



**EXPERIENTIAL LEARNING & GLOBAL ENGAGEMENT**

## **Open Source Engineering Report**

**Student Name:** Sravani Gajula(2400060027)

**Branch:** B.Tech – Electrical and electronics Engineering (EEE)

**Course:** Open Source Engineering

**Academic Year:** 2025–2026

**Submitted To:**

Dr. Arunekumar Bala / EL&GE / KL University

# **Contents**

<b>1 About Linux Distro Used: Ubuntu</b>	<b>2</b>
<b>2 Encryption and GPG</b>	<b>2</b>
<b>3 Sending Encrypted Email</b>	<b>3</b>
<b>4 Privacy Tools (PRISM-BREAK)</b>	<b>3</b>
<b>5 Open Source License Used-MIT LICENSE</b>	<b>4</b>
<b>6 Self Hosted Server -UPTIME KUMA</b>	<b>4</b>
<b>7 Open Source Contributions</b>	<b>6</b>
<b>8 LinkedIn Posts</b>	<b>10</b>

# 1 About Linux Distro Used: Ubuntu

Ubuntu is one of the most popular Linux distributions used by developers, students and beginners. It is based on Debian and is known for its stability, regular updates and a friendly graphical interface. Ubuntu is widely used in software development, cloud computing and open-source learning labs.

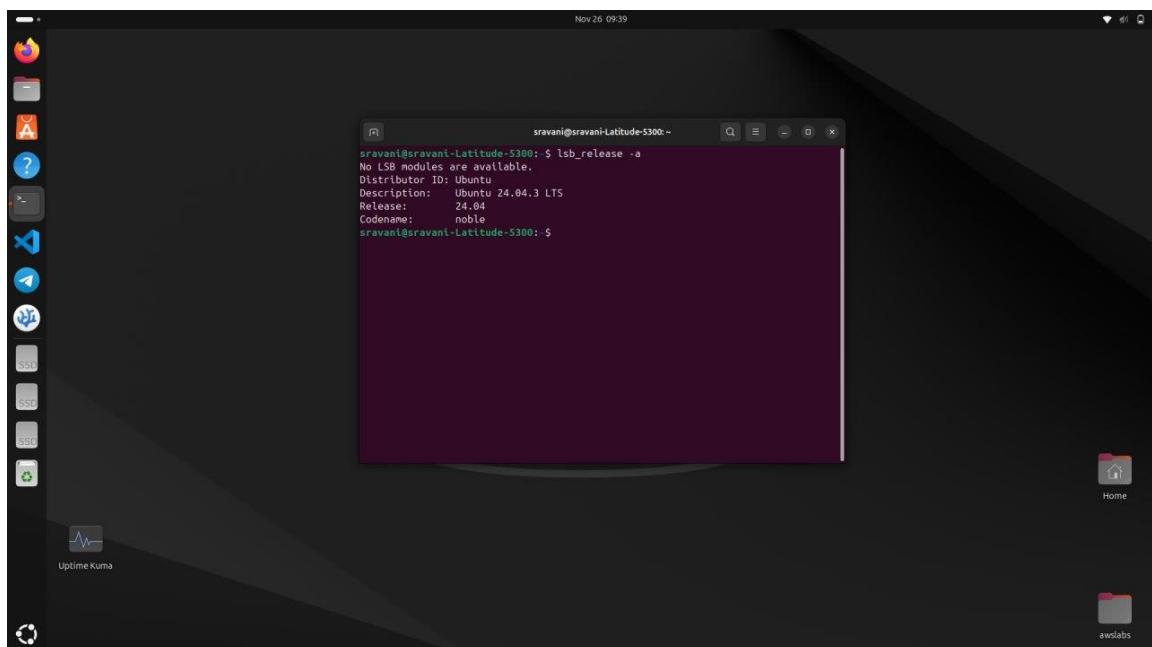
Ubuntu provides thousands of free and open-source packages through the apt package manager. Using simple commands, we can install compilers, editors, servers and security tools. This makes it a very good choice for students who are just starting with Linux.

A key advantage of Ubuntu is its Long-Term Support (LTS) releases. LTS versions receive security and bug fix updates for five years, so they are trusted by companies and universities. Most major cloud platforms like AWS, Azure and Google Cloud support Ubuntu images by default.

