubuntu Server” (or similar option) using your keyboard’s arrow keys and press **Enter**.
-

6. Follow the Server Installation Wizard

The installation wizard is text-based and will guide you through the setup. Use the keyboard (arrow keys, Tab, and Enter) to navigate the menus.

a. Select Your Language

- **Screen:** “Welcome to Ubuntu Server Installer”
- **Action:** Choose your preferred language and press **Enter**.

b. Select Your Keyboard Layout

- **Screen:** “Keyboard Configuration”
- **Action:** Choose your layout (often “English (US)” or your regional setting) and press **Enter**.

c. Configure the Network

- **Screen:** “Network Connections”
- **Action:**
 - The installer will attempt to configure networking via DHCP.
 - If using a static IP or additional network interfaces, follow the prompts to set up manually.
 - **Tip:** Ensure your network connection is active so that installation can download updates and packages if needed.

d. Configure the Proxy (If Applicable)

- **Screen:** “HTTP Proxy”
- **Action:** Enter your proxy settings if required, or leave blank if not using one.

e. Choose a Mirror for Package Updates

- **Screen:** “Ubuntu Archive Mirror”
- **Action:** Select the appropriate mirror (usually detected automatically) and press **Enter**.

f. Storage Configuration

- **Screen:** “Partition Disks”
- **Options:**
 - **Guided - use entire disk:** For a fully automatic partitioning scheme (this will erase all data).
 - **Manual:** For custom partitioning (for example, creating separate partitions for `/`, `/var`, or swap).
- **Action:** Choose your preferred option, confirm any warnings, and proceed.
 - For UEFI systems, ensure an EFI System Partition ($\approx 300\text{--}500$ MB, FAT32) is created if not already present.
 - Set up the root partition (`/`) (at least 10–20 GB recommended).
 - Optionally configure a swap area (or use a swap file post-installation).

g. Profile Setup

- **Screen:** “Who Are You?”
- **Action:**
 - Enter your full name, a username, and a secure password.
 - Configure whether to enable SSH access during installation (highly recommended for headless servers).
 - **Tip:** Many server installers offer to install and enable the OpenSSH server automatically—confirm this to allow remote management.

h. Feature Selection (Optional)

- **Screen:** “Featured Server Snaps” or “Additional Software”
- **Action:**
 - You might have options to install services like LXD, Docker, or other server applications.
 - Select the ones you need, or skip if you plan to install them later.

i. Finalize the Installation

- **Screen:** “Summary”

- **Action:** Review your configuration and confirm to begin installation.
 - The installer will now copy files and configure your system. This process may take several minutes.
-

7. Finishing Up

1. **Installation Complete:**

Once the process is finished, you'll be prompted to remove the installation media.

2. **Restart Your Server:**

- Remove the USB drive.
- Press **Enter** to reboot.

3. **First Boot:**

- Your server will boot into a command-line login prompt.
 - Log in using the username and password you created during installation.
-

8. Post-Installation Configuration

a. Update the System

1. **Log In via Console or SSH:**

If you enabled the SSH server, you can now connect remotely.

2. **Update Package Lists and Upgrade Packages:**

```
sudo apt update
sudo apt upgrade -y
```

3. **Clean Up Unused Packages:**

```
sudo apt autoremove -y
```

b. Secure Remote Access

- **SSH Configuration:**

The server installer should have set up SSH, but you can further harden it:

- Edit `/etc/ssh/sshd_config` to disable root login and consider changing the default SSH port.

- Restart SSH:

```
sudo systemctl restart ssh
```

- **Firewall Setup with UFW:**

1. **Enable UFW:**

```
sudo ufw allow ssh  
sudo ufw enable
```

2. **Check Firewall Status:**

```
sudo ufw status verbose
```

c. Install Essential Server Packages

- Install utilities and tools that are commonly used:

```
sudo apt install build-essential curl git vim htop -y
```

- **Note:** Since this is a headless server, you won't install any desktop environments.

d. Configure Networking (If Needed)

- Use commands like `ip a` or `ifconfig` to check your network settings.
- For advanced networking, edit configuration files in `/etc/netplan/` (Ubuntu's default on recent releases) and apply changes using:

```
sudo netplan apply
```

e. Set Up Automated Backups

- Consider installing and configuring backup tools (e.g., `rsnapshot` or `borgbackup`) to safeguard your server data.
- Configure a regular backup schedule via cron jobs.

