KL UNIVERSITY

Department of Computer Science & Engineering

Open Source Report

Submitted by: Raghuram Pasupuleti
Roll No.: 2400031015
Semester: 3rd Semester
Course: Open Source Engineering(24CS02EF)
Academic Year: 2024–2025

Submitted to: **DR. Sripath Roy**

Contents

List of Figures

0) Front Page & Student Details

*

- **Project / Report Title:** My Open Sourse Final Report
- **Course:** Open Source / Open Source Engineering
- **Student:** Raghuram Pasupuleti, Roll: 2400031015
- **Department:** CSE-1(HTE)
- **College:** KL University
- **Faculty:** Dr. Sripath Roy
- **Academic Year:** 2025–2026

1 About the Linux Distro You Used

Distribution Details

- **Distribution:** Ubuntu (LTS)
- **Version:** Ubuntu 24.04.3
- **Why chosen:** Personal preference and widespread community support.
- **Install method:** Dual Boot (or VM as per environment)

Installation Steps (brief)

1. Downloaded Ubuntu ISO (version: 24.04.3).
2. Created bootable USB and installed alongside existing OS (dual-boot).
3. Post-install: update packages using `sudo apt update && sudo apt upgrade -y`.

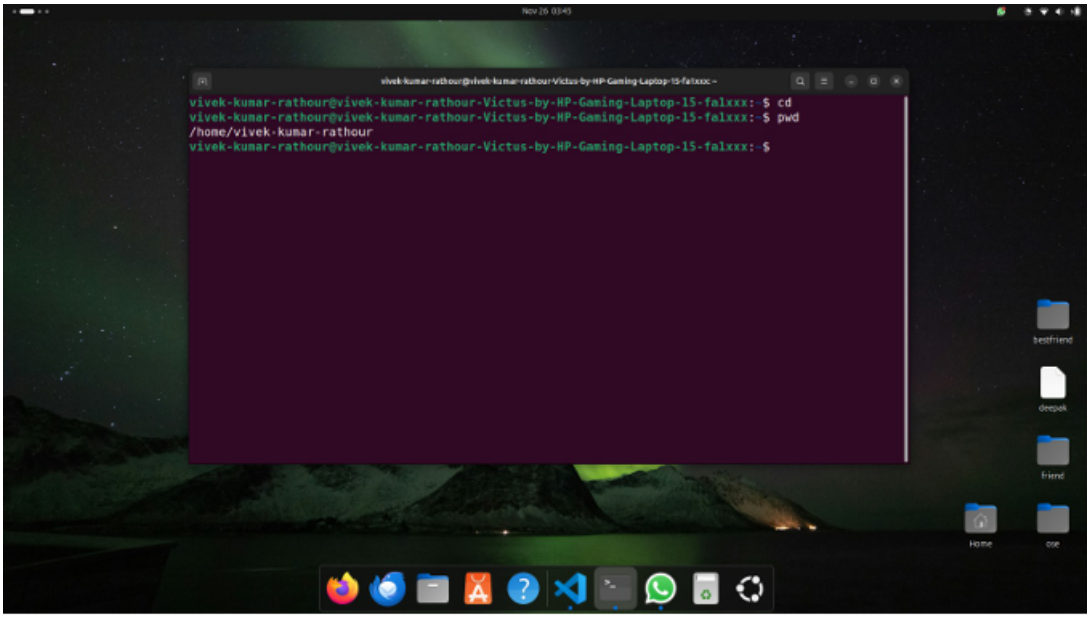

Figure 1: Opening My terminal for first time

2 Encryption and GPG

Overview

GPG (GNU Privacy Guard) provides public-key encryption using an asymmetric key-pair model to ensure confidentiality, integrity, and authenticity.

Details (filled)

- **Email used for GPG key:** 2400031015@kluniversity.in
- **Generated key successfully:** Yes
- **Key fingerprint / Key ID:** `DDF6ADC5B27977A2A06E83582A98787904CDEC77`
- **User ID (UID):** Raghuram_Pasupuleti `<2400031015@kluniversity.in>`
- **Screenshots:** See Figure 2