In this course, Ubuntu helped me learn:

- Basic terminal commands for navigation and file handling
- Installing and updating software using apt
- Managing users, permissions and executable files
- Using Git and GitHub directly from the terminal
- Running and testing self-hosted services such as HedgeDoc

Overall, Ubuntu gave me a strong foundation in using Linux as a development environment for open source engineering.



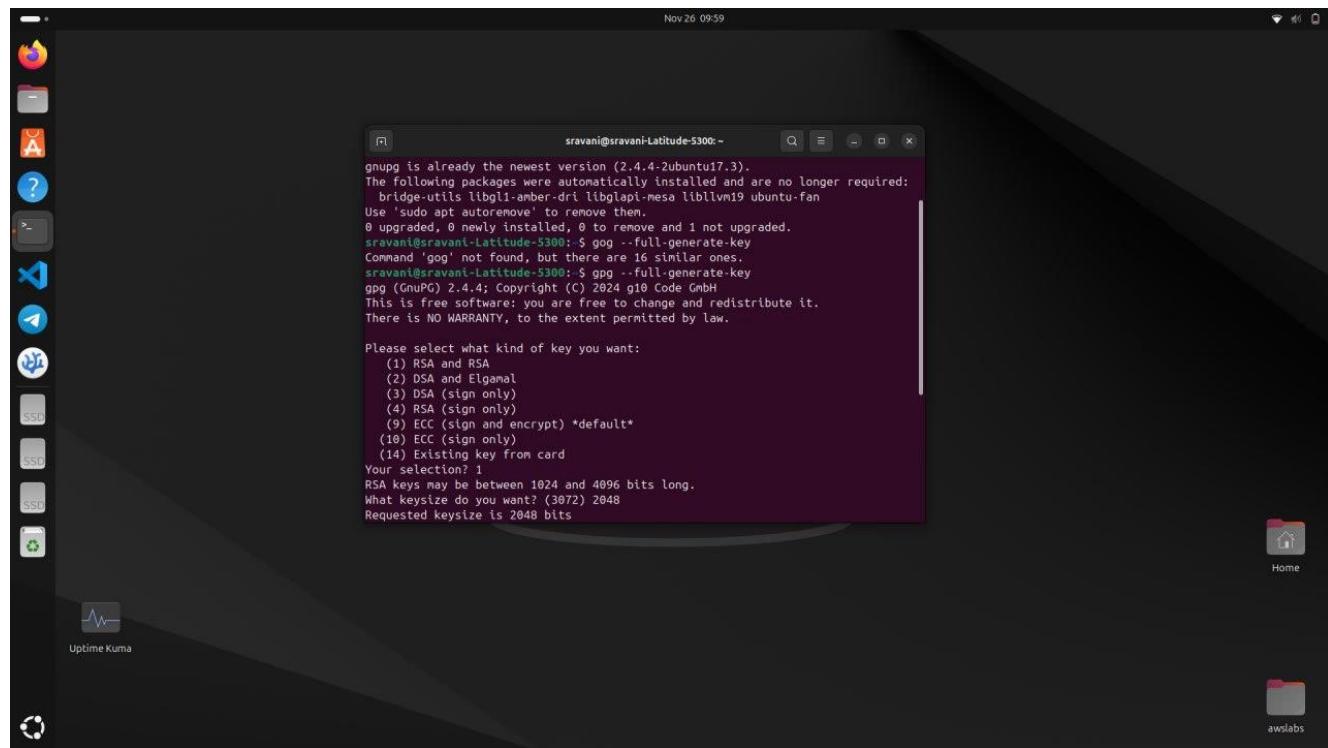
## 2 Encryption and GPG

GNU Privacy Guard (GPG) is a free and open-source implementation of the OpenPGP standard. It is used for encrypting files, signing data and verifying signatures. The main idea is public-key cryptography: each user has a **public key** (can be shared) and a **private key** (kept secret).

When someone wants to send us a secret message, they encrypt it with our public key. Only our private key can decrypt that message. In the same way, if we sign a file with our private key, others can verify the signature with our public key and confirm that it really came from us and has not been modified.

### Common GPG Commands

- `gpg --full-generate-key` – Generate a new key pair (public + private)
- `gpg --list-keys` – Show the public keys stored in our keyring
- `gpg --export --armor > publickey.asc` – Export our public key so that we can share it
- `gpg --encrypt --recipient <email> file.txt` – Encrypt file.txt for a specific user



- `gpg --decrypt file.txt.gpg` – Decrypt an encrypted file using our private key

In the lab we practised generating keys, exporting the public key and encrypting and decrypting sample files. This helped me understand how many open-source projects sign their releases and how users can verify authenticity.