9. Final Thoughts

- **Documentation:** Familiarize yourself with [Ubuntu Server documentation](#) for advanced configuration and troubleshooting.
- **Community:** Ubuntu Forums and Ask Ubuntu are excellent resources for support and tips.
- **Monitoring:** Set up system monitoring tools (e.g., `netdata`, `nagios`, or `prometheus`) to keep an eye on server health.

Congratulations! Your Ubuntu 24.04 headless server is now installed, updated, and configured for remote management and further customization. Enjoy your new server environment!

DigitalOcean: Install Ubuntu 24.04

Below is a step-by-step guide for launching and configuring an Ubuntu 24.04 headless server on DigitalOcean. This tutorial will walk you through creating a Droplet, connecting via SSH, and performing basic post-deployment configuration.

1. Create a DigitalOcean Account (if you haven't already)

1. Sign Up / Log In:

- Visit [DigitalOcean](#) and log in or sign up for an account.

2. Billing & Payment:

- Make sure your billing information is set up so you can create Droplets.
-

2. Create a New Droplet with Ubuntu 24.04

1. Navigate to Droplets:

- In the DigitalOcean control panel, click **Create** and select **Droplets**.

2. Choose an Image:

- Under “Distributions”, find and select **Ubuntu 24.04 LTS**.

“Note: DigitalOcean’s images are headless by default (no GUI).”

3. Select a Plan:

- Choose the Droplet size that best fits your needs (e.g., Basic with 1-2 GB of RAM for lightweight tasks, or larger for production workloads).

4. Choose a Datacenter Region:

- Select the region closest to your target audience for better latency.
5. **Authentication:**
 - **SSH Keys (Recommended):**
 - Add your SSH public key to securely access the Droplet.
 - **Password Authentication:**
 - Alternatively, choose to use a strong password (less secure; not recommended if you can use SSH keys).
 6. **Additional Options:**
 - You may optionally enable backups, monitoring, and IPv6 if needed.
 - Tag your Droplet for easier organization.
 7. **Finalize and Create:**
 - Click **Create Droplet**.
 - After a few moments, your Droplet will be ready. You'll see its public IP address on the DigitalOcean dashboard.
-

3. Connect to Your Droplet via SSH

1. **Open a Terminal:**
 - On Linux/macOS, open your terminal.
 - On Windows, use an SSH client like PuTTY or the built-in Windows Terminal (if using OpenSSH).
 2. **Connect to the Droplet:**
 - Run the following command, replacing `username` (usually `root` on first login) and `<droplet_ip>` with your Droplet's IP address:

```
ssh root@<droplet_ip>
```
 - If using an SSH key, ensure your key is loaded (using `ssh-agent` or specifying `-i path/to/your/key`).
 3. **Accept the Fingerprint:**
 - When prompted, type **yes** to continue connecting.
-

4. Initial Post-Installation Setup

Once connected, perform these basic steps to secure and update your server.

a. Update and Upgrade Packages

1. **Update Package Lists:**

```
apt update
```

2. **Upgrade Packages:**

```
apt upgrade -y
```

b. Create a Non-Root User (Optional but Recommended)

1. **Create a User:**

```
adduser yourusername
```

2. **Grant Sudo Privileges:**

```
usermod -aG sudo yourusername
```

3. **Test by Logging in as Your New User:**

- Log out of the root session:

```
exit
```

- Log in again using:

```
ssh yourusername@<droplet_ip>
```

c. Secure SSH Access

1. **Edit the SSH Configuration:**

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

2. **Recommended Changes:**

- Disable root login (if not needed):

```
PermitRootLogin no
```

- (Optional) Change the default SSH port (e.g., to 2222):

```
Port 2222
```

- Save and exit the editor (in nano, press `Ctrl+X`, then `Y`, then `Enter`).

