

A Private Git Server on Ubuntu

Objective:

Deploy a self-hosted Git server using Gitea on an Ubuntu machine.

This project involves setting up a private Git server on an AWS EC2 Linux instance with secure SSH authentication, user access control, and repository management, enabling a secure and private environment for team-based software development and code collaboration. The server will feature a graphical user interface (GUI) for ease of use and administration. Additionally, a custom domain (saurabhcloud.fun) will be configured to provide easy access, and an optional Gitea web interface will be integrated for visual repository management. For added security, SSL encryption using Let's Encrypt will be implemented to ensure encrypted access over HTTPS.

Tools & Technologies:

- AWS EC2
- Ubuntu 22.04 LTS
- Git
- Gitea (Optional GUI)
- Nginx
- Certbot (SSL)
- Custom Domain: saurabhcloud.fun

Part 1: Launch and Connect to EC2 Instance

1. Launch EC2 Instance:

- AMI: Ubuntu 22.04 LTS
- Instance Type: t2.medium
- Key Pair: Use an existing or create a new one
- Security Group:
 - Allow:
 - 22 (SSH)
 - 80 (HTTP)
 - 443 (HTTPS)
 - 587 (Custom TCP – for email)
 - 3000–11000 (Custom TCP – for web apps like Gitea)

Assign EC2 Instance Name

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The top navigation bar includes links for Launch an instance, EC2, S3, IAM, VPC, and Route 53. The main content area has sections for 'Name and tags' (with 'git-server' entered), 'Application and OS Images (Amazon Machine Image)' (with a search bar and a list of AMIs including Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and SUSE Linux), and 'Summary' (which lists the number of instances, software image (Amazon Linux 2023 AMI 2023.7.2), virtual server type (t2.micro), firewall (New security group), storage (1 volume(s) - 8 GiB), and provides 'Cancel', 'Launch instance', and 'Preview code' buttons). The bottom navigation bar includes CloudShell, Feedback, and various application icons.

Select AMI According to Your configuration .

The screenshot shows the 'Launch an instance' wizard with the 'Quick Start' tab selected. It displays a grid of AMI icons for Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and SUSE. A 'Browse more AMIs' button is available. Below the grid, the 'Amazon Machine Image (AMI)' section details the selected Ubuntu Server 24.04 LTS (HVM), SSD Volume Type (ami-04f167a56786e4b09 / ami-0ae6f07ad3a8ef182 (64-bit (Arm))), and indicates it is 'Free tier eligible'. The right side of the screen shows the same 'Summary' section as the previous screenshot, including the 'Launch instance' button. The bottom navigation bar is identical to the first screenshot.

Select the Instance Type According to Your configuration .

The screenshot shows the AWS EC2 'Launch an instance' wizard. In the 'Instance type' section, 't2.medium' is selected. Below it, a detailed description of the instance type is provided, including its family (t2), vCPUs (2), memory (4 GiB), and current generation status. A note states that additional costs apply for AMIs with pre-installed software. In the 'Key pair (login)' section, there is a dropdown menu for selecting a key pair, with 'Select' currently chosen. On the right side, the 'Summary' panel shows the configuration: 1 instance, AMI: Canonical, Ubuntu, 24.04, amd64, Virtual server type: t2.medium, and Storage (volumes): 1 volume(s) - 8 GiB. At the bottom right are 'Cancel', 'Launch instance', and 'Preview code' buttons.

Create key Pair .

The screenshot shows the 'Create key pair' dialog box overlaid on the EC2 instance launch wizard. In the 'Key pair name' field, 'git-server-access-key' is entered. The 'Key pair type' section shows two options: 'RSA' (selected) and 'ED25519'. Under 'Private key file format', the radio button for '.pem' is selected. A warning message at the bottom of the dialog box reads: '⚠️ When prompted, store the private key in a secure and accessible location on'. The background shows the same EC2 instance configuration details as the previous screenshot, including the summary and launch buttons.

Create Security Group or Use existing Security Group

(But only check it which port is allocated if already exist Security Group .)

The screenshot shows the 'Create security group' page in the AWS EC2 console. In the 'Basic details' section, the 'Security group name' is set to 'Primary-SG'. The 'Description' is 'Allow all traffic'. The 'VPC' dropdown is set to 'vpc-02ccf3089f48a3687'. Under 'Inbound rules', it says 'This security group has no inbound rules.' and there is a blue 'Add rule' button.

If it is not allocated. Then allocate this ports .

The screenshot shows the 'sg-06e69bcaca80d4b52 - Primary-SG' page. It displays 5 inbound rules and 1 outbound rule. The inbound rules table is as follows:

Security group rule ID	IP version	Type	Protocol	Port range	Source
sgr-0f1441821dfa444f	IPv4	Custom TCP	TCP	3000 - 11000	0.0.0.0/0
sgr-057f618eed8e7fdd5	IPv4	HTTPS	TCP	443	0.0.0.0/0
sgr-02020c76b7568f5ab	IPv4	SSH	TCP	22	0.0.0.0/0
sgr-0478181f7fec1e519	IPv4	HTTP	TCP	80	0.0.0.0/0
sgr-0e644c29dc6093c69	IPv4	Custom TCP	TCP	587	0.0.0.0/0

The screenshot shows the same 'sg-06e69bcaca80d4b52 - Primary-SG' page, specifically the 'Inbound rules' section. The table of inbound rules is identical to the one shown in the previous screenshot.

Now select your Security Group .

The screenshot shows the AWS EC2 'Launch an instance' wizard. In the 'Security Group' section, the 'Select existing security group' button is selected, pointing to 'Primary-SG'. The 'Configure storage' section shows a root volume of 8 GiB (gp3). On the right, the 'Summary' panel indicates 1 instance, using the Canonical, Ubuntu 24.04 AMI, and a t2.medium instance type. The 'Launch instance' button is prominent.

Configure the storage of your Machine .

The screenshot shows the AWS EC2 'Launch an instance' wizard. In the 'Configure storage' section, the root volume size is increased to 25 GiB (gp3). A message box informs users about free tier storage limits. The 'Summary' panel on the right shows 1 instance, using the Canonical, Ubuntu 24.04 AMI, and a t2.medium instance type. The 'Launch instance' button is present.

Launch Your Ec2 Instance and Now its Started Successfully.

And Copy Public Ip for Remote Access.

The screenshot shows the AWS Management Console with the EC2 Instances page open. On the left, there's a navigation sidebar with sections like EC2, Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, and AMI Catalog. The main area displays a table of instances with one row selected: "git-server" (Instance ID: i-066a5c003c1ad7fb6). A tooltip is overlaid on the "Public IPv4 address" field, which contains "18.220.19.210 | open address". Other columns in the table include Instance state (Running), Instance type (t2.medium), Status check (Initializing), and Alarm status (View alarms +). At the top right, there are buttons for Actions (Launch instances) and Connect. The bottom right corner shows the AWS logo and the date/time: 08-05-2025, 01:44 PM.

Connect to EC2 via SSH and Upload your Private Key.

The screenshot shows the MobaXterm application window. The title bar says "18.220.19.210 (ec2-user)". The main interface includes a toolbar with icons for Session, Servers, Tools, Games, Sessions, View, X server, and Help. On the left, there's a sidebar with "User sessions" showing a connection to "3.17.162.110 (ec2-user)". The central part of the window is the "Session settings" dialog, which is expanded. It has tabs for SSH, Telnet, Rsh, Xdmcp, RDP, VNC, FTP, SFTP, Serial, File, Shell, Browser, Mosh, Aws S3, and WSL. The "SSH" tab is selected. Under "Basic SSH settings", the "Remote host" is set to "18.220.19.210", "Specify username" is set to "ubuntu", and the "Port" is set to "22". Under "Advanced SSH settings", there are checkboxes for "X11-Forwarding", "Compression", "Execute command", "SSH-browser type" (set to "SFTP protocol"), "Use private key" (with a file path "C:\Users\Lenovo\Downloads\git-"), and "Expert SSH settings". There are also fields for "Execute macro at session start" and "SSH key icon". At the bottom of the dialog are "OK" and "Cancel" buttons. The bottom of the screen shows the Windows taskbar with various pinned icons and the date/time: 08-05-2025, 01:13 PM.

Now Successfully getting Connect to EC2 via SSH.

Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1024-aws x86_64)

```
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/pro

System information as of Thu May 8 07:44:00 UTC 2025

System load: 0.01      Processes: 125
Usage of /: 7.2% of 23.17GB  Users logged in: 0
Memory usage: 5%      IPv4 address for enX0: 172.31.11.120
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

/usr/bin/xauth: file /home/ubuntu/.Xauthority does not exist
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-11-120:~$
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

01:14 PM
08-05-2025

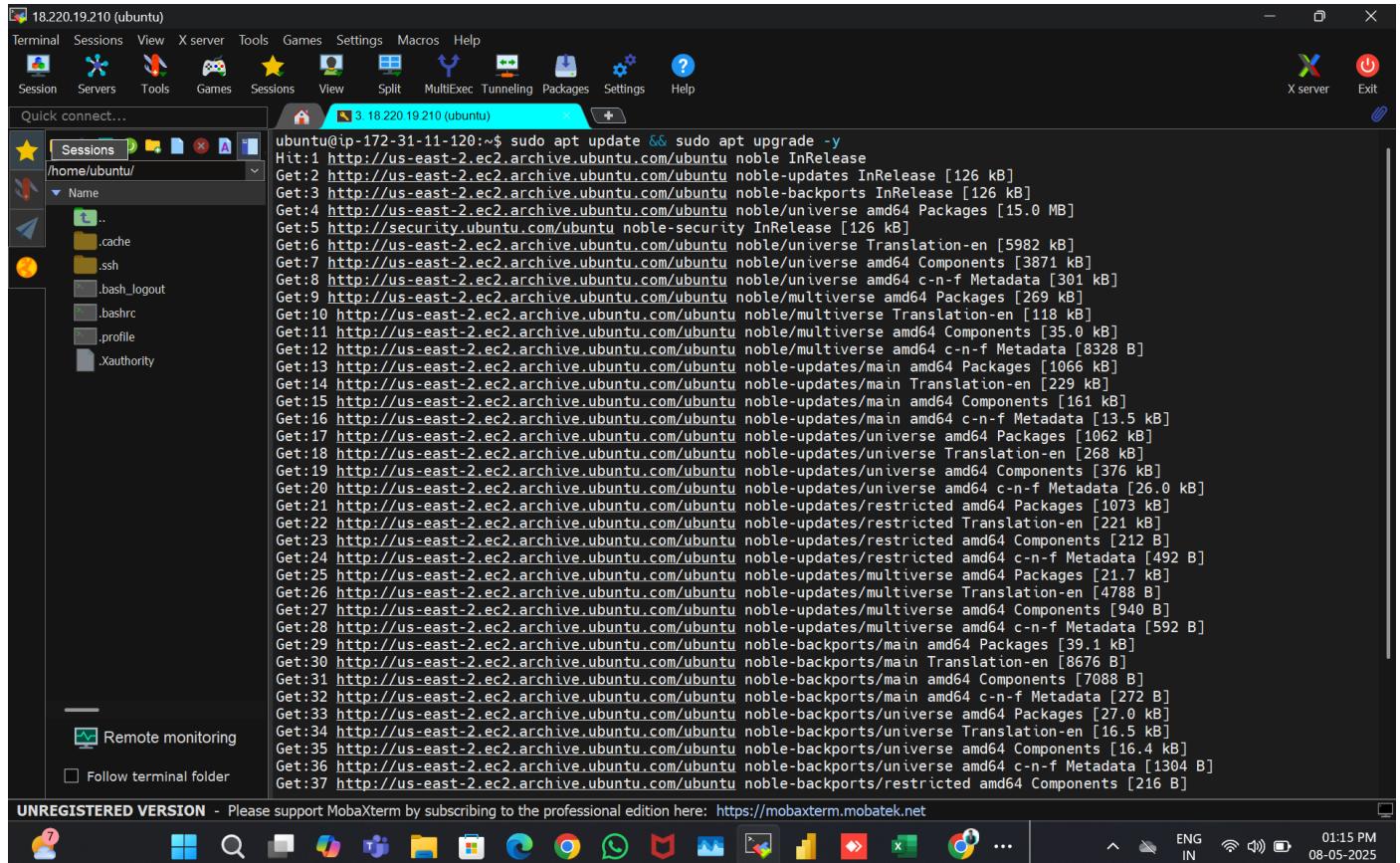
Part 2: Install GUI (Desktop Environment)

Install the Ubuntu desktop environment and enable RDP access using XRDP to manage the server with a graphical interface.

Step-by-Step Implementation and Following this Commands .

1 . Install Ubuntu & Update System

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



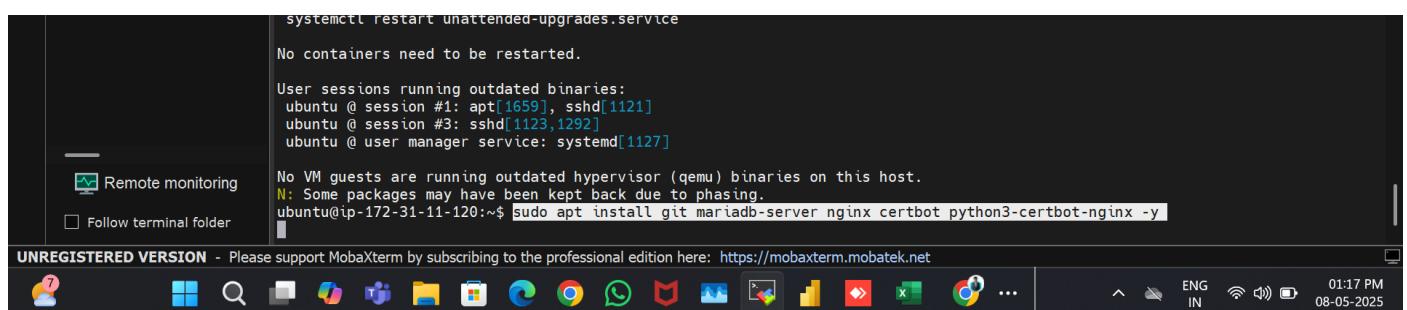
The screenshot shows a MobaXterm session titled "18.220.19.210 (ubuntu)". The terminal window displays the command output:

```
ubuntu@ip-172-31-11-120:~$ sudo apt update && sudo apt upgrade -y
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:5 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:6 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:8 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:9 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:10 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:11 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:12 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 kB]
Get:13 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [1066 kB]
Get:14 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [229 kB]
Get:15 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [161 kB]
Get:16 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [13.5 kB]
Get:17 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1062 kB]
Get:18 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [268 kB]
Get:19 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [376 kB]
Get:20 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [26.0 kB]
Get:21 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [1073 kB]
Get:22 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted Translation-en [221 kB]
Get:23 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Components [212 B]
Get:24 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 c-n-f Metadata [492 B]
Get:25 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [21.7 kB]
Get:26 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse Translation-en [4788 B]
Get:27 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [940 B]
Get:28 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 c-n-f Metadata [592 B]
Get:29 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Packages [39.1 kB]
Get:30 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-backports/main Translation-en [8676 B]
Get:31 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [7088 B]
Get:32 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 c-n-f Metadata [272 B]
Get:33 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [27.0 kB]
Get:34 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-backports/universe Translation-en [16.5 kB]
Get:35 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [16.4 kB]
Get:36 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 c-n-f Metadata [1304 B]
Get:37 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [216 B]
```

At the bottom of the terminal window, there is a message: "UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>".

2. Install Required Packages

```
sudo apt install git mariadb-server nginx certbot python3-certbot-nginx -y
```



The screenshot shows a MobaXterm session titled "18.220.19.210 (ubuntu)". The terminal window displays the command output:

```
systemctl restart unattended-upgrades.service
No containers need to be restarted.

User sessions running outdated binaries:
  ubuntu @ session #1: apt[1659], sshd[1121]
  ubuntu @ session #3: sshd[1123,1292]
  ubuntu @ user manager service: systemd[1127]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
N: Some packages may have been kept back due to phasing.
ubuntu@ip-172-31-11-120:~$ sudo apt install git mariadb-server nginx certbot python3-certbot-nginx -y
```

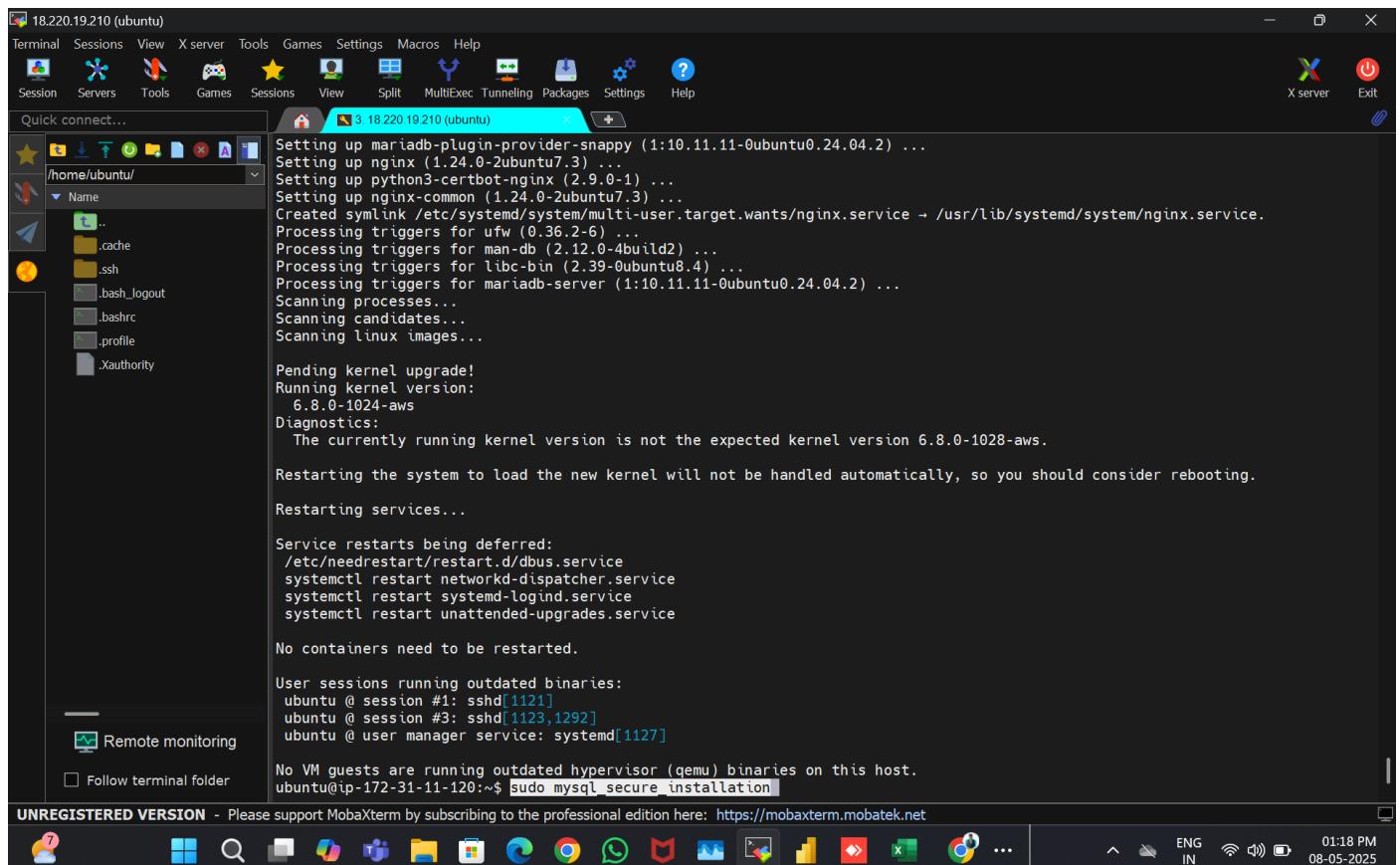
At the bottom of the terminal window, there is a message: "UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>".

3 . Secure MariaDB

sudo mysql_secure_installation

Enhance the security of your MariaDB server by running sudo mysql_secure_installation. This guided setup allows you to:

- **Set a strong root password** to prevent unauthorized access.
- **Remove anonymous users** who could pose a security risk.
- **Disable remote root login** to ensure the root account can only be accessed locally, reducing exposure to external threats.



```
Setting up mariadb-plugin-provider-snappy (1:10.11.11-0ubuntu0.24.04.2) ...
Setting up nginx (1.24.0-2ubuntu7.3) ...
Setting up python3-cerbot-nginx (2.9.0-1) ...
Setting up nginx-common (1.24.0-2ubuntu7.3) ...
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /usr/lib/systemd/system/nginx.service.
Processing triggers for ufw (0.36.2-6) ...
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.4) ...
Processing triggers for mariadb-server (1:10.11.11-0ubuntu0.24.04.2) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Pending kernel upgrade!
Running kernel version:
  6.8.0-1024-aws
Diagnostics:
  The currently running kernel version is not the expected kernel version 6.8.0-1028-aws.

Restarting the system to load the new kernel will not be handled automatically, so you should consider rebooting.

Restarting services...

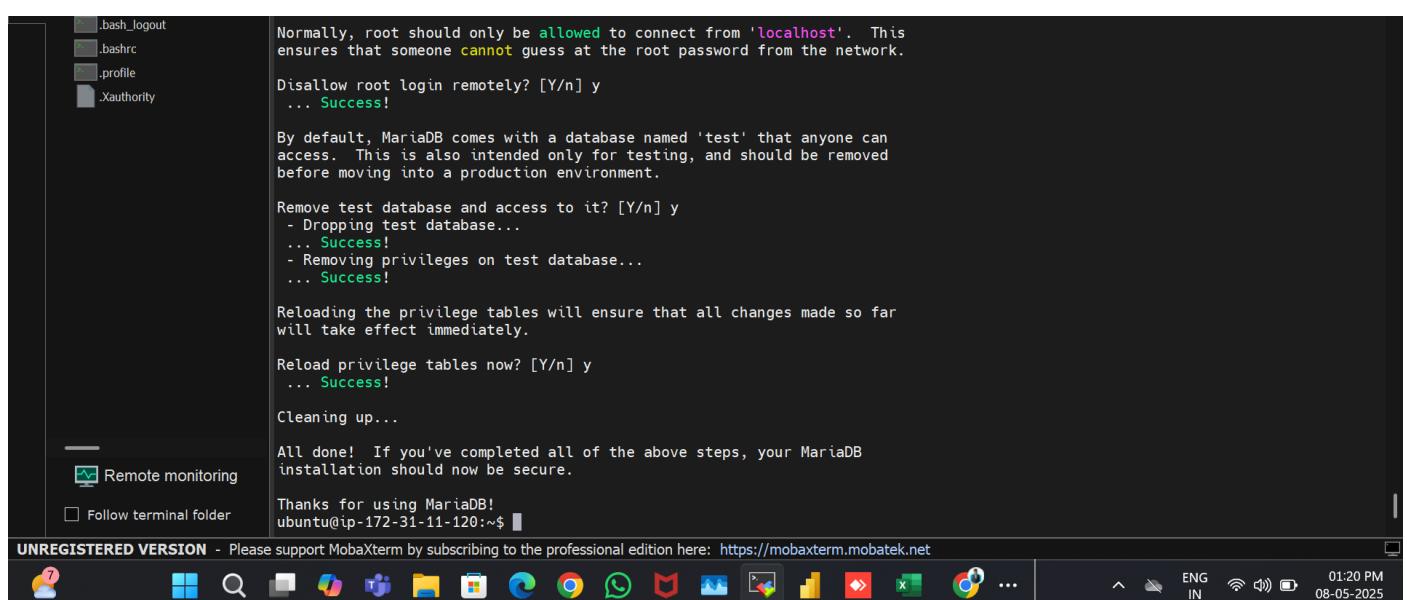
Service restarts being deferred:
/etc/needrestart/restart.d/dbus.service
systemctl restart networkd-dispatcher.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
  ubuntu @ session #1: sshd[1121]
  ubuntu @ session #3: sshd[1123,1292]
  ubuntu @ user manager service: systemd[1127]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-11-120:~$ sudo mysql_secure_installation