## 3 Sending Encrypted Email

Normal email is like sending a postcard: anyone on the path can read the content. To protect privacy, we can combine email with GPG encryption. For this we can use tools such as Thunderbird with built-in OpenPGP support or browser plugins like Mailvelope.

### Steps for Encrypted Email

- Both sender and receiver generate their own GPG key pairs.
- Each person shares their **public key** with the other, usually as a .asc file or via a key server.
- In the email client, we import the other person's public key and mark it as trusted.
- While composing a mail, we select the option “Encrypt” (and optionally “Sign”).
- The email body is encrypted with the recipient's public key and sent over the internet.
- The recipient opens the mail, enters their passphrase and decrypts the message using their private key.

This activity showed me how encryption is used in real life for secure communication and how public-key infrastructure works beyond theory.

## 4 Privacy Tools (PRISM-BREAK)

PRISM-BREAK is a community-driven website that lists privacy-respecting alternatives to many popular services. Its goal is to help users avoid mass surveillance and tracking by using open-source and decentralised software.

Some tools we explored are:

- **Brave Browser** – A privacy-focused web browser that blocks ads and trackers by default, supports Tor integration, and rewards users with cryptocurrency for opting into privacy-respecting ads.
- **ProtonMail** – An end-to-end encrypted email service based in Switzerland, offering zero-access encryption where even the service provider cannot read user emails.
- **Bitwarden** – An open-source password manager that stores credentials in an encrypted vault, supports self-hosting, and offers cross-platform synchronization.

- **Nextcloud** – A self-hosted cloud storage and collaboration platform that provides file sync, calendar, contacts, and document editing as alternatives to Google Drive and Dropbox.
- **Element (Matrix)** – A decentralized, end-to-end encrypted messaging platform based on the Matrix protocol, allowing users to communicate across different servers.

These examples helped me see that privacy is not only a theory topic. There are real open-source tools available for almost every daily use-case.

## 5 Open Source License Used – MIT LICENSE

The MIT License is a highly permissive open-source license widely used in modern software projects. It prioritizes freedom, flexibility, and ease of reuse, making it extremely popular among developers and organizations.

- Allows anyone to freely use, copy, modify, merge, publish, distribute, and sublicense the software
- Permits integration into both open-source and closed-source/proprietary systems
- Requires only a simple copyright notice and license text to be included in copies
- Imposes no copyleft requirements — modifications do not need to be open-sourced
- Encourages rapid innovation, commercial use, and wide adoption

The MIT License is commonly chosen for libraries, frameworks, and tools where maximum flexibility and minimal restrictions are desired