3. Restart SSH Service:

```
sudo systemctl restart ssh
```

“Note: If you change the SSH port, connect with:
`ssh -p 2222 yourusername@<droplet_ip>`

d. Configure the UFW Firewall

1. Allow SSH Connections:

```
sudo ufw allow ssh
```

- If you changed your SSH port, use:

```
sudo ufw allow 2222/tcp
```

2. Enable UFW:

```
sudo ufw enable
```

3. Check Firewall Status:

```
sudo ufw status verbose
```

5. Install Essential Server Packages

Depending on your intended use, install additional tools. Here are some common utilities:

```
sudo apt install build-essential curl git vim htop -y
```

You can install other services (e.g., web server, database, etc.) as needed.

6. Additional DigitalOcean Features

DigitalOcean offers a range of additional features to manage your server:

- **Snapshots and Backups:**

Consider enabling regular backups or taking snapshots via the DigitalOcean control panel.

- **Monitoring:**

Use DigitalOcean Monitoring to track your Droplet's performance.

- **Cloud Firewalls:**

Optionally configure DigitalOcean Cloud Firewalls from the control panel for an extra layer of network security.

7. Final Thoughts

- **Documentation:**

Check out [DigitalOcean's tutorials](#) and [Ubuntu Server documentation](#) for advanced configurations.

- **Security:**

Regularly update your server and monitor logs to keep your server secure.

Congratulations! Your Ubuntu 24.04 headless server on DigitalOcean is now set up and ready for further customization and deployment of your applications. Enjoy your new DigitalOcean Droplet!

VMWare: Install Ubuntu 24.04

Below is a comprehensive, step-by-step guide for deploying an Ubuntu 24.04 headless server as a virtual machine in VMware vCenter (version 7 or 8). This tutorial covers everything from obtaining the ISO and uploading it to your datastore, creating a new VM, performing the installation, and carrying out basic post-deployment configuration.

// Important:

- Ensure you have administrative privileges in vCenter.
- Back up any important data before making changes.
- A headless server does not include a desktop GUI.

1. Pre-Deployment Preparations

a. Verify Requirements

- **vCenter Access:** VMware vCenter 7/8 environment with rights to create VMs.
- **Hardware Resources:** Ensure the host/cluster has enough CPU, RAM, and storage for your VM.
- **ISO Image:** Download the Ubuntu 24.04 Server ISO from [Ubuntu Server Downloads](#).

b. Upload the Ubuntu ISO to a Datastore

1. Log In to vCenter:

Open the vSphere Client and log in to your vCenter instance.

2. Select a Datastore:

Navigate to **Storage** and choose the datastore where you'd like to store the ISO.

3. Upload the ISO:

- Right-click the datastore and select **Upload Files**.
- Browse for the downloaded `ubuntu-24.04-live-server-amd64.iso` file and upload it.

- Note the datastore path for later use.
-

2. Create a New Virtual Machine in vCenter

a. Launch the New VM Wizard

1. Right-Click Host/Cluster:

In the vSphere Client inventory, right-click your desired host or cluster and select **New Virtual Machine**.

2. Choose Creation Method:

- Select **Create a new virtual machine** and click **Next**.

b. Configure VM Settings

1. Name and Location:

- **Name:** Enter a meaningful name (e.g., `Ubuntu24-Server`).
- **Folder/Resource Pool:** Choose the appropriate folder or resource pool.

2. Select a Compute Resource:

- Choose the host or cluster where the VM will run.

3. Select a Storage:

- Choose the datastore where the VM's virtual disk will reside.

4. Compatibility & Guest OS:

- **Compatibility:** Use the default or choose one that fits your environment.
- **Guest OS:**
 - **Family:** Linux
 - **Version:** Ubuntu Linux (64-bit)

“Note: Ubuntu 24.04 may not be explicitly listed; select the closest match (Ubuntu 64-bit).”