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net
```



```
... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
ubuntu@ip-172-31-11-120:~$
```

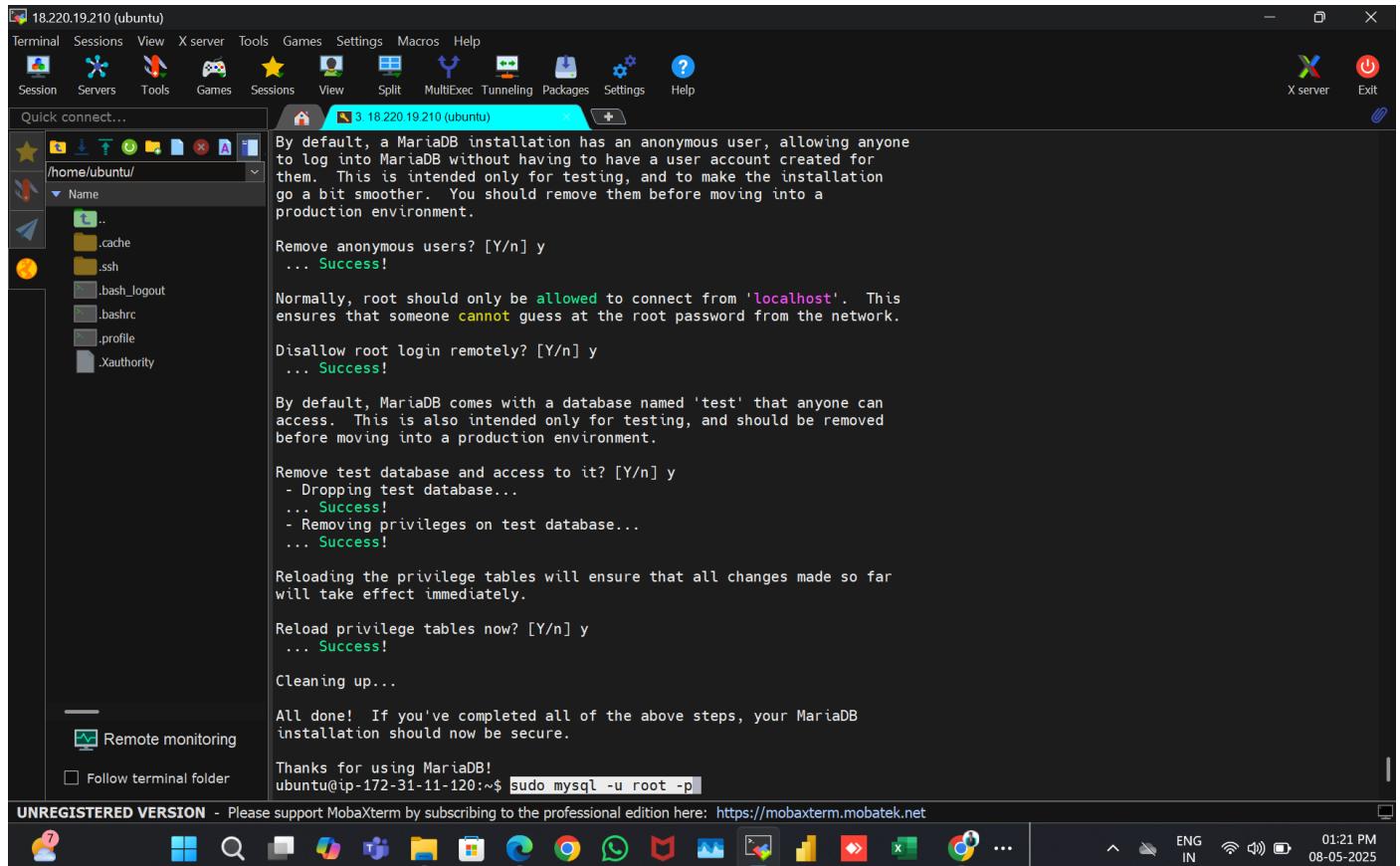
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

01:20 PM
08-05-2025

4. Create Gitea Database & User

sudo mysql -u root -p

Log in to MariaDB using `sudo mysql -u root -p`, then create a dedicated **Gitea database and user** with appropriate privileges to manage repositories and user data.



The screenshot shows a terminal window titled "18.220.19.210 (ubuntu)" running on MobaXterm. The terminal displays the following MySQL setup commands:

```
By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y
... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
ubuntu@ip-172-31-11-120:~$ sudo mysql -u root -p
```

The terminal window includes a sidebar with session management tools like Session, Servers, Tools, Games, and Sessions. The status bar at the bottom shows the date and time: "08-05-2025 01:21 PM".

Access MariaDB and run the following commands to create the Gitea database and user with full privileges :

Run:

```
CREATE DATABASE gitea CHARACTER SET 'utf8mb4' COLLATE 'utf8mb4_unicode_ci';
```

```
CREATE USER 'gitea'@'localhost' IDENTIFIED BY 'Saurabh@29';
```

```
GRANT ALL PRIVILEGES ON gitea.* TO 'gitea'@'localhost';
```

```
FLUSH PRIVILEGES;
```

```
EXIT;
```

```

18.220.19.210 (ubuntu)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
3 18.220.19.210 (ubuntu) + +
Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
ubuntu@ip-172-31-11-120:~$ sudo mysql -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 41
Server version: 10.11.11-MariaDB-0ubuntu0.24.04.2 Ubuntu 24.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE gitea CHARACTER SET 'utf8mb4' COLLATE 'utf8mb4_unicode_ci';
Query OK, 1 row affected (0.001 sec)

MariaDB [(none)]> CREATE USER 'gitea'@'localhost' IDENTIFIED BY 'Saurabh@29';
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON gitea.* TO 'gitea'@'localhost';
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> EXIT;
Bye
ubuntu@ip-172-31-11-120:~$ 

```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

5 . Install & Configure Gitea

Download & Install Gitea

Create necessary directories and a system user for Gitea, then download the Gitea binary :

```
sudo mkdir -p /var/lib/gitea
```

```
sudo useradd --system --home /var/lib/gitea --shell /bin/bash gitea
```

```
sudo wget -O /usr/local/bin/gitea https://dl.gitea.com/gitea/1.23.4/gitea-1.23.4-linux-amd64
```

```
sudo chmod +x /usr/local/bin/gitea
```

```

Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> EXIT;
Bye
ubuntu@ip-172-31-11-120:~$ sudo mkdir -p /var/lib/gitea
sudo useradd --system --home /var/lib/gitea --shell /bin/bash gitea
sudo wget -O /usr/local/bin/gitea https://dl.gitea.com/gitea/1.23.4/gitea-1.23.4-linux-amd64
sudo chmod +x /usr/local/bin/gitea
--2025-05-08 07:54:45- https://dl.gitea.com/gitea/1.23.4/gitea-1.23.4-linux-amd64
Resolving dl.gitea.com (dl.gitea.com)... 18.238.25.36, 18.238.25.102, 18.238.25.61, ...
Connecting to dl.gitea.com (dl.gitea.com)|18.238.25.36|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 114113872 (109M) [binary/octet-stream]
Saving to: '/usr/local/bin/gitea'

/usr/local/bin/gitea          100%[=====] 108.83M  36.5MB/s   in 3.0s

2025-05-08 07:54:48 (36.5 MB/s) - '/usr/local/bin/gitea' saved [114113872/114113872]
ubuntu@ip-172-31-11-120:~$ 

```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Create Configuration & Data Directories.

Set up Gitea's required directories, assign proper ownership to the gitea user, and secure the configuration file :

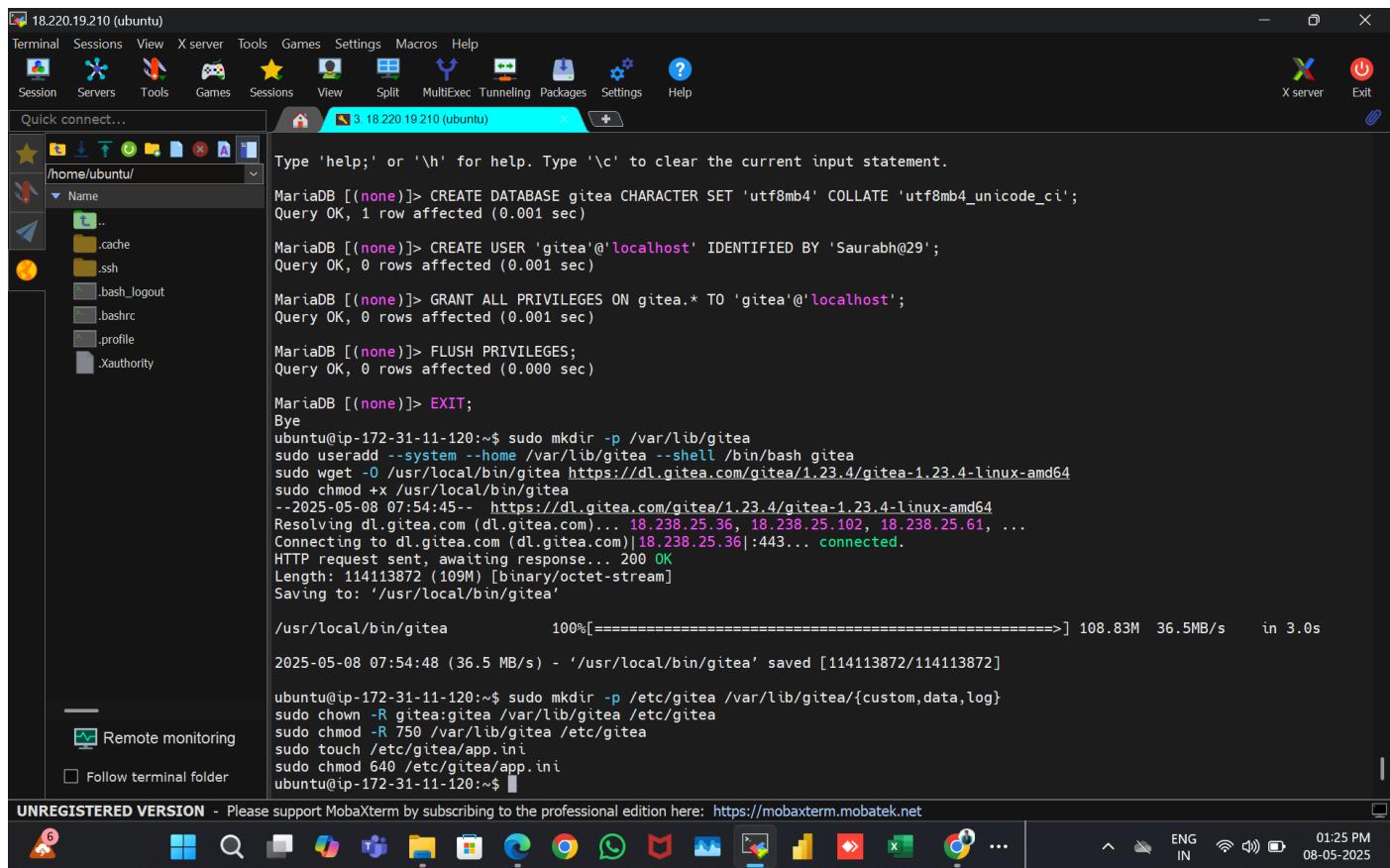
```
sudo mkdir -p /etc/gitea /var/lib/gitea/{custom,data,log}
```

```
sudo chown -R gitea:gitea /var/lib/gitea /etc/gitea
```

```
sudo chmod -R 750 /var/lib/gitea /etc/gitea
```

```
sudo touch /etc/gitea/app.ini
```

```
sudo chmod 640 /etc/gitea/app.ini
```



The screenshot shows a MobaXterm session titled "18.220.19.210 (ubuntu)". The terminal window displays the following commands and their output:

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
MariaDB [(none)]> CREATE DATABASE gitea CHARACTER SET 'utf8mb4' COLLATE 'utf8mb4_unicode_ci';  
Query OK, 1 row affected (0.001 sec)  
MariaDB [(none)]> CREATE USER 'gitea'@'localhost' IDENTIFIED BY 'Saurabh@29';  
Query OK, 0 rows affected (0.001 sec)  
MariaDB [(none)]> GRANT ALL PRIVILEGES ON gitea.* TO 'gitea'@'localhost';  
Query OK, 0 rows affected (0.001 sec)  
MariaDB [(none)]> FLUSH PRIVILEGES;  
Query OK, 0 rows affected (0.000 sec)  
MariaDB [(none)]> EXIT;  
Bye  
ubuntu@ip-172-31-11-120:~$ sudo mkdir -p /var/lib/gitea  
sudo useradd --system --home /var/lib/gitea --shell /bin/bash gitea  
sudo wget -O /usr/local/bin/gitea https://dl.gitea.com/gitea/1.23.4/gitea-1.23.4-linux-amd64  
sudo chmod +x /usr/local/bin/gitea  
--2025-05-08 07:54:45- https://dl.gitea.com/gitea/1.23.4/gitea-1.23.4-linux-amd64  
Resolving dl.gitea.com (dl.gitea.com)... 18.238.25.36, 18.238.25.102, 18.238.25.61, ...  
Connecting to dl.gitea.com (dl.gitea.com)|18.238.25.36|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 114113872 (109M) [binary/octet-stream]  
Saving to: '/usr/local/bin/gitea'  
  
/usr/local/bin/gitea          100%[=====] 108.83M  36.5MB/s   in 3.0s  
2025-05-08 07:54:48 (36.5 MB/s) - '/usr/local/bin/gitea' saved [114113872/114113872]  
  
ubuntu@ip-172-31-11-120:~$ sudo mkdir -p /etc/gitea /var/lib/gitea/{custom,data,log}  
sudo chown -R gitea:gitea /var/lib/gitea /etc/gitea  
sudo chmod -R 750 /var/lib/gitea /etc/gitea  
sudo touch /etc/gitea/app.ini  
sudo chmod 640 /etc/gitea/app.ini  
ubuntu@ip-172-31-11-120:~$
```

The terminal window also shows a file browser sidebar on the left and a taskbar at the bottom with various icons.