## 6 Self Hosted Server – Uptime kuma

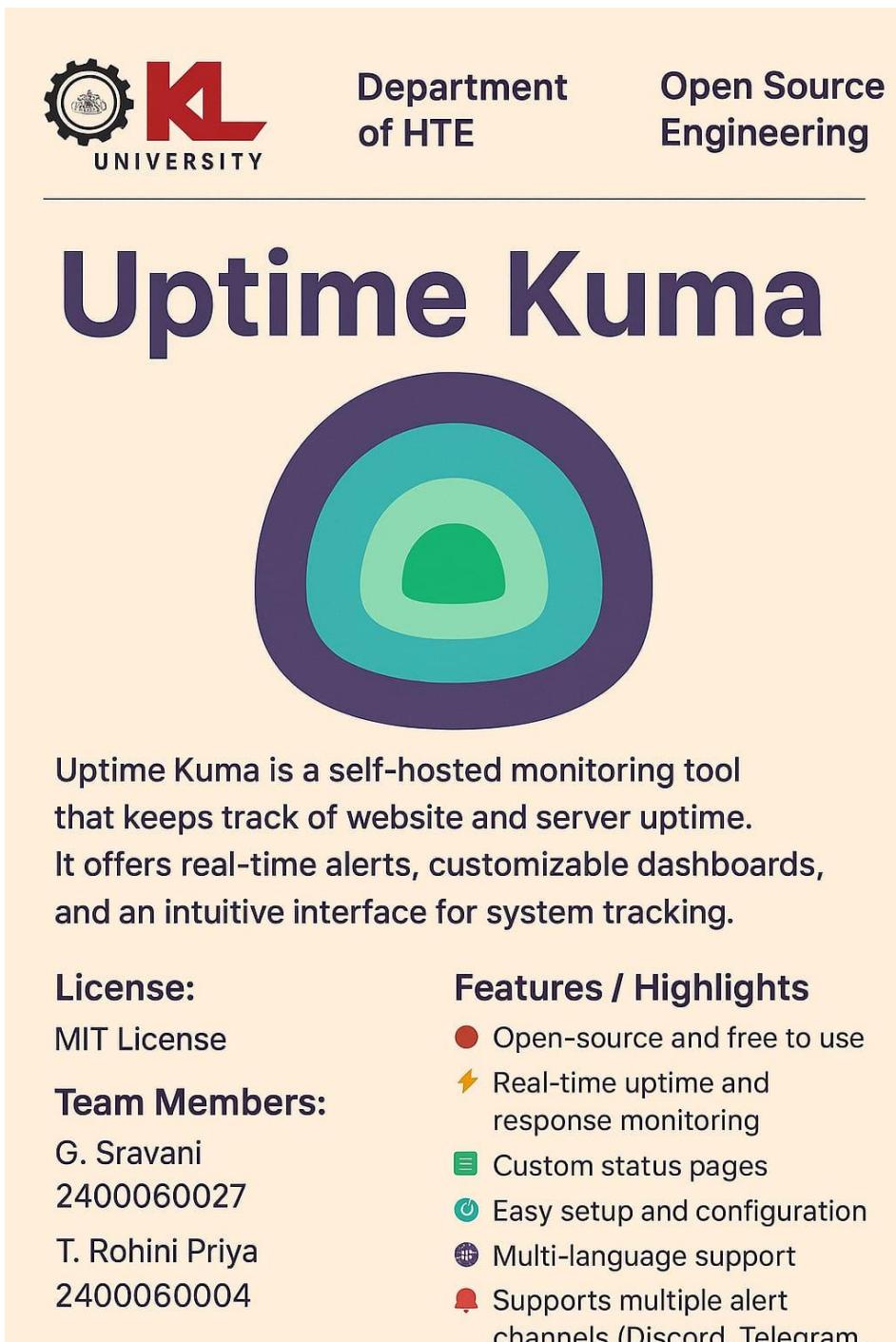
Uptime Kuma is an open-source, self-hosted monitoring tool used to track the uptime and performance of websites, APIs, servers, and network services. It provides easy-to-read status dashboards and instant notifications, making popular for personal and team infrastructure monitoring.

### Features

- Simple, user-friendly dashboard for monitoring uptime
- Supports HTTP(s), TCP, Ping, DNS, Docker containers, and more
- Customizable status pages for public or private use
- Multiple notification integrations (Telegram, Discord, Email, Slack, etc.)
- Real-time alerts when a service goes down or recovers
- Easy to deploy on Docker, Linux servers, or local machines

- Regular health checks with detailed history and logs

## POSTER :

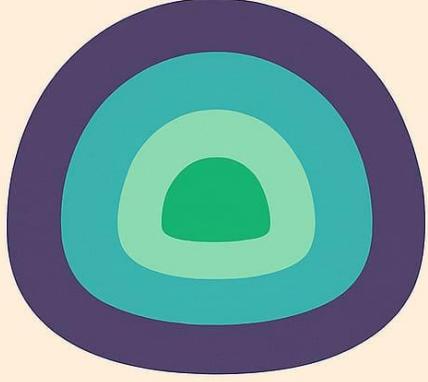


The poster features a light beige background. At the top left is the KL University logo, which includes a gear icon and the text "KL UNIVERSITY". To the right of the logo are two columns of text: "Department of HTE" and "Open Source Engineering". Below this section is a large, bold title "Uptime Kuma" in dark blue. Underneath the title is a large, stylized graphic consisting of three concentric circles in shades of purple, teal, and green. To the right of the graphic is a descriptive paragraph about Uptime Kuma. Below this paragraph are two sections: "License:" followed by "MIT License" and "Team Members:" followed by the names and IDs of two team members. To the right of these sections is a "Features / Highlights" section with a bulleted list of seven items, each preceded by a small colored icon.

**Department of HTE**

**Open Source Engineering**

# Uptime Kuma



Uptime Kuma is a self-hosted monitoring tool that keeps track of website and server uptime. It offers real-time alerts, customizable dashboards, and an intuitive interface for system tracking.

