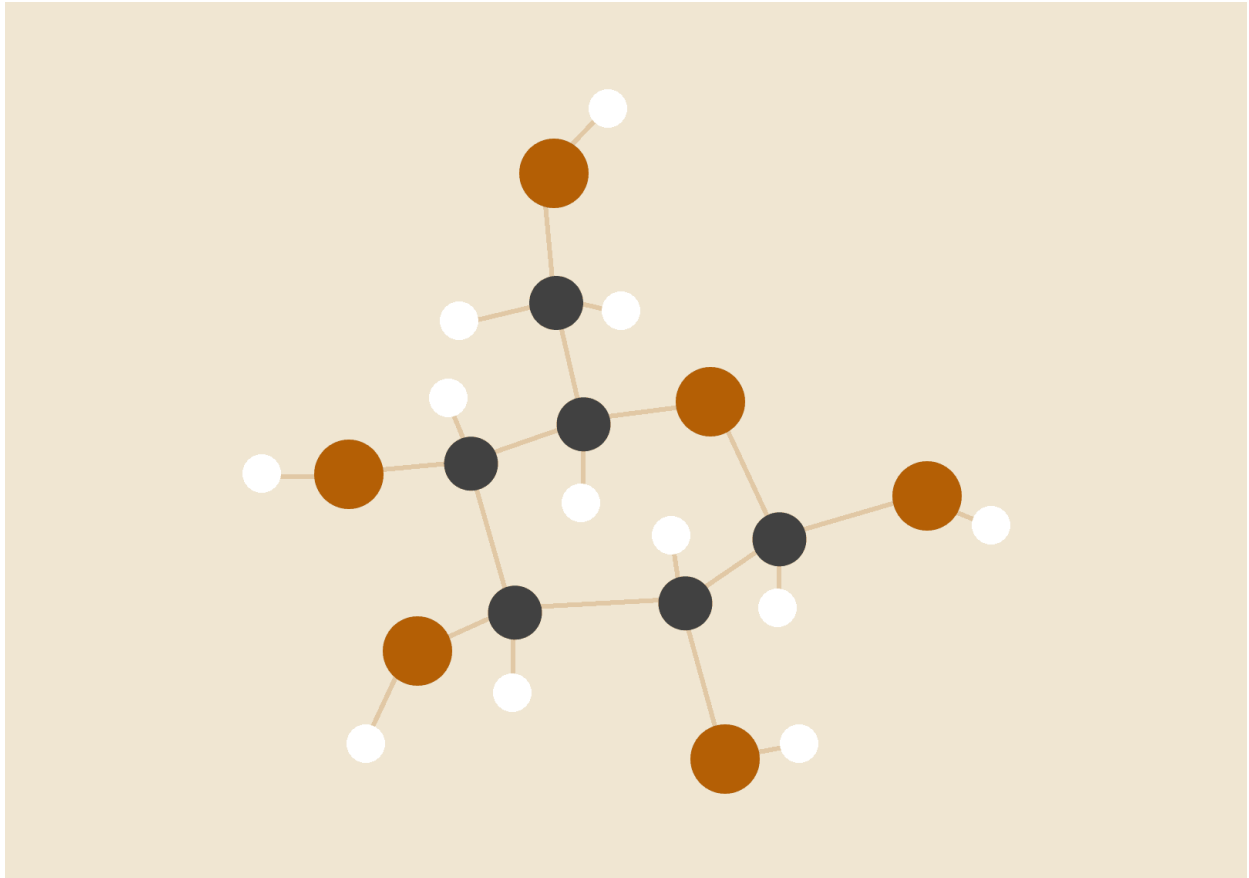


TASK 14 REPORT



Aswin Suresh

14.09.2025

ICT Kerala

1.0 Introduction

This report provides a detailed overview of the implementation steps for Task 14, a project focused on deploying a full-stack web application on the Google Cloud Platform (GCP). The project involved configuring a robust infrastructure, including an Apache web server and a MySQL database, to support a PHP-based user management application with core functionalities such as user registration and authentication.

2.0 Implementation Steps

2.1 GCP VM Creation

- An `e2-micro` instance running Ubuntu 20.04 was created on Google Cloud Platform.
- Firewall rules were configured to allow HTTP and HTTPS traffic from external sources.
- The external IP address assigned to the VM is 34.45.46.33.

2.2 SSH Configuration

- An SSH key pair was generated on a Kali Linux machine to secure the connection.
- The public key was added to the GCP VM's authorized keys.
- A successful and secure connection was established via SSH.

2.3 Apache Installation

- The package list was updated using `sudo apt update`.
- The Apache web server was installed using `sudo apt install apache2 -y`.
- The Apache service was started to ensure the web server was running correctly.

2.4 Repository Deployment

- The Task 12 repository, containing the application source code, was cloned using `git clone https://github.com/aswinsuresh487/task12`.
- The application files were copied to the Apache web root directory, `/var/www/html/`.
- File permissions were adjusted to ensure the web server could access and serve the content.