5. Configure Virtual Hardware:

- **CPU:** Allocate at least 1-2 vCPUs.
- **Memory:** Allocate at least 2 GB (more if your workloads require it).
- **Disk:** Create a virtual disk of 20 GB or more (adjust based on your needs).
- **Network:** Add a network adapter (VMXNET3 is recommended) and ensure it's connected.
- **CD/DVD Drive:**

- Select **Datastore ISO File**.
- Browse to the location where you uploaded the Ubuntu 24.04 ISO.
- Check the option **Connect At Power On**.

6. **Finish the Wizard:**

Review your settings and click **Finish** to create the VM.

3. Booting and Installing Ubuntu 24.04 Server

a. Power On the VM and Open the Console

1. **Start the VM:**

Right-click the newly created VM and select **Power On**.

2. **Launch Console:**

Open the VM console from the vSphere Client to interact with the installation.

b. Boot from the ISO and Begin Installation

1. **Boot Menu:**

- The VM should boot from the ISO automatically. If not, enter the VM's BIOS/UEFI (using the console's key prompts) and adjust the boot order to boot from the CD/DVD drive.

2. **Select "Install Ubuntu Server":**

- At the text-based installer menu, select **Install Ubuntu Server** and press **Enter**.

c. Follow the Text-Based Installation Wizard

1. **Language and Keyboard:**

- **Select Language:** Choose your preferred language.
- **Keyboard Layout:** Confirm or adjust the layout.

2. **Network Configuration:**

- The installer will attempt to configure networking via DHCP.
- If you need a static IP, follow the prompts to set it manually.

3. **Storage Configuration:**

- **Partitioning Options:**

- Choose **Guided - use entire disk** for simplicity (this will erase any data on the virtual disk).
 - Optionally, choose **Manual** if you wish to customize partitions.
 - **EFI Partition:**
 - For UEFI-based VMs, ensure an EFI System Partition is created (usually around 300–500 MB, formatted as FAT32).
 - 4. **User and SSH Setup:**
 - **Profile Setup:**
 - Enter your full name, username, and a secure password.
 - **SSH Server:**
 - When prompted, choose to install and enable the OpenSSH server. This is essential for remote management of your headless server.
 - 5. **Finalizing Installation:**
 - Review your configuration on the summary screen and confirm to begin the installation.
 - The installer will copy files and configure the system. This may take several minutes.
 - 6. **Complete Installation:**
 - Once the installation is complete, you'll be prompted to remove the installation media.
 - In vCenter, edit the VM's settings to disconnect the ISO from the CD/DVD drive.
 - Reboot the VM.
-

4. Post-Installation Configuration

a. Log In to Your New Server

- **Console or SSH:**

Use the VM console or connect via SSH (if configured during installation) using:

```
ssh yourusername@<vm_ip_address>
```

b. Update and Upgrade Packages

1. **Update Package Lists:**

```
sudo apt update
```

2. **Upgrade Packages:**

```
sudo apt upgrade -y
```

c. Secure Your Server

1. Disable Root SSH Login (if not already done):

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

- Find the line `PermitRootLogin` and set it to `no`.
- Save and exit (Ctrl+X, then Y, then Enter).

2. Restart SSH Service:

```
sudo systemctl restart ssh
```

d. Configure the Firewall with UFW

1. Allow SSH Connections:

```
sudo ufw allow ssh
```

- If you changed the SSH port, allow that port instead (e.g., `sudo ufw allow 2222/tcp`).

2. Enable UFW:

```
sudo ufw enable
```

3. Check Firewall Status:

```
sudo ufw status verbose
```

e. Install Essential Tools

- For basic system administration, install packages such as:

```
sudo apt install build-essential curl git vim htop -y
```

5. Final Thoughts

- **Snapshots:**

Use vCenter's snapshot feature to create a restore point after configuring your server.

- **Monitoring & Backups:**

Consider setting up monitoring tools and regular backups to safeguard your VM.

- **Further Customization:**

Explore additional packages and configurations based on your server's intended role.

Your Ubuntu 24.04 headless server is now deployed on VMware vCenter and ready for further customization and application deployment. Enjoy your new virtual server environment!