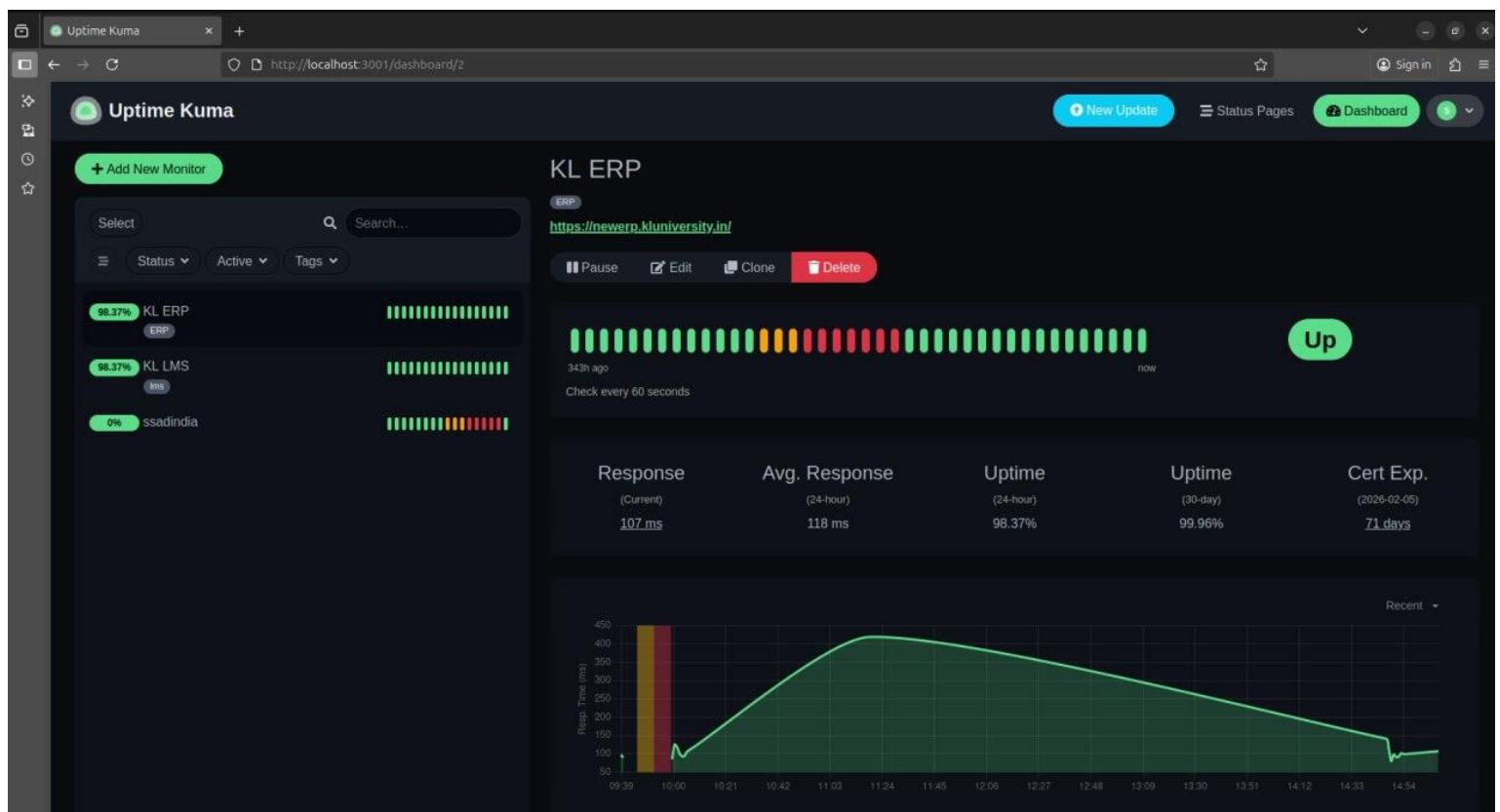
**License:**  
MIT License

**Team Members:**  
G. Sravani  
2400060027  
T. Rohini Priya  
2400060004

**Features / Highlights**

- Open-source and free to use
- ⚡ Real-time uptime and response monitoring
- ☰ Custom status pages
- ⌚ Easy setup and configuration
- 🌐 Multi-language support
- 🔔 Supports multiple alert channels (Discord, Telegram,

# Server Screenshot



## Open Source Contributions

GitHub Username: **Sravz0408**

In this course we were asked to contribute to real open-source projects. The following is a list of my successfully merged pull requests:

### List of Pull Requests

- **PR1: firstcontributions / first-contributions** – “Add simple spellchecker in c”

I developed a simple spellchecker in C that loads a dictionary and checks whether an input word is correct. It uses a hash table for fast lookup and gives basic suggestions for misspelled words. This mini-project helped me understand file handling, hashing, and simple text-processing concepts.