6 . Configure app.ini

Edit the Gitea configuration file to define repository paths, server settings, and logging details :

Open the Gitea configuration file :

```
sudo vi /etc/gitea/app.ini
```

MariaDB [(none)]> CREATE DATABASE gitea CHARACTER SET 'utf8mb4' COLLATE 'utf8mb4_unicode_ci';
Query OK, 1 row affected (0.001 sec)
MariaDB [(none)]> CREATE USER 'gitea'@'localhost' IDENTIFIED BY 'Saurabh@29';
Query OK, 0 rows affected (0.001 sec)
MariaDB [(none)]> GRANT ALL PRIVILEGES ON gitea.* TO 'gitea'@'localhost';
Query OK, 0 rows affected (0.001 sec)
MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.000 sec)
MariaDB [(none)]> EXIT;
Bye
ubuntu@ip-172-31-11-120:~\$ sudo mkdir -p /var/lib/gitea
sudo useradd --system --home /var/lib/gitea --shell /bin/bash gitea
sudo wget -O /usr/local/bin/gitea https://dl.gitea.com/gitea/1.23.4/gitea-1.23.4-linux-amd64
sudo chmod +x /usr/local/bin/gitea
--2025-05-08 07:54:45-- https://dl.gitea.com/gitea/1.23.4/gitea-1.23.4-linux-amd64
Resolving dl.gitea.com (dl.gitea.com)... 18.238.25.36, 18.238.25.102, 18.238.25.61, ...
Connecting to dl.gitea.com (dl.gitea.com)|18.238.25.36|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 114113872 (109M) [binary/octet-stream]
Saving to: '/usr/local/bin/gitea'

/usr/local/bin/gitea 100%[=====] 108.83M 36.5MB/s in 3.0s
2025-05-08 07:54:48 (36.5 MB/s) - '/usr/local/bin/gitea' saved [114113872/114113872]

ubuntu@ip-172-31-11-120:~\$ sudo mkdir -p /etc/gitea /var/lib/gitea/{custom,data,log}
sudo chown -R gitea:gitea /var/lib/gitea /etc/gitea
sudo chmod -R 750 /var/lib/gitea /etc/gitea
sudo touch /etc/gitea/app.ini
sudo chmod 640 /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~\$ sudo vi /etc/gitea/app.ini
[1]+ Stopped sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~\$ sudo vi /etc/gitea/app.ini

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Update Repository Paths

[repository]

ROOT = /var/lib/gitea/data/gitea-repositories

Update Server Configuration

[server]

APP_DATA_PATH = /var/lib/gitea/data

DOMAIN = git.saurabhccloud.fun

SSH_DOMAIN = git.saurabhccloud.fun

HTTP_PORT = 3000

ROOT_URL = https://git.saurabhccloud.fun/

LFS_CONTENT_PATH = /var/lib/gitea/data/fs

Update Log Configuration

[log]

ROOT_PATH = /var/lib/gitea/log

The screenshot shows the MobaXterm interface with a terminal window open to the file `/etc/gitea/app.ini`. The file contains configuration settings for Gitea, including the repository root, server port, and log path. The terminal window has tabs for session 3 and 4, and a status bar at the bottom indicating the file is 13L and 315B.

```
[repository]
ROOT = /var/lib/gitea/data/gitea-repositories

[server]
APP_DATA_PATH = /var/lib/gitea/data
DOMAIN = git.saurabhcloud.fun
SSH_DOMAIN = git.saurabhcloud.fun
HTTP_PORT = 3000
ROOT_URL = https://git.saurabhcloud.fun/
LFS_CONTENT_PATH = /var/lib/gitea/data/fs

[log]
ROOT_PATH = /var/lib/gitea/log
```

Fix permissions :

```
sudo chown -R gitea:gitea /etc/gitea
```

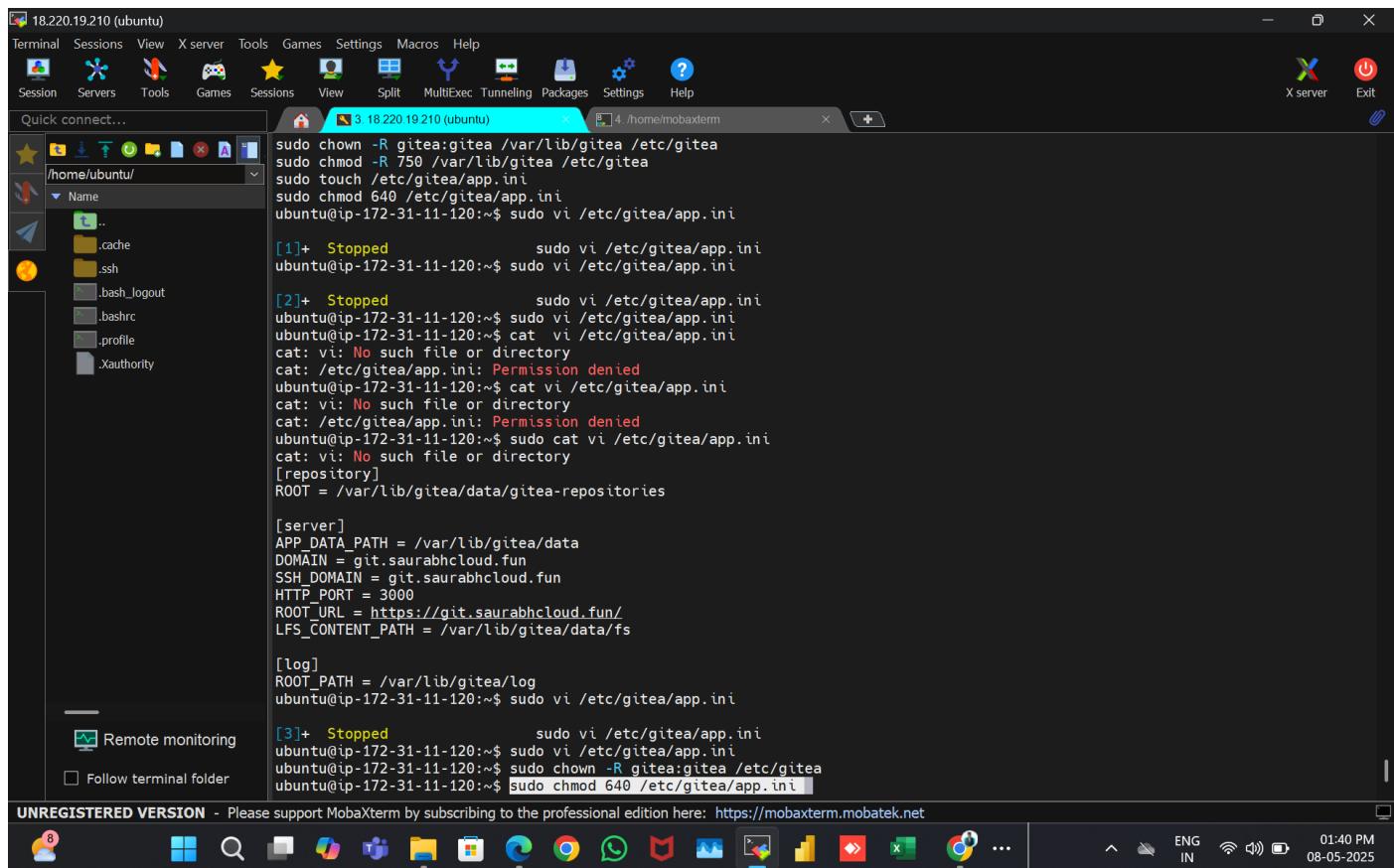
The screenshot shows the MobaXterm interface with a terminal window open to a series of sudo commands being run. The user is attempting to change ownership of the /etc/gitea directory to the gitea user and group, but is denied permission due to insufficient privileges. The terminal window has tabs for session 3 and 4, and a status bar at the bottom indicating the file is 13L and 315B.

```
ubuntu@ip-172-31-11-120:~$ sudo mkdir -p /etc/gitea/{custom,data,log}
sudo chown -R gitea:gitea /var/lib/gitea/{custom,data,log}
sudo chmod -750 /var/lib/gitea /etc/gitea
sudo touch /etc/gitea/app.ini
sudo chmod 640 /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/gitea/app.ini
[1]+  Stopped                  sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/gitea/app.ini
[2]+  Stopped                  sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ cat vi /etc/gitea/app.ini
cat: vi: No such file or directory
cat: /etc/gitea/app.ini: Permission denied
ubuntu@ip-172-31-11-120:~$ cat vi /etc/gitea/app.ini
cat: vi: No such file or directory
cat: /etc/gitea/app.ini: Permission denied
ubuntu@ip-172-31-11-120:~$ sudo cat vi /etc/gitea/app.ini
cat: vi: No such file or directory
[repository]
ROOT = /var/lib/gitea/data/gitea-repositories

[server]
APP_DATA_PATH = /var/lib/gitea/data
DOMAIN = git.saurabhcloud.fun
SSH_DOMAIN = git.saurabhcloud.fun
HTTP_PORT = 3000
ROOT_URL = https://git.saurabhcloud.fun/
LFS_CONTENT_PATH = /var/lib/gitea/data/fs

[log]
ROOT_PATH = /var/lib/gitea/log
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/gitea/app.ini
[3]+  Stopped                  sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ sudo chown -R gitea:gitea /etc/gitea
```

```
sudo chmod 640 /etc/gitea/app.ini
```



```
sudo chown -R gitea:gitea /var/lib/gitea/etc/gitea
sudo chmod -R 750 /var/lib/gitea/etc/gitea
sudo touch /etc/gitea/app.ini
sudo chmod 640 /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/gitea/app.ini
[1]+ Stopped sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/gitea/app.ini
[2]+ Stopped sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ cat vi /etc/gitea/app.ini
cat: vi: No such file or directory
cat: /etc/gitea/app.ini: Permission denied
ubuntu@ip-172-31-11-120:~$ cat vi /etc/gitea/app.ini
cat: vi: No such file or directory
cat: /etc/gitea/app.ini: Permission denied
ubuntu@ip-172-31-11-120:~$ sudo cat vi /etc/gitea/app.ini
cat: vi: No such file or directory
[repository]
ROOT = /var/lib/gitea/data/gitea-repositories

[server]
APP_DATA_PATH = /var/lib/gitea/data
DOMAIN = git.saurabhcloud.fun
SSH_DOMAIN = git.saurabhcloud.fun
HTTP_PORT = 3000
ROOT_URL = https://git.saurabhcloud.fun/
LFS_CONTENT_PATH = /var/lib/gitea/data/fs

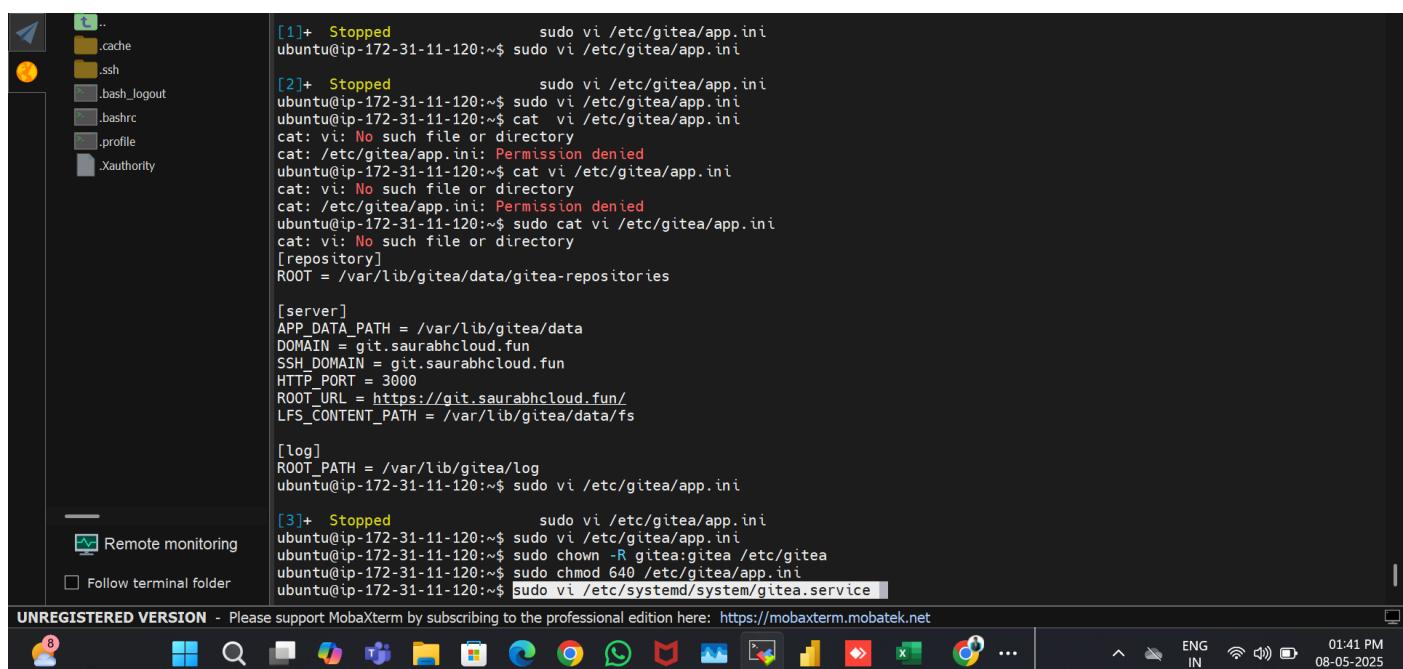
[log]
ROOT_PATH = /var/lib/gitea/log
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/gitea/app.ini
[3]+ Stopped sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ sudo chown -R gitea:gitea /etc/gitea
ubuntu@ip-172-31-11-120:~$ sudo chmod 640 /etc/gitea/app.ini
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

9. Configure Systemd Service for Gitea :

Create a systemd service file to manage Gitea as a background service :