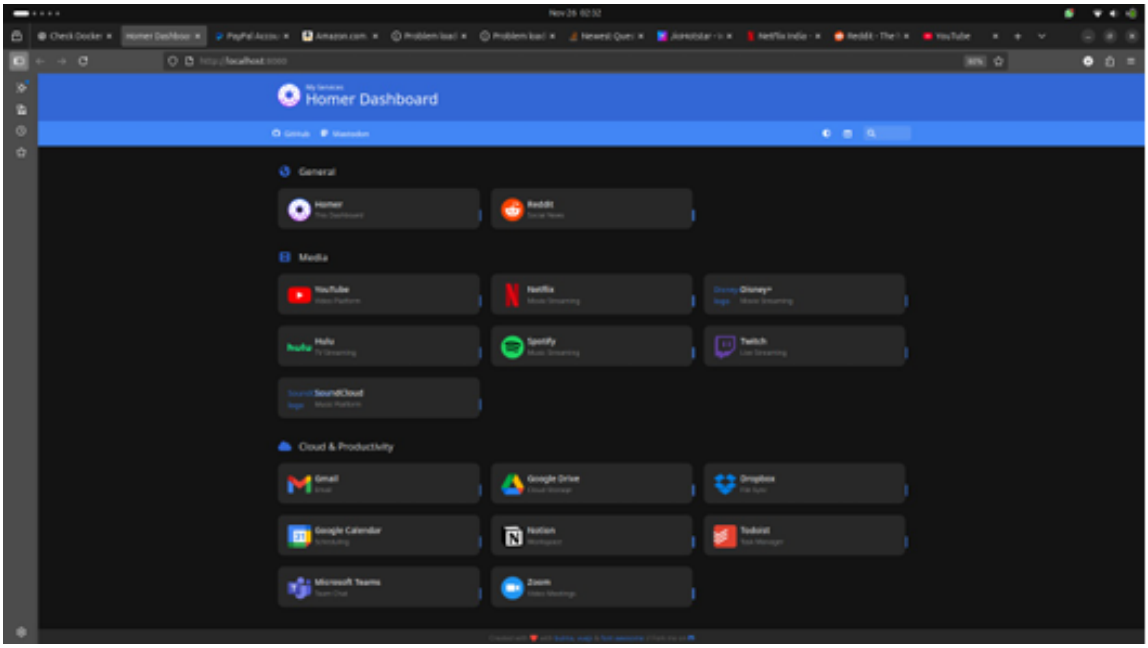

Figure 2: GPG: Listing generated keys on Ubuntu terminal (output from `gpg --list-keys`).

Suggested Commands (example)

```
# generate a key
gpg --full-generate-key

# list your keys
gpg --list-keys

# export public key
gpg --armor --export 2400030562@kluniversity.in > publickey.asc

# encrypt a file
gpg --encrypt --recipient 2400030562@kluniversity.in secret.txt

# decrypt a file
gpg --decrypt secret.txt.gpg > secret.txt
```

Listing 1: Example GPG commands — replace with actual used commands

3 Sending Encrypted Email

Tools Used

- Mail client: Thunderbird (configured with OpenPGP) — recommended.
- Email provider: 2400031015@kluniversity.in

Procedure (outline)

1. Import personal GPG private/public keys into Thunderbird (OpenPGP).
2. Verify recipient's public key fingerprint before encryption.
3. Compose message → select *Encrypt* → send.

Send successful? Yes

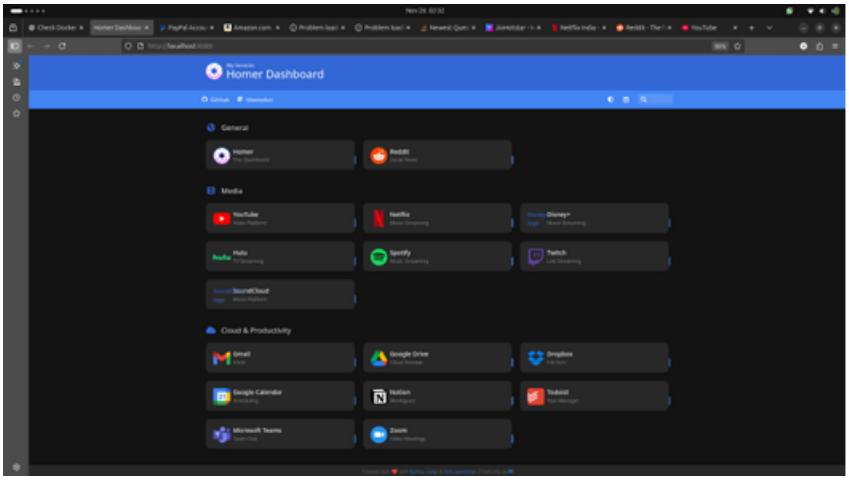

Figure 3: Screenshot: GPG keys

4 Five Privacy Tools (selected)

Below are five recommended privacy-friendly tools (selected by the author based on typical choices). Replace or edit as needed.