A screenshot of a GitHub pull request page. The title is "feat: add simple spellchecker in C #2". The status bar indicates it is merged. The commit history shows a single commit from "Sravz0408" merged into "final-pr" on Oct 21. The conversation shows a comment from Sravz0408 about adding a C-based spellchecker and fixing the short description. The merge message says "Pull request successfully merged and closed". The right sidebar shows review stats (18 reviews), assignees (Sravz0408), labels (None yet), projects (None yet), milestones (None yet), development (None yet), and notifications (Unsubscribe).

## • PR2: KLGLUG/Y24OpenSourceEngineering – “Add function to generate valid parentheses combinations”

I implemented a function in C to generate all valid combinations of parentheses for a given number n. The logic uses recursion and backtracking: adding an opening bracket when allowed and adding a closing bracket only when it keeps the expression valid. This helped me understand recursive programming, constraints handling, and how backtracking is used to generate structured outputs efficiently.

A screenshot of a GitHub pull request page. The title is "Add function to generate valid parentheses combinations #1". The status bar indicates it is merged. The commit history shows a single commit from "Sravz0408" merged into "master" from "Sravz0408-patch-1" on Oct 27. The conversation shows a comment from Sravz0408 about implementing the function using backtracking, and another comment adding the "hacktoberfest-accepted" label. The merge message says "Pull request successfully merged and closed". The right sidebar shows review stats (37 reviews), assignees (Sravz0408), labels (hacktoberfest-accepted), projects (None yet), milestones (None yet), development (None yet), and notifications (Unsubscribe).

## 8 LinkedIn Posts

I have shared my open-source journey on LinkedIn. These posts helped me explain my learning and connect with other developers.

- Self Hosting Post –

[https://www.linkedin.com/posts/sravani-gajula-17b974339\\_opensource-selfhosting-upmekuma-activity-7399342275996274688-cfbE?utm\\_source=social\\_share\\_send&utm\\_medium=android\\_app&rcm=ACoAAFUJzu8BhoMr\\_SfZpN\\_c6OiN0Q5onVKFNak&utm\\_campaign=copy\\_link](https://www.linkedin.com/posts/sravani-gajula-17b974339_opensource-selfhosting-upmekuma-activity-7399342275996274688-cfbE?utm_source=social_share_send&utm_medium=android_app&rcm=ACoAAFUJzu8BhoMr_SfZpN_c6OiN0Q5onVKFNak&utm_campaign=copy_link)

- PR Merge Post –

[https://www.linkedin.com/posts/sravani-gajula-17b974339\\_hacktoberfest-opensource-klu-activity-7390931717794205696-GvBA?utm\\_source=social\\_share\\_send&utm\\_medium=android\\_app&rcm=ACoAAFUJzu8BhoMr\\_SfZpN\\_c6OiN0Q5onVKFNak&utm\\_campaign=copy\\_link](https://www.linkedin.com/posts/sravani-gajula-17b974339_hacktoberfest-opensource-klu-activity-7390931717794205696-GvBA?utm_source=social_share_send&utm_medium=android_app&rcm=ACoAAFUJzu8BhoMr_SfZpN_c6OiN0Q5onVKFNak&utm_campaign=copy_link)

- Blog Post –

[https://www.linkedin.com/posts/sravani-gajula-17b974339\\_opensource-selfhosting-upmekuma-activity-7399342275996274688-cfbE?utm\\_source=social\\_share\\_send&utm\\_medium=android\\_app&rcm=ACoAAFUJzu8BhoMr\\_SfZpN\\_c6OiN0Q5onVKFNak&utm\\_campaign=copy\\_link](https://www.linkedin.com/posts/sravani-gajula-17b974339_opensource-selfhosting-upmekuma-activity-7399342275996274688-cfbE?utm_source=social_share_send&utm_medium=android_app&rcm=ACoAAFUJzu8BhoMr_SfZpN_c6OiN0Q5onVKFNak&utm_campaign=copy_link)