```
sudo vi /etc/systemd/system/gitea.service
```



```
[1]+ Stopped sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/gitea/app.ini
[2]+ Stopped sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ cat vi /etc/gitea/app.ini
cat: vi: No such file or directory
cat: /etc/gitea/app.ini: Permission denied
ubuntu@ip-172-31-11-120:~$ cat vi /etc/gitea/app.ini
cat: vi: No such file or directory
cat: /etc/gitea/app.ini: Permission denied
ubuntu@ip-172-31-11-120:~$ sudo cat vi /etc/gitea/app.ini
cat: vi: No such file or directory
[repository]
ROOT = /var/lib/gitea/data/gitea-repositories

[server]
APP_DATA_PATH = /var/lib/gitea/data
DOMAIN = git.saurabhcloud.fun
SSH_DOMAIN = git.saurabhcloud.fun
HTTP_PORT = 3000
ROOT_URL = https://git.saurabhcloud.fun/
LFS_CONTENT_PATH = /var/lib/gitea/data/fs

[log]
ROOT_PATH = /var/lib/gitea/log
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/gitea/app.ini
[3]+ Stopped sudo vi /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ sudo chown -R gitea:gitea /etc/gitea
ubuntu@ip-172-31-11-120:~$ sudo chmod 640 /etc/gitea/app.ini
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/systemd/system/gitea.service
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Add the following configuration :

[Unit]

Description=Gitea Self-Hosted Git Server

After=network.target mariadb.service

[Service]

User=gitea

Group=gitea

ExecStart=/usr/local/bin/gitea web --config /etc/gitea/app.ini

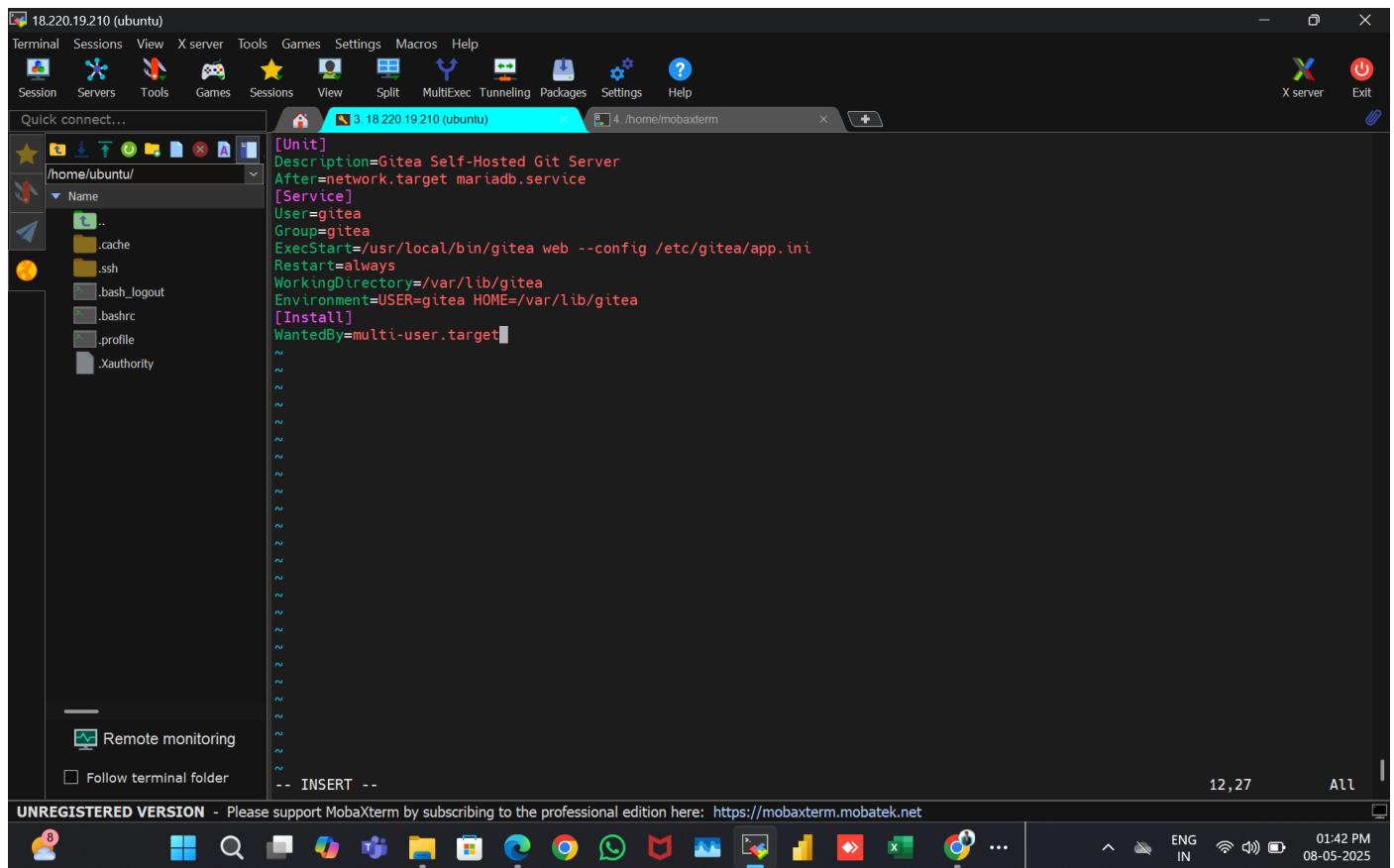
Restart=always

WorkingDirectory=/var/lib/gitea

Environment=USER=gitea HOME=/var/lib/gitea

[Install]

WantedBy=multi-user.target



The screenshot shows a terminal window titled '18.220.19.210 (ubuntu)' in MobaXterm. The window displays the following configuration file:

```
[Unit]
Description=Gitea Self-Hosted Git Server
After=network.target mariadb.service
[Service]
User=gitea
Group=gitea
ExecStart=/usr/local/bin/gitea web --config /etc/gitea/app.ini
Restart=always
WorkingDirectory=/var/lib/gitea
Environment=USER=gitea HOME=/var/lib/gitea
[Install]
WantedBy=multi-user.target
```

The terminal window also shows a sidebar with session management icons and a status bar at the bottom indicating 'UNREGISTERED VERSION' and system information like battery level, network, and date.

Reload Systemd & Start Gitea

Activate the Gitea service so it starts immediately and on boot :

```
sudo systemctl daemon-reload
```

```
sudo systemctl enable --now gitea
```

```
sudo systemctl status gitea
```

```
ubuntu@ip-172-31-11-120:~$ sudo systemctl daemon-reload
sudo systemctl enable --now gitea
sudo systemctl status gitea
Created symlink /etc/systemd/system/multi-user.target.wants/gitea.service → /etc/systemd/system/gitea.service.
● gitea.service - Gitea Self-Hosted Git Server
    Loaded: loaded (/etc/systemd/system/gitea.service; enabled; preset: enabled)
    Active: active (running) since Thu 2025-05-08 08:13:59 UTC; 35ms ago
      Main PID: 20247 (gitea)
        Tasks: 5 (limit: 4674)
       Memory: 2.5M (peak: 2.5M)
         CPU: 22ms
        CGroup: /system.slice/gitea.service
                └─20247 /usr/local/bin/gitea web --config /etc/gitea/app.ini

May 08 08:13:59 ip-172-31-11-120 systemd[1]: Started gitea.service - Gitea Self-Hosted Git Server.
ubuntu@ip-172-31-11-120:~$
```

8 . Set Up Reverse Proxy with Nginx

Create an Nginx config file to route traffic from your custom domain to the Gitea app :

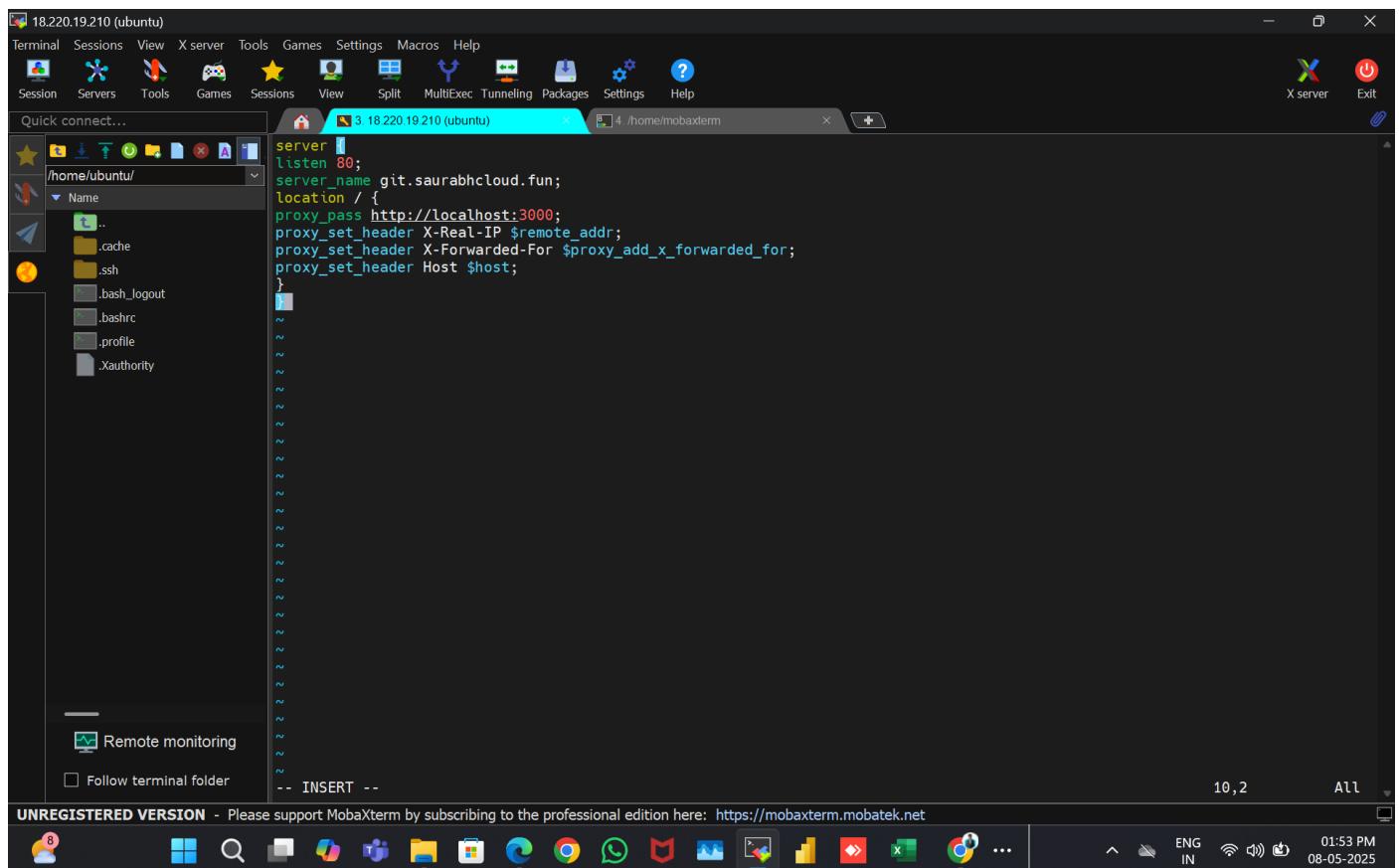
```
sudo vi /etc/nginx/sites-available/gitea
```

```
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/nginx/sites-available/gitea
ubuntu@ip-172-31-11-120:~$ cat /etc/nginx/sites-available/gitea
server {
    listen 80;
    server_name _;
    location / {
        proxy_pass http://127.0.0.1:3000;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    }
}

May 08 08:13:59 ip-172-31-11-120 systemd[1]: Started gitea.service - Gitea Self-Hosted Git Server.
ubuntu@ip-172-31-11-120:~$
```