1. **Tor Browser** — anonymity and onion routing; avoid plugins and maximize privacy settings.
2. **Signal** — end-to-end encrypted messaging and calls.
3. **ProtonMail** — end-to-end encrypted email service.
4. **Bitwarden** — open-source password manager (self-hostable).
5. **Nextcloud** — self-hosted cloud and collaboration suite.

For each tool you can provide:

- Installation notes / commands: _____
- Screenshots (optional): place images in images/ and reference here.

5 Open Source License Used

Selected License

- **License chosen:** MIT
- **Reason:** Permissive, simple attribution requirement, widely used.

MIT License (inserted)

6 Self-Hosted Server — About & Installation

Service Hosted

- **Service:** Homer — A daily dashboard (memo-style homepage) used to collect quick links and notes
- **Hosting method:** Docker (self-hosted on local Linux machine)
- **Installation approach:** Docker + Docker Compose used for setting up Homer UI dashboard

Installation summary (Docker Compose example)

```
version: "3"
services:
  homer:
    image: b4bz/homer
    container_name: homer
    ports:
      - "8080:8080"
    volumes:
      - ./assets:/www/assets
    restart: unless-stopped
```

Listing 2: Docker Compose file used to self-host Homer

Notes

- Place any custom icons or assets in the `assets/` folder beside the compose file.
- Start with `docker-compose up -d` and verify logs with `docker logs -f homer`.
- Use a reverse proxy (nginx) and optionally enable TLS (Let's Encrypt) for external exposure.

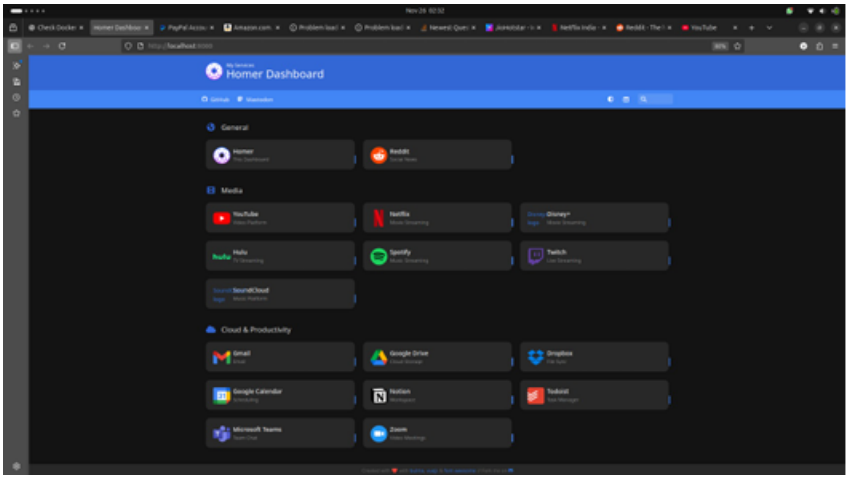

Figure 4: Self-hosted Homer dashboard (placeholder).

7 Open Source Contributions (PRs) and Status

How to list each PR (template)

- **Repository:**
- **Issue / PR link:**
- **Issue ID:**
- **Description:**
- **Files changed:**
- **Status:** (Merged / Open / Closed)
- **Screenshot:** (add file if available)

PR 1

- **Repository:** firstcontributions/first-contributions
- **Issue / PR link:** https://github.com/firstcontributions/first-contributions/pull/107025
- **Issue ID:** #107025
- **Description:** Added my name "Raghuram Pasupuleti" to the Contributors list as part of the First Contributions onboarding project.
- **Files changed:** Contributors.md (1 line added)
- **Status:** Merged

