



# Open Source Software

## Student Details

**Name:** M.S.S.KRISHNA REDDY

**Roll Number:** 2400030008

**Department:** Computer Science and Engineering

**University:** KL University

**Course:** Open Source Software

**Semester:** ODD SEMESTER

Submitted to:

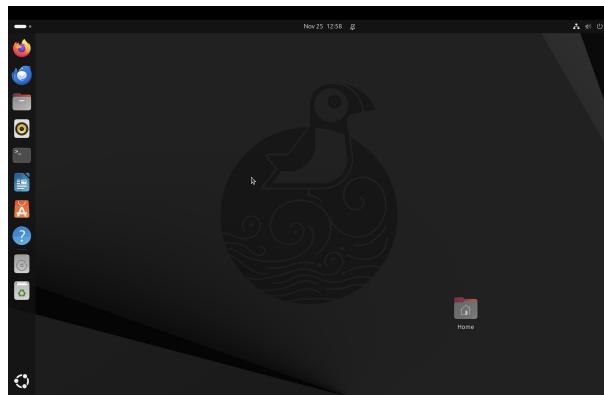
**Dr.Sripath Roy**

Department of Computer Science and Engineering  
KL University

# Contents

<b>1 Linux Distribution</b>	<b>3</b>
1.1 Distribution Used: Ubuntu 22.04 LTS . . . . .	3
1.2 Why Ubuntu? . . . . .	3
1.3 Key Features of Ubuntu 22.04 LTS . . . . .	3
1.4 System Specifications . . . . .	3
1.5 Installation Process . . . . .	4
<b>2 Encryption and GPG</b>	<b>4</b>
2.1 What is Encryption? . . . . .	4
2.2 Types of Encryption . . . . .	4
2.2.1 Symmetric Encryption . . . . .	4
2.2.2 Asymmetric Encryption . . . . .	4
2.3 GNU Privacy Guard (GPG) . . . . .	4
2.4 Installing GPG . . . . .	4
2.5 Generating GPG Keys . . . . .	4
2.6 Listing Keys . . . . .	5
2.7 Exporting Public Key