Add:

```
server {  
    listen 80;  
  
    server_name git.skjptpp.in;  
  
    location / {  
  
        proxy_pass http://localhost:3000;  
  
        proxy_set_header X-Real-IP $remote_addr;  
  
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;  
  
        proxy_set_header Host $host;  
  
    }  
}
```

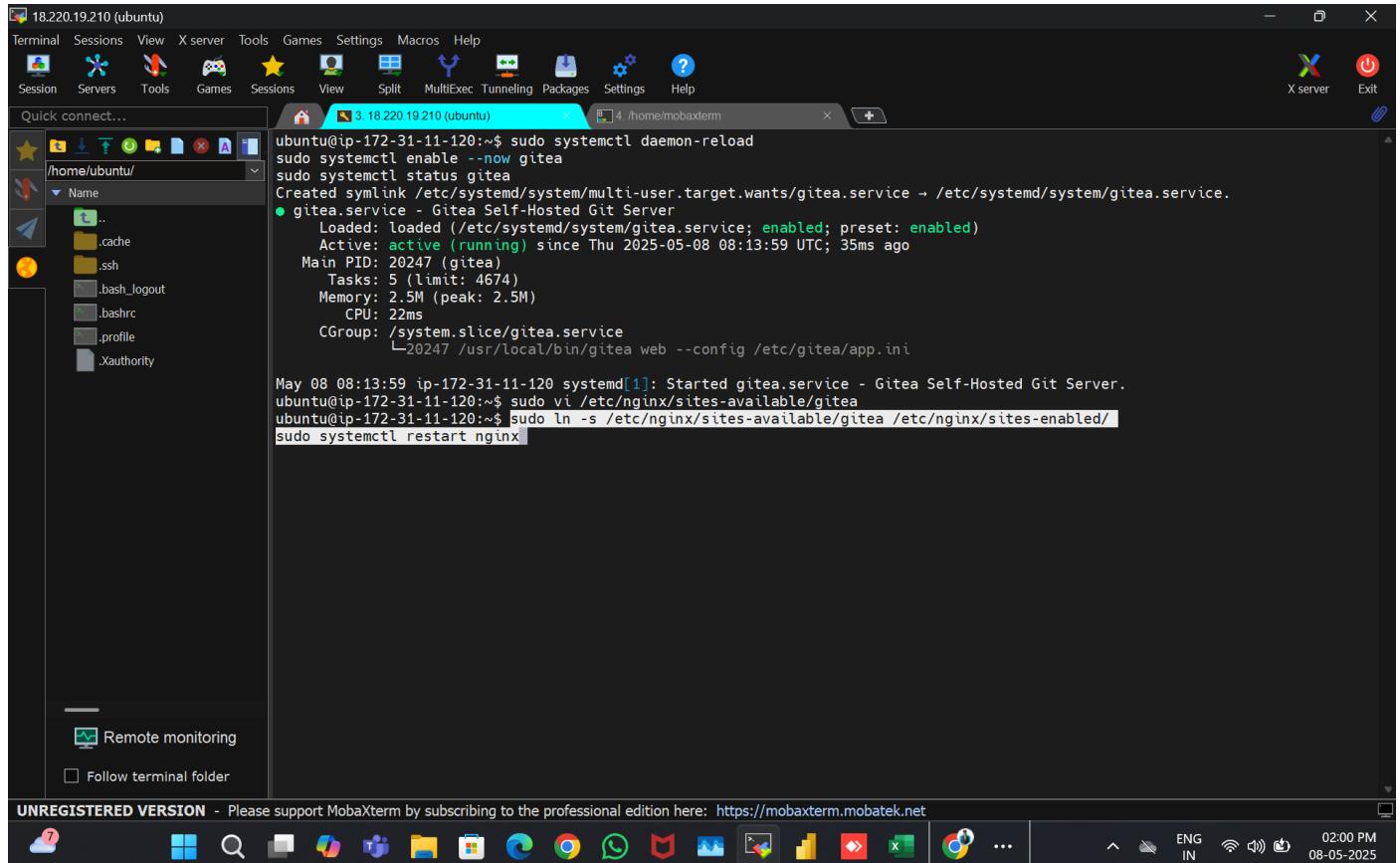


Enable & Restart Nginx :

Activate the Gitea Nginx configuration and restart the Nginx service to apply changes.

```
sudo ln -s /etc/nginx/sites-available/gitea /etc/nginx/sites-enabled/
```

```
sudo systemctl restart nginx
```



```
ubuntu@ip-172-31-11-120:~$ sudo systemctl daemon-reload
sudo systemctl enable --now gitea
sudo systemctl status gitea
Created symlink /etc/systemd/system/multi-user.target.wants/gitea.service → /etc/systemd/system/gitea.service.
● gitea.service - Gitea Self-Hosted Git Server
    Loaded: loaded (/etc/systemd/system/gitea.service; enabled; preset: enabled)
    Active: active (running) since Thu 2025-05-08 08:13:59 UTC; 35ms ago
      Main PID: 20247 (gitea)
        Tasks: 5 (limit: 4674)
       Memory: 2.5M (peak: 2.5M)
         CPU: 22ms
        CGroup: /system.slice/gitea.service
                └─20247 /usr/local/bin/gitea web --config /etc/gitea/app.ini

May 08 08:13:59 ip-172-31-11-120 systemd[1]: Started gitea.service - Gitea Self-Hosted Git Server.
ubuntu@ip-172-31-11-120:~$ sudo vi /etc/nginx/sites-available/gitea
ubuntu@ip-172-31-11-120:~$ sudo ln -s /etc/nginx/sites-available/gitea /etc/nginx/sites-enabled/
sudo systemctl restart nginx
```

Point Your Domain to Your EC2 Instance

To make (git.saurabhcloud.fun) point to your EC2 instance:

Step-by-step:

1. **Log in to your domain registrar's website**
(e.g., GoDaddy, Namecheap, Hostinger, etc.)
2. **Go to DNS Management or DNS Zone Editor**
3. **Create or edit an “A” Record:**
 - **Host/Name:** git (or @ if you're pointing the root domain like saurabhcloud.fun)
 - **Type:** A
 - **Value:** Your EC2 Public IPv4 Address
 - **TTL:** 300 or default
4. **Save the record**

The screenshot shows the Hostinger DNS management interface. On the left, a sidebar lists various services: Overview, Domains (selected), Domain portfolio, Get a New Domain, Transfers, Emails, VPS, Dark web monitoring, Billing, Marketplace, and Account Sharing. The main panel displays a table of DNS records:

Type	Name	Priority	Content	TTL	Delete	Edit
CAA	@	0	O issue "digicert.com"	14400	Delete	Edit
CAA	@	0	O issue "comodoca.com"	14400	Delete	Edit
A	@	0	84.32.84.32	14400	Delete	Edit
A	@	0	18.220.19.210	14400	Delete	Edit

Below the table is a button labeled "Reset DNS records". A tooltip explains: "This feature resets all existing DNS records of saurabhcloud.fun to default." The bottom of the interface includes a navigation bar with icons for file operations, a search bar, and system status indicators.

Type Name Value (EC2 IP) TTL
A git 13.234.XX.XX 300

Now, when someone opens <https://git.saurabhcloud.fun>, it will route to your EC2 instance running Gitea.

The screenshot shows the Hostinger DNS management interface after changes have been made. The sidebar remains the same. The main panel now displays a table of DNS records with the following entries:

Type	Name	Priority	Content	TTL	Delete	Edit
CNAME	www	0	saurabhcloud.fun	300	Delete	Edit
A	git	0	18.220.19.210	14400	Delete	Edit

9. Secure Gitea with SSL (Let's Encrypt)

Use Certbot to automatically fetch and configure a free SSL certificate for your domain:

```
sudo certbot --nginx -d git.saurabhcloud.fun
```

```
ubuntu@ip-172-31-11-120:~$ sudo certbot --nginx -d git.saurabhcloud.fun
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Enter email address (used for urgent renewal and security notices)
(Enter 'c' to cancel): ssjadhav292001@gmail.com

-----
Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.5-February-24-2025.pdf. You must
agree in order to register with the ACME server. Do you agree?
-----
```

(Y)es/(N)o: y

```
-----
```

Would you be willing, once your first certificate is **successfully** issued, to share your email address with the Electronic Frontier Foundation, a founding partner of the Let's Encrypt project and the non-profit organization that develops Certbot? We'd like to send you email about our work encrypting the web, EFF news, campaigns, and ways to support digital freedom.

```
-----
```

(Y)es/(N)o: y

```
-----
```

Account registered.

Requesting a certificate for git.saurabhcloud.fun

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

02:03 PM 08-05-2025

Certificate Successfully Enabled .

```
18.220.19.210 (ubuntu)
Terminal Sessions View Xserver Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Xserver Exit
Quick connect... 3 18.220.19.210 (ubuntu) 4 ./home/mobaxterm
Enter email address (used for urgent renewal and security notices)
(Enter 'c' to cancel): ssjadhav292001@gmail.com

-----
Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.5-February-24-2025.pdf. You must
agree in order to register with the ACME server. Do you agree?
-----
```

(Y)es/(N)o: y

```
-----
```

Would you be willing, once your first certificate is **successfully** issued, to share your email address with the Electronic Frontier Foundation, a founding partner of the Let's Encrypt project and the non-profit organization that develops Certbot? We'd like to send you email about our work encrypting the web, EFF news, campaigns, and ways to support digital freedom.

```
-----
```

(Y)es/(N)o: y

```
-----
```

Account registered.

