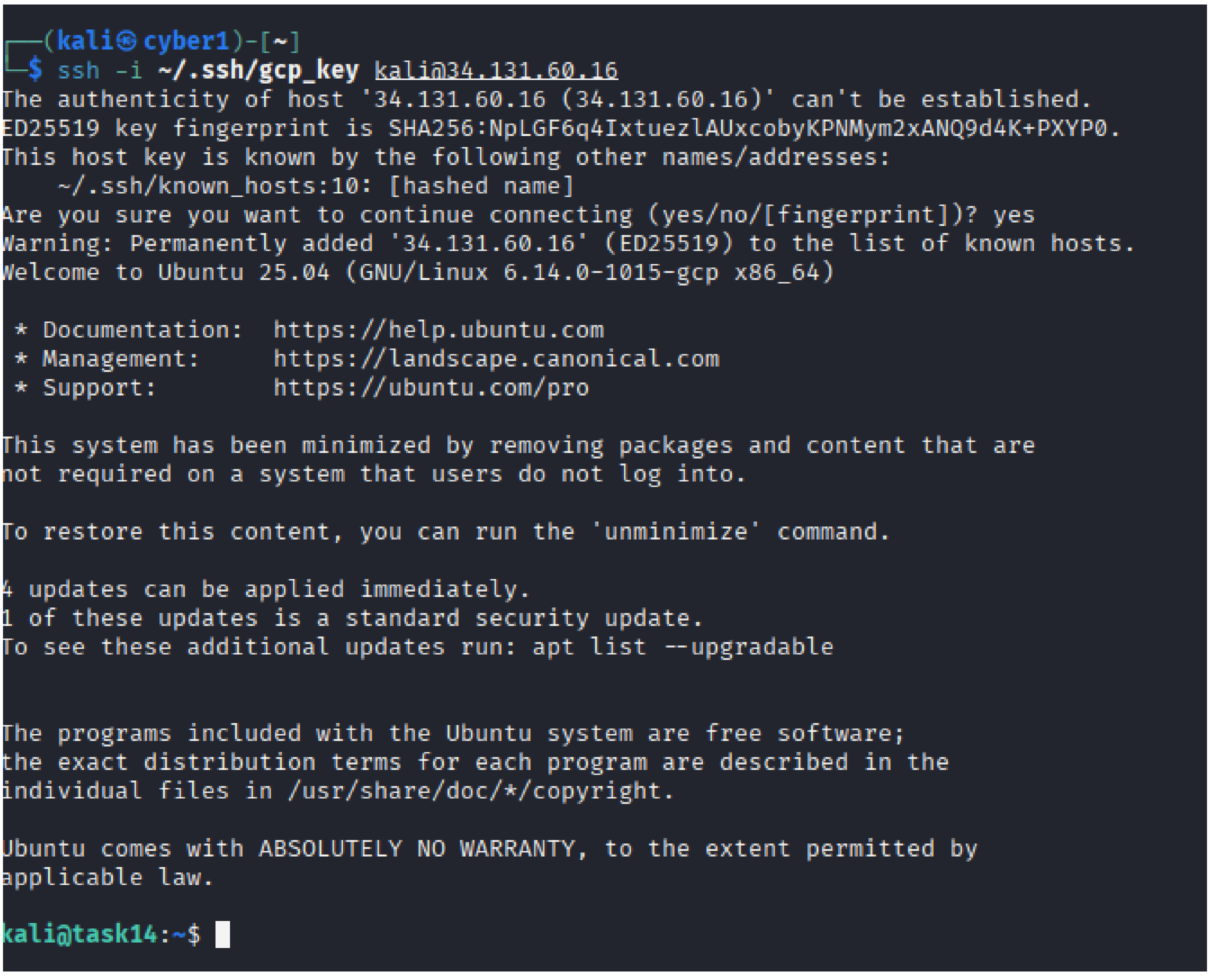
Cloud VM Deployment

Step 1: Get the VM Ready

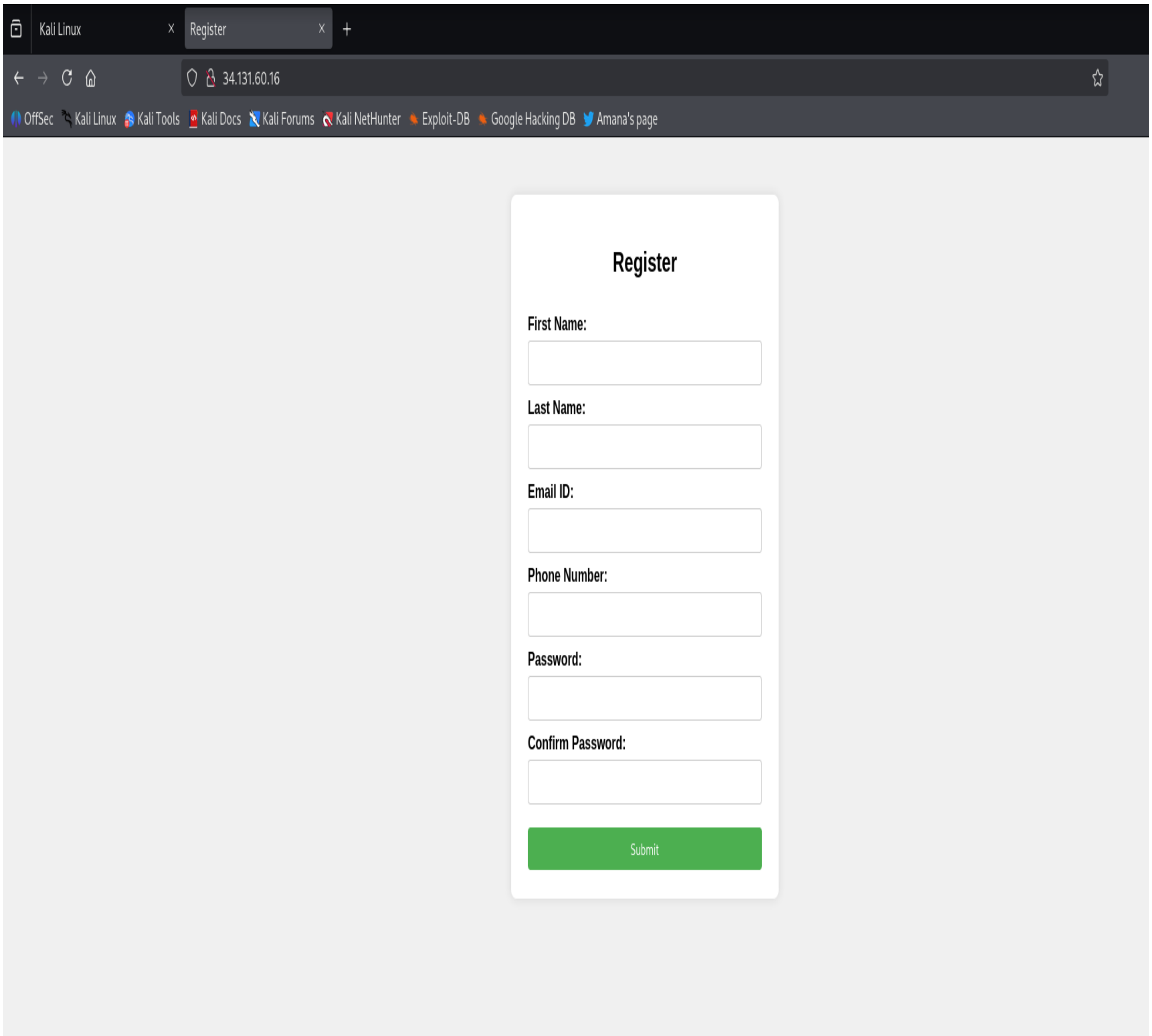
To start, you need to create a free virtual machine (VM) in a cloud service like AWS, GCP, or Azure. Once the VM is running, you'll establish a secure connection to it from your Kali Linux terminal using SSH.



```
┌──(kali㉿cyber1)-[~]
└─$ ssh -i ~/.ssh/gcp_key kali@34.131.60.16
The authenticity of host '34.131.60.16 (34.131.60.16)' can't be established.
ED25519 key fingerprint is SHA256:NpLGF6q4IxtuezlAUxcobyKPNMym2xANQ9d4K+PXYP0.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:10: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '34.131.60.16' (ED25519) to the list of known hosts.
Welcome to Ubuntu 25.04 (GNU/Linux 6.14.0-1015-gcp x86_64)

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

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

4 updates can be applied immediately.
1 of these updates is a standard security update.
To see these additional updates run: apt list --upgradable


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.

kali@task14:~$ 
```