2.5 PHP and MySQL Setup

- PHP and MySQL were installed with the command `sudo apt install php php-mysql mysql-server -y`.
- The MySQL installation was secured with `mysql_secure_installation`.
- A new database, `task14_db`, and a table, `users`, were created.
- A dedicated user, `webuser`, was created with the necessary permissions for the application to interact with the database.

2.6 Application Development

- `config.php` was created to handle the database connection.
- `register.php` was developed to allow new user registration.
- `login.php` was developed to handle user authentication.
- `dashboard.php` was created as the protected page for logged-in users.
- A logout functionality was implemented to allow users to securely end their sessions.

3.0 Testing and Verification

Upon completion of the setup, the following tests were conducted to verify the application's functionality:

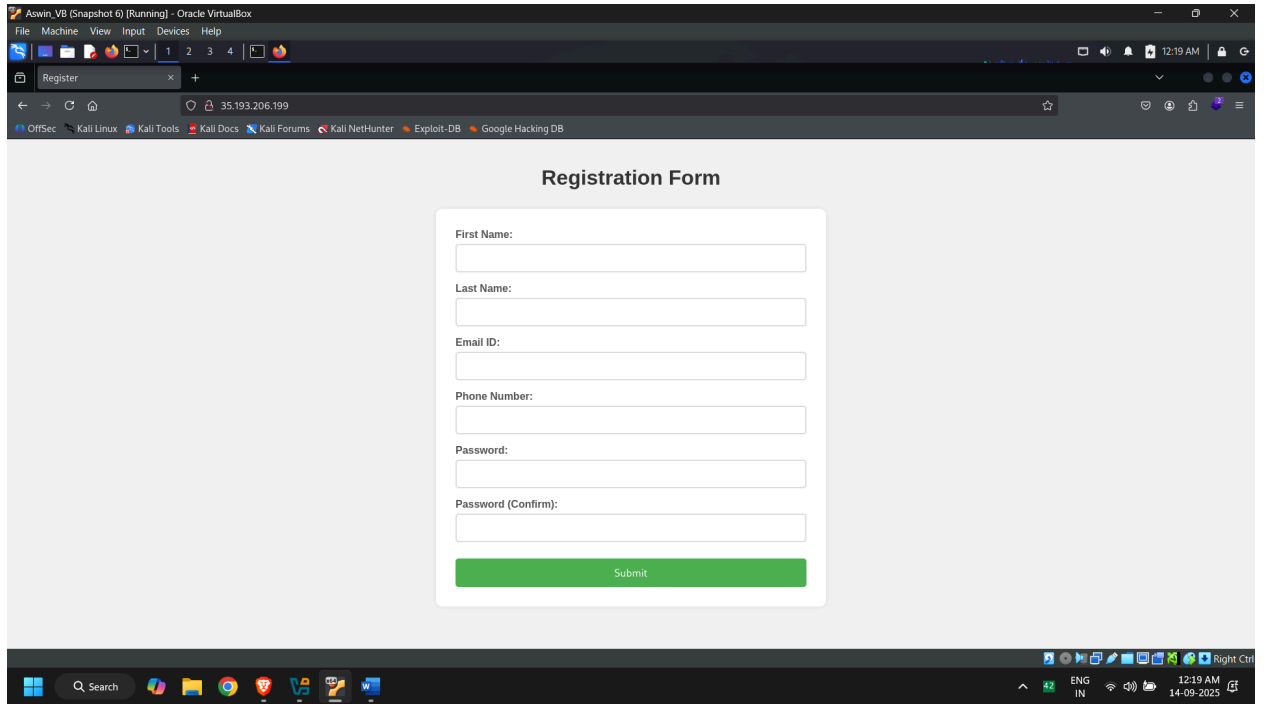
- **Registration:** A new user was successfully registered, and their data was stored in the database.
- **Login:** The newly registered user was able to log in successfully and access the dashboard.
- **Database Storage:** User data was confirmed to be correctly stored in the `task14_db` database.
- **Public Access:** The web page was accessible from a public network, confirming that firewall and network settings were correct.

Screenshots were taken to document the process, including:

1. The SSH connection from Kali Linux.

```
aswin487@instance-20250913-161612:~$ uname -a
Linux instance-20250913-161612 6.1.0-39-cloud-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.148-1 (2025-08-26) x86_64 GNU/Linux
aswin487@instance-20250913-161612:~$ id
uid=1000(aswin487) gid=1001(aswin487) groups=1001(aswin487),4(adm),30(dip),44(video),46(plugdev),1000(google-sudoers)
aswin487@instance-20250913-161612:~$
```