Requesting a certificate for git.saurabhcloud.fun

```
-----
```

Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/git.saurabhcloud.fun/fullchain.pem
Key is saved at: /etc/letsencrypt/live/git.saurabhcloud.fun/privkey.pem
This certificate expires on 2025-08-06.
These files will be updated when the certificate renews.
Certbot has set up a scheduled task to automatically renew this certificate in the background.

```
-----
```

Deploying certificate
Successfully deployed certificate for git.saurabhcloud.fun to /etc/nginx/sites-enabled/gitea
Congratulations! You have **successfully** enabled HTTPS on https://git.saurabhcloud.fun

```
-----
```

If you like Certbot, please consider supporting our work by:
* Donating to ISRG / Let's Encrypt: <https://letsencrypt.org/donate>
* Donating to EFF: <https://eff.org/donate-le>

```
-----
```

ubuntu@ip-172-31-11-120:~\$

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

02:04 PM 08-05-2025

Set up auto-renewal :

Schedule automatic SSL certificate renewal using a cron job:

```
sudo crontab -e
```

And Enter 2nd for Vim Editor

```
ubuntu@ip-172-31-11-120:~$ sudo crontab -e
no crontab for root - using an empty one

Select an editor. To change later, run 'select-editor'.
 1. /bin/nano   <---- easiest
 2. /usr/bin/vim.basic
 3. /usr/bin/vim.tiny
 4. /bin/ed

Choose 1-4 [1]: 2
crontab: installing new crontab
ubuntu@ip-172-31-11-120:~$
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

08-05-2025 02:06 PM ENG IN

Add :

Add the following line to run renewal daily at 3 AM:

0 3 * * * certbot renew --quiet

```
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcvf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
0 3 * * * certbot renew --quiet
```

-- INSERT --

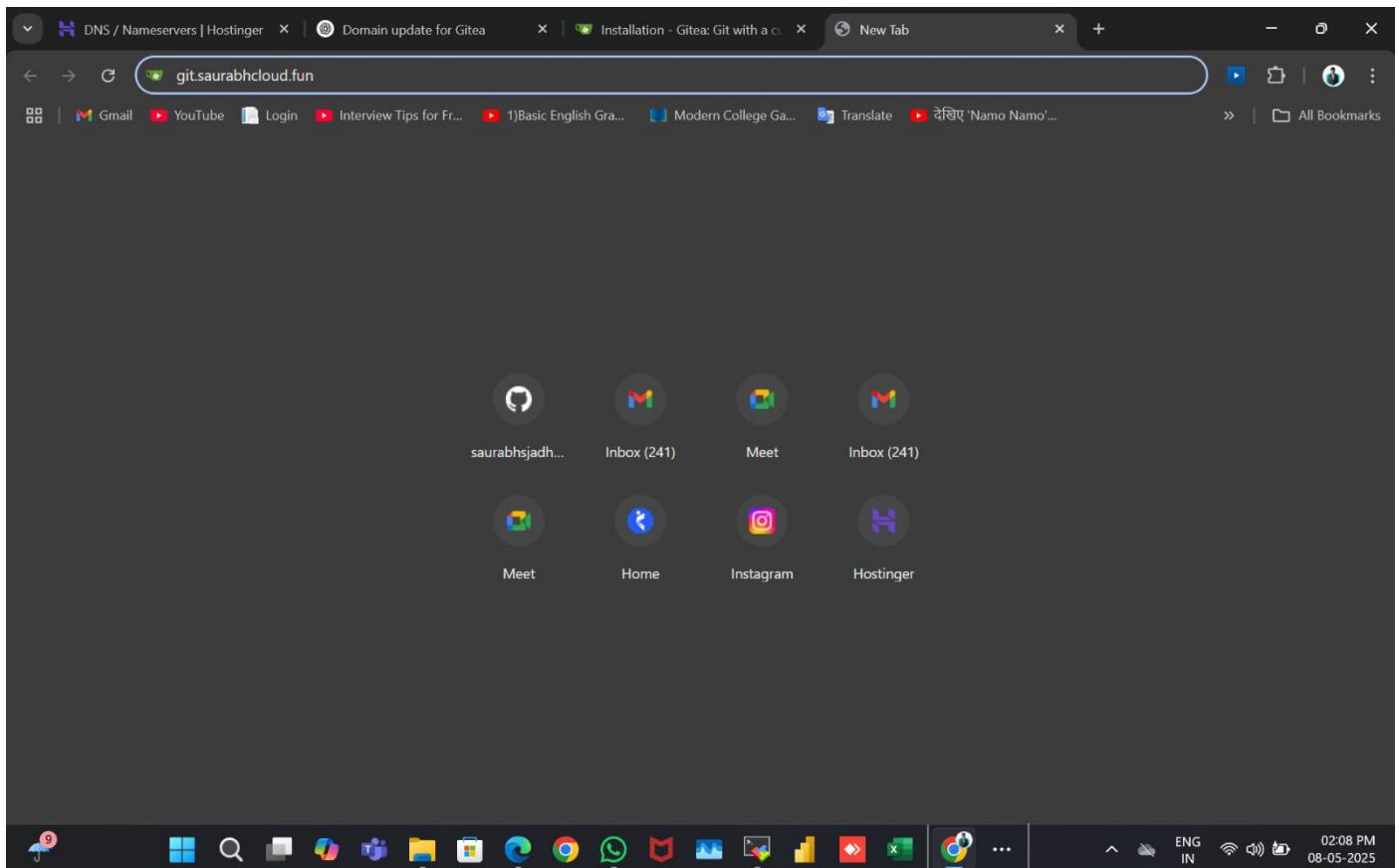
26,1 All

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

08-05-2025 02:06 PM ENG IN

Open Any Browser and hit your domain name.

Open <https://git.saurabhcloud.fun>



Access Gitea Web Interface

- Complete initial setup
- Use Gitea database credentials
- Create an admin user

The screenshot shows the 'Initial Configuration' page of the Gitea web interface. The page has two main sections: 'Database Settings' and 'General Settings'.

Database Settings:

- Database Type: MySQL
- Host: 127.0.0.1:3306
- Username: gitea
- Password: (redacted)
- Database Name: gitea

General Settings:

- Site Title: Gitea: Git with a cup of tea
- Repository Root Path: /var/lib/gitea/data/gitea-repositories
- Git LFS Root Path: /var/lib/gitea/data/f...

Install Gitea

Screenshot of the Gitea installation configuration page.

Gitea Base URL *: `https://git.saurabhcloud.fun/`
Base address for HTTP(S) clone URLs and email notifications.

Log Path *: `/var/lib/gitea/log`
Log files will be written to this directory.

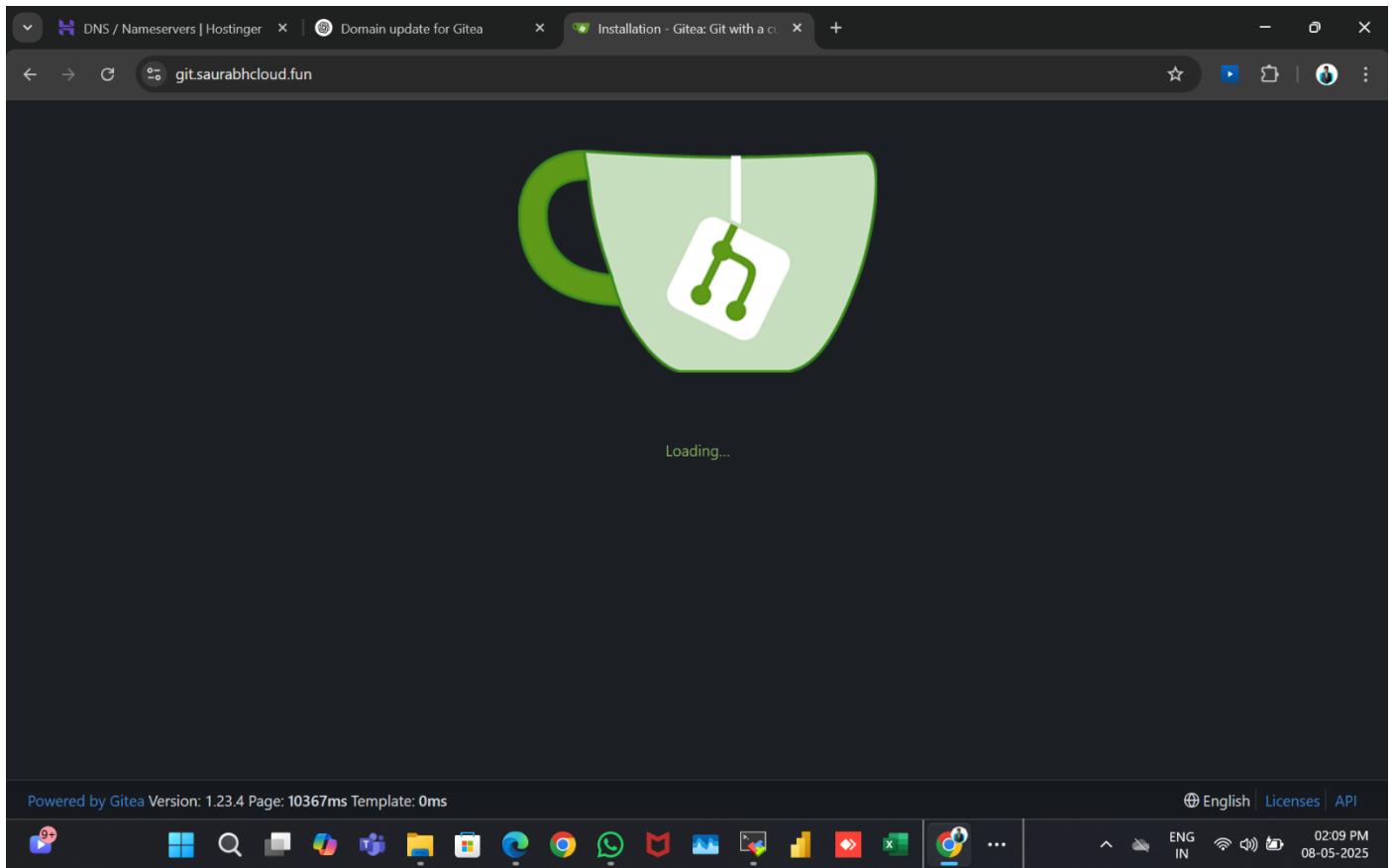
Enable Update Checker
Checks for new version releases periodically by connecting to gitea.io.

Optional Settings

- ▶ Email Settings
- ▶ Server and Third-Party Service Settings
- ▶ Administrator Account Settings

These configuration options will be written into: `/etc/gitea/app.ini` [Edit](#)

Install Gitea



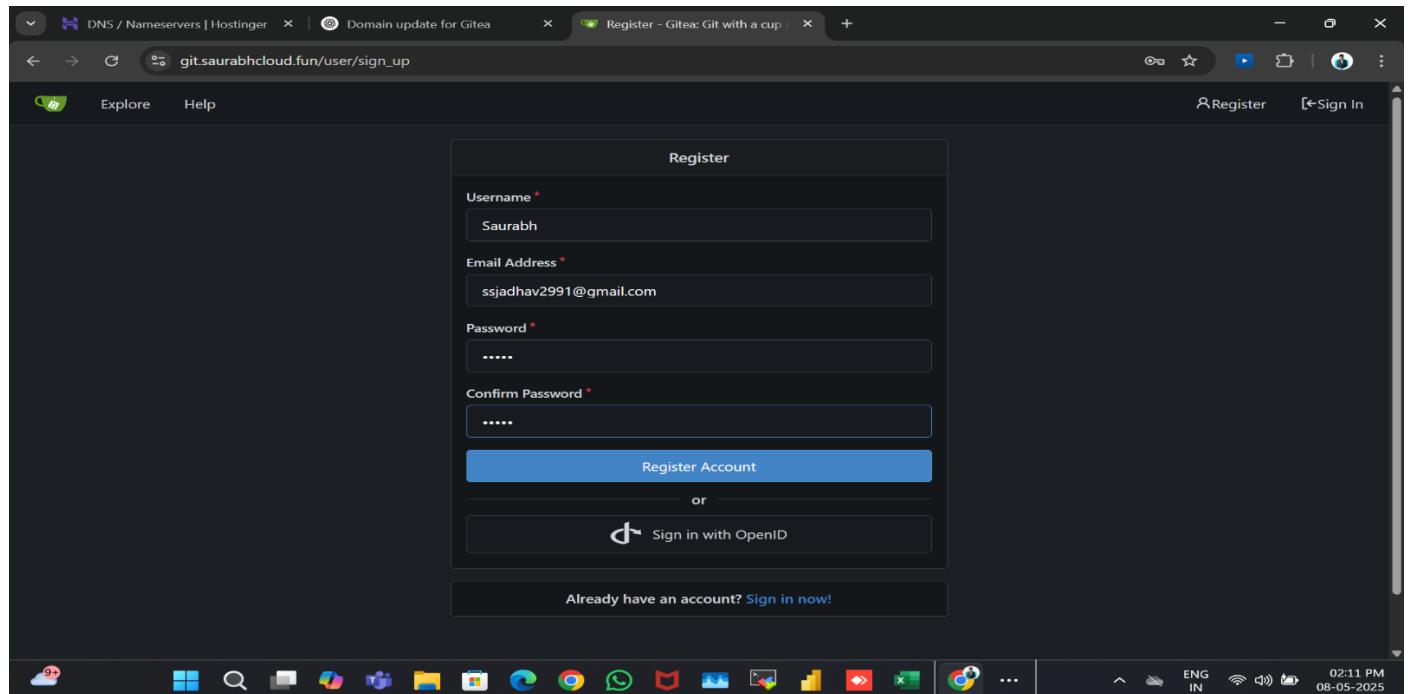
Enable SSH Access for Git Repositories.

Add your current user to the Gitea group to allow SSH-based Git operations, then restart Gitea:

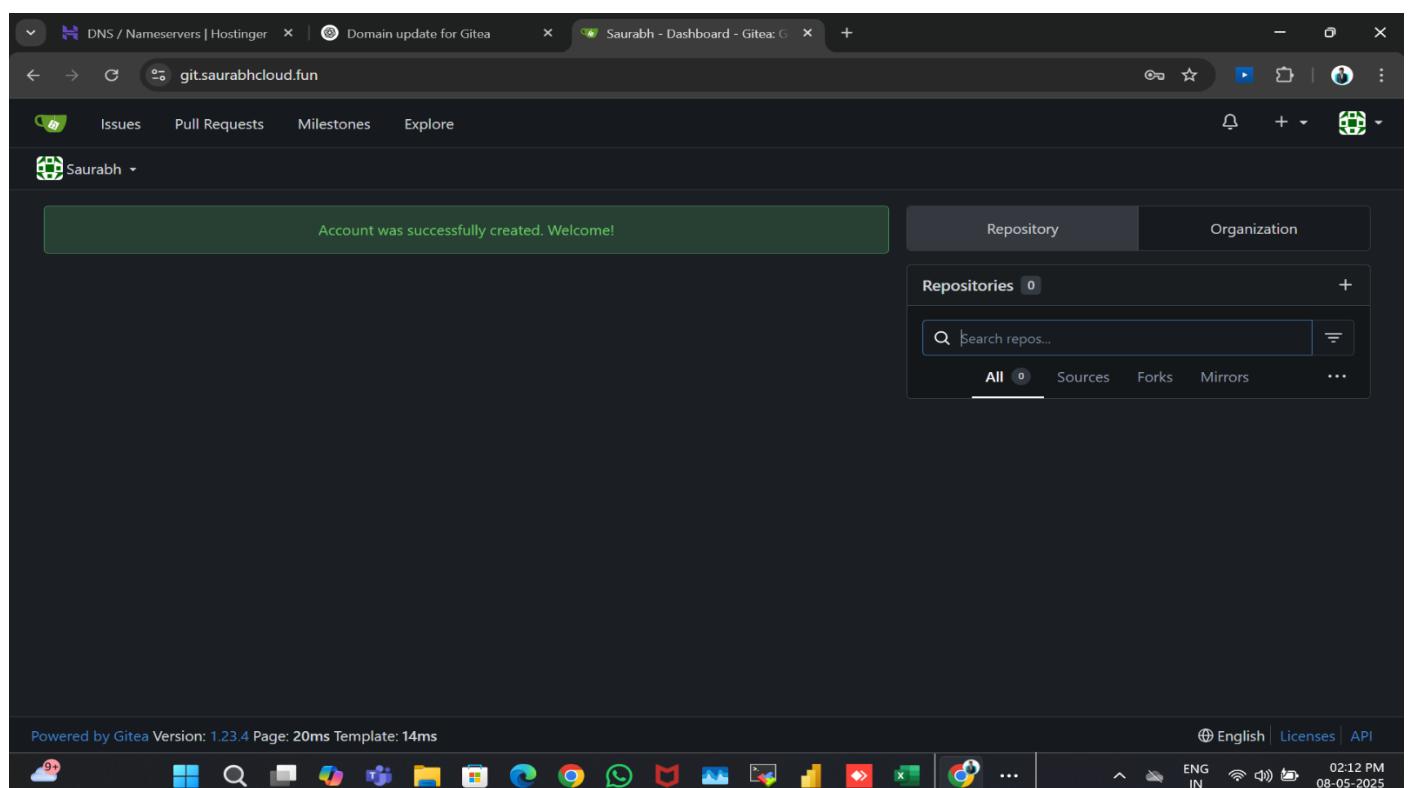
```
sudo usermod -aG gitea $(whoami)
```

```
sudo systemctl restart gitea
```

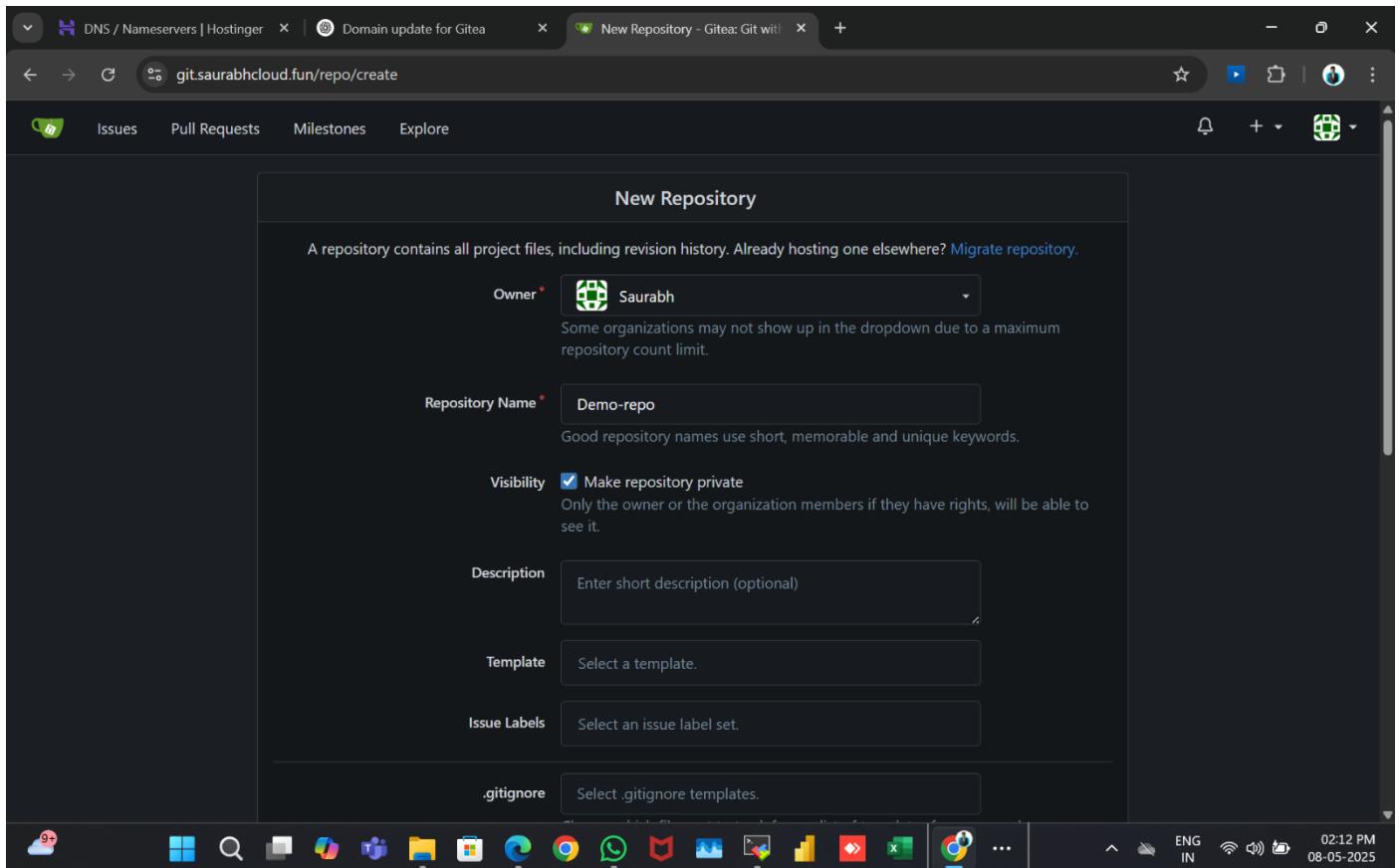
```
git clone git@git.saurabhcloud.fun:your-saurabhsjadhav/your-repo.git
```



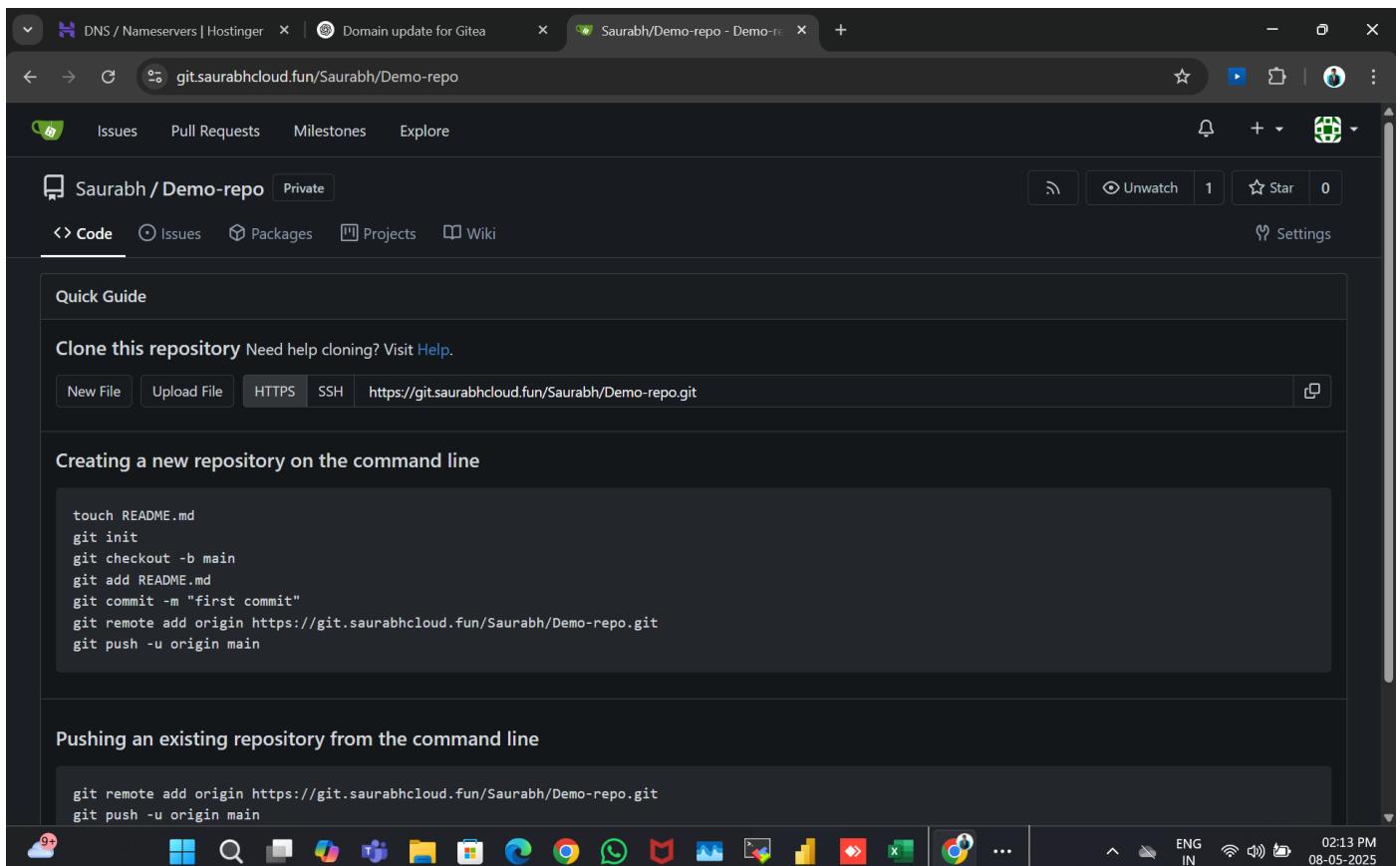
Account Successfully Created.



Create a repo.



Your Repo is Successfully Created.



Maintain Your Profile .

The screenshot shows a Gitea user profile for 'Saurabh'. The profile includes a profile picture of a man standing outdoors, basic statistics (0 Followers, 0 Following), and contact information (email: ssjadav2991@gmail.com). The user joined on May 8, 2025. The main interface shows a repository list with one item: 'Demo-repo' (Private, updated 2 minutes ago). The interface has a dark theme with light-colored text and icons.

Successfully set up a secure, private Git server with Gitea on AWS using your custom domain — ready for powerful team collaboration and code management.

Thank You.