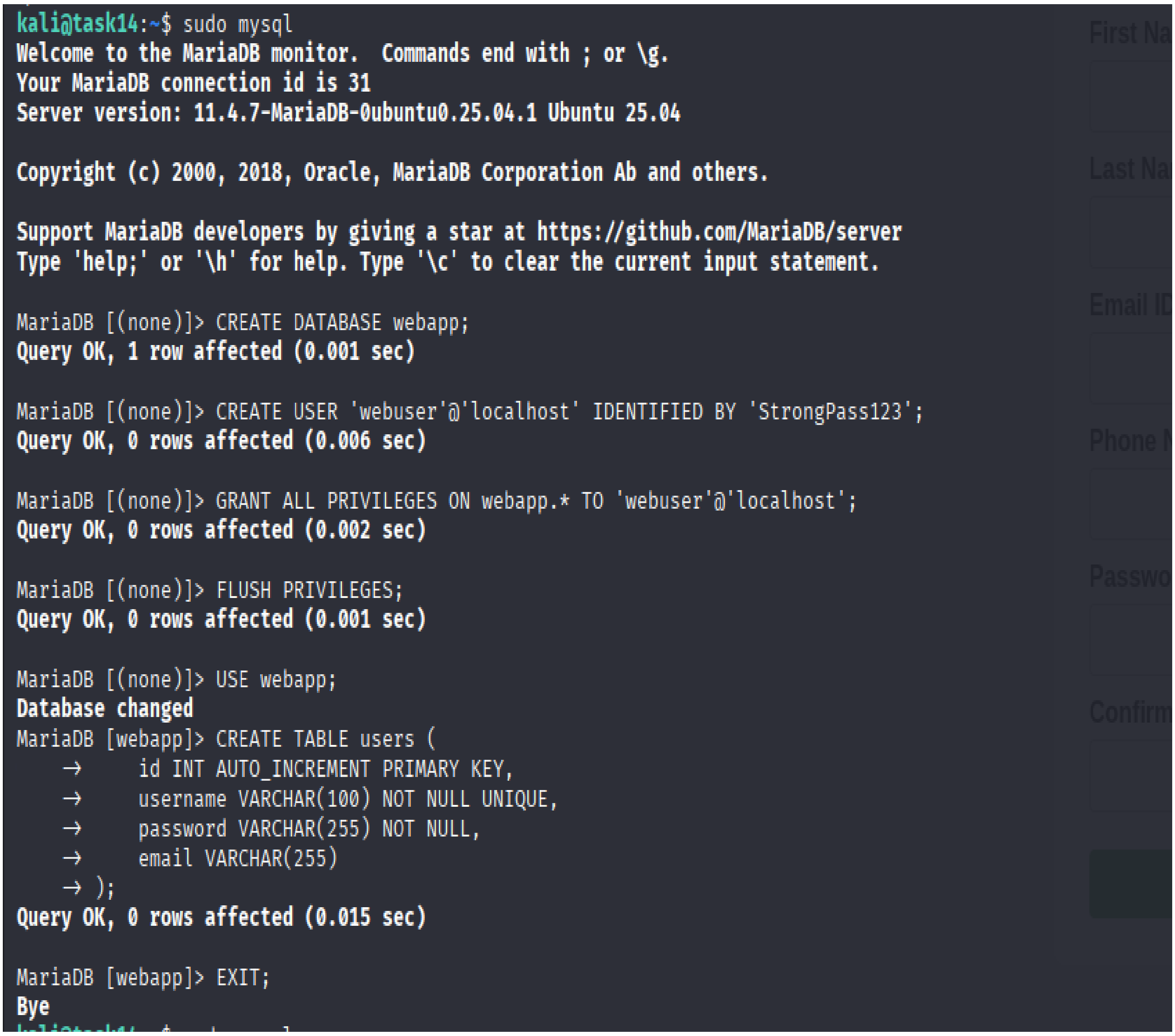
Step 2: Put Your Web Page Online

After you're connected, you'll get your web page files onto the VM. First, **clone the GitHub repo**. Then, you'll **install the Apache web server and copy your files** into its root directory. This makes your webpage accessible to the public internet. Your second screenshot will show your webpage successfully loaded in a web browser using the VM's public IP address.



Step 3: Set Up the Database

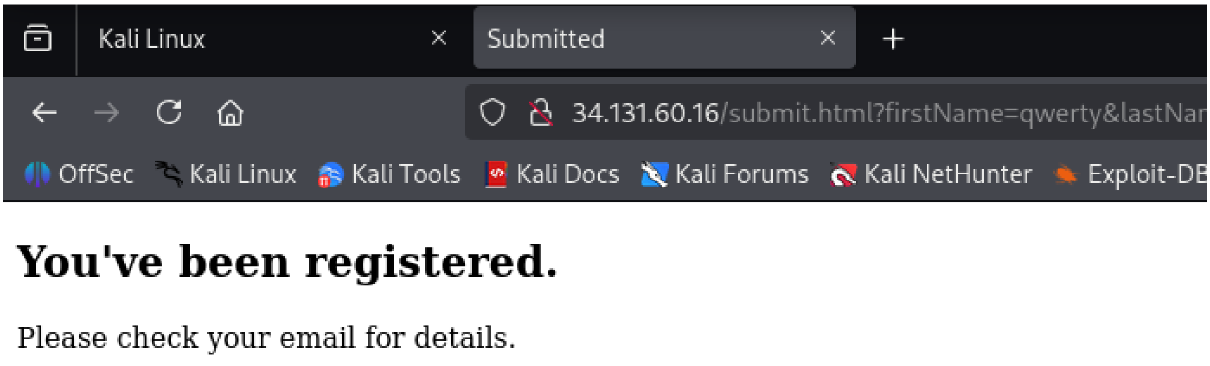
To enable user registration,Need a place to store the data. You will **install PHP and MySQL** on the VM. After the installation, **log into MySQL** and **create a new database, a table, and a user**. This provides the necessary structure to store of user information. the third screenshot will capture the terminal commands you used to set up the database components.



```
kali@task14:~$ sudo mysql
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 31
Server version: 11.4.7-MariaDB-0ubuntu0.25.04.1 Ubuntu 25.04

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

Support MariaDB developers by giving a star at https://github.com/MariaDB/server
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE webapp;
Query OK, 1 row affected (0.001 sec)

MariaDB [(none)]> CREATE USER 'webuser'@'localhost' IDENTIFIED BY 'StrongPass123';
Query OK, 0 rows affected (0.006 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON webapp.* TO 'webuser'@'localhost';
Query OK, 0 rows affected (0.002 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> USE webapp;
Database changed
MariaDB [webapp]> CREATE TABLE users (
    →     id INT AUTO_INCREMENT PRIMARY KEY,
    →     username VARCHAR(100) NOT NULL UNIQUE,
    →     password VARCHAR(255) NOT NULL,
    →     email VARCHAR(255)
    → );
Query OK, 0 rows affected (0.015 sec)

MariaDB [webapp]> EXIT;
Bye
```

Step 4: Create a User Account

In this final step, The web application fully functional. we need to **create the necessary PHP files** that will handle user registration and login. Once the files are in place, **register a new user and then log back in** to test the system. Your fourth and final screenshot will show that you have successfully logged in, confirming everything is working as it should.



34.131.60.16/submit.html?firstName=qwerty&lastNar

You've been registered.

Please check your email for details.

Final Checklist

- **Note the VM's public IP:** This is the address people will use to access the webpage.
- **Upload the files:** Create a text file with these steps, the four screenshots, and the IP, and upload it all to your GitHub repo named `task14`.
- **Terminate the VM:** Be sure to terminate the VM after a week to avoid any unexpected charges.