2. The web page is accessible from a local browser.



3. The creation of the MySQL database and table.

```
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 task14_db;
Query OK, 1 row affected (0.004 sec)

MariaDB [(none)]> USE task14_db;
Database changed
MariaDB [task14_db]> CREATE TABLE users (
  →      id INT AUTO INCREMENT PRIMARY KEY,
  →      username VARCHAR(50) NOT NULL,
  →      password VARCHAR(255) NOT NULL
  → );
Query OK, 0 rows affected (0.097 sec)

MariaDB [task14_db]> show tables
  → ;
+-----+
| Tables_in_task14_db |
+-----+
| users                |
+-----+
1 row in set (0.001 sec)

MariaDB [task14_db]> CREATE USER 'webuser'@'localhost' IDENTIFIED BY 'password';
Query OK, 0 rows affected (0.004 sec)

MariaDB [task14_db]> GRANT ALL PRIVILEGES ON task14_db.* TO 'webuser'@'localhost';
Query OK, 0 rows affected (0.002 sec)

MariaDB [task14_db]> SHOW TABLES;
+-----+
| Tables_in_task14_db |
+-----+
| users                |
+-----+
1 row in set (0.000 sec)

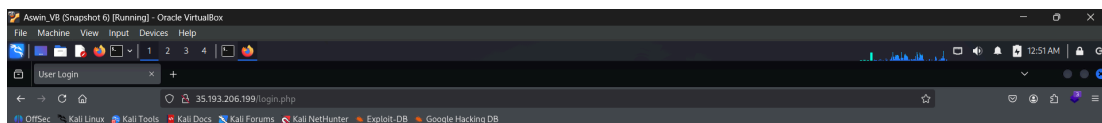
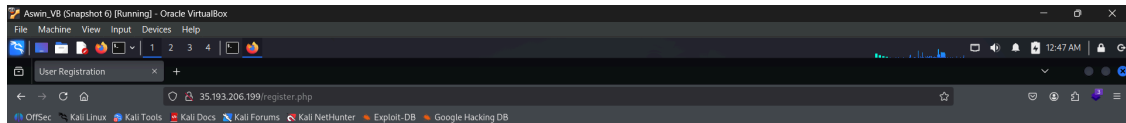
MariaDB [task14_db]> █
```

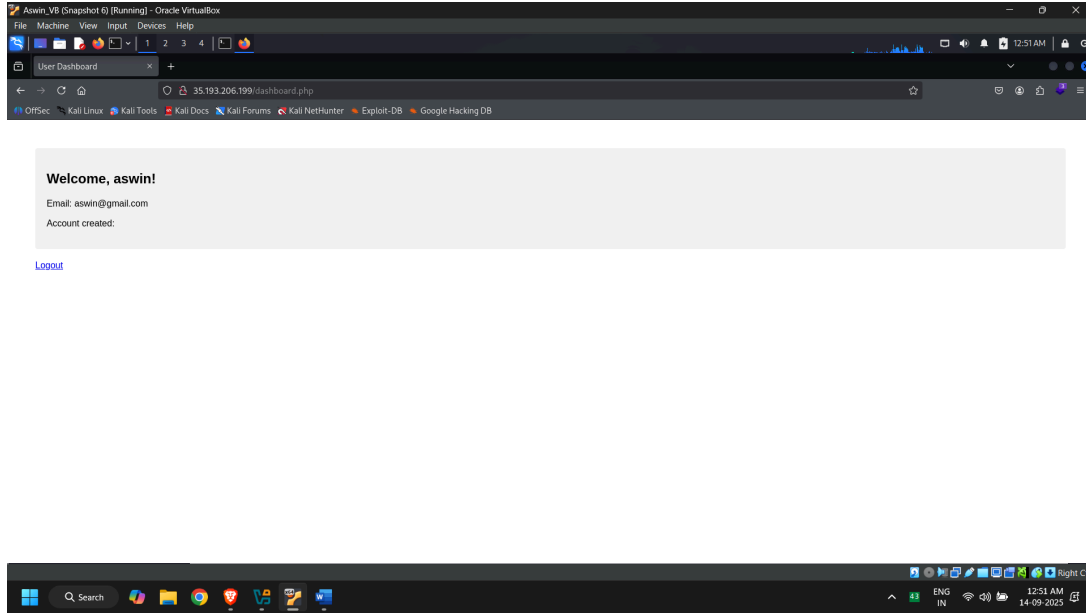
```
MariaDB [task14_db]> select * from users;
```

id	username	email	password
1	aswin	aswin@gmail.com	\$2y\$10\$eerUmh8K3dpmwAQGmMYJae0CjjJs3TI201SH//WeSTnoMAnGhPQRS
2	arjun	arjun@gmail.com	\$2y\$10\$vdrBCBVSQLf bu20a.fNhfuRwvjnNlwJH6aEsLW7I0J0DQmwfbtMKS
3	vaishnav	vaishnav@gmail.com	\$2y\$10\$1sEuBSlFc0ni0SmMeLUXs03MpjLxQWPqbywMieRNkNrd9NN4u6p4G
4	rohit	rohit@gmail.com	\$2y\$10\$Gv9M3.E.uCBHPguu9VzmCe5pG1NMmYWIRFefIQS2W.ep/DReCcXhm

4 rows in set (0.000 sec)

4. The user dashboard after a successful login.





The public IP address for the application is 34.45.46.33, and the key URLs are:

- Main page: <http://34.45.46.33/>
- Registration: <http://34.45.46.33/register.php>
- Login: <http://34.45.46.33/login.php>

4.0 Conclusion

In conclusion, the successful deployment of this web application on GCP demonstrated not only the technical skills required for cloud infrastructure and server configuration but also the importance of systematic testing and verification. This project was a valuable exercise in deploying a full-stack application within a professional cloud environment.