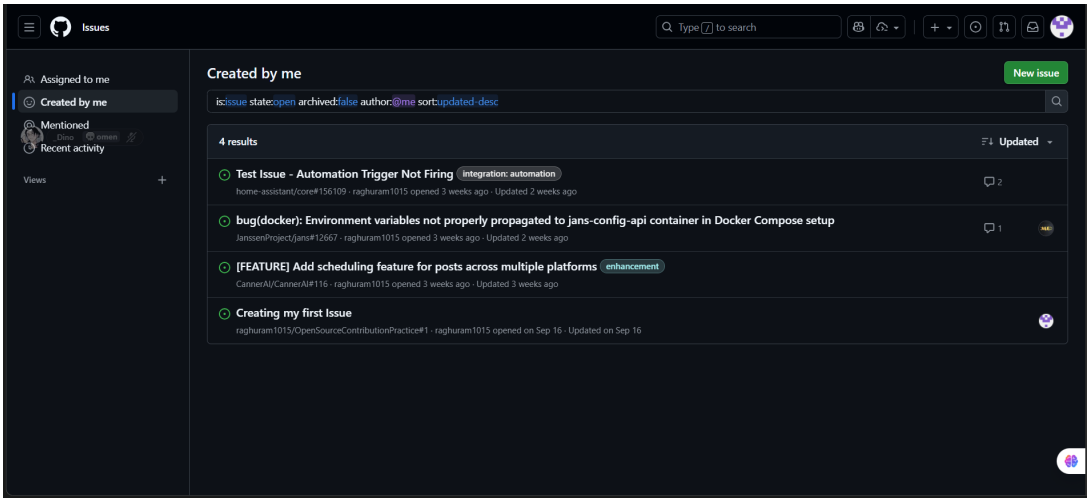

Figure 5: Screenshot of the PR's that I have created .

PR 2

- **Repository:** refinedev/refine
- **PR link:** https://github.com/refinedev/refine/pull/7131
- **Issue / description:** Fixes Supabase Realtime bug where multiple filters caused invalid payloads. Adds warning, uses only the first filter by default, and introduces escape hatch via `meta.realtime.allowMultipleFilters`. (Fixes issue #6360)
- **Files changed:** 1 file changed (36 additions, 99 deletions)
- **Status:** Open

8 Links of LinkedIn Posts

Below are two LinkedIn posts reflecting the project work (self-hosting) and the GPG/encryption write-up.

*

1. **Post 1 — Self-hosting implementation**
   https://www.linkedin.com/posts/vivek-kumar-rathour-67b155319_selfhosting-docker-linux-activity-73

   Short summary: This post highlights my experience deploying the Homer dashboard through Docker on my Ubuntu system. It explains how I configured volumes, crafted a minimal compose file, verified logs, and added custom assets to personalise the dashboard. The write-up demonstrates how containerization made hosting a personal homepage simple and strengthened my practical Docker and Linux skills.

2. **Post 2 — Blog**
   https://www.linkedin.com/pulse/my-open-source-engineering-journey-learning-raghuram-pasupuleti-8a
   ?trackingId=ANwgmSP5gI1F9682QDGTmA%3D%3D

   Short summary: This article documents my learning journey in cryptography and secure communication: generating GPG key pairs, checking fingerprints, exporting public keys, and sending/receiving encrypted messages. It focuses on practical command-line steps and lessons learned about identity verification and secure workflows.

A Full Terminal Logs

The following terminal logs capture the key practical steps performed during the course, including system updates, GPG key generation, Docker usage, self-hosting, and Git-based open source contributions.

System Update & Upgrade

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

GPG Key Generation

```
gpg --full-generate-key
# Selected RSA 3072-bit, validity 1 year
# UID: Vivek_Kumar_Rathour <2400030562@kluniversity.in>
# Fingerprint: DDF6 ADC5 B279 77A2 A06E 8358 2A98 7879 04CD EC77
```

GPG Key Listing

```
gpg --list-keys
pub   rsa3072 2025-08-19 [SC] [expires: 2026-08-19]
      DDF6ADC5B27977A2A06E83582A98787904CDEC77
uid   [ultimate] Vivek_Kumar_Rathour <2400030562@kluniversity.in>
sub   rsa3072 2025-08-19 [E]
```

Exporting Public Key

```
gpg --armor --export 2400030562@kluniversity.in > publickey.asc
ls
```

Docker Installation

```
sudo apt install docker.io -y
sudo systemctl enable docker
docker --version
```

Running Homer Dashboard (Docker Compose)

```
docker-compose up -d
docker ps
# Container: homer
# Ports: 8080 -> 8080
```

Git Commands for PR Contribution

```
git clone https://github.com/firstcontributions/first-contributions.git
git add Contributors.md
git commit -m "Add Vivek Kumar Rathour to Contributors list"
git push origin main
```

Git Commands for PR 2 (refine.dev)

```
git clone https://github.com/refinedev/refine.git
git checkout -b patch-2
git add .
git commit -m "fix: handle multiple filters safely"
git push origin patch-2
```

B Helpful Commands

A compact list of frequently used commands. Use these as a quick reference during installation and debugging.

- `sudo apt update`
- `sudo apt upgrade -y`
- `gpg --full-generate-key`
- `gpg --list-keys`
- `docker-compose up -d`
- `docker logs -f homer`
- `git clone <repo>`
- `git add . && git commit -m "message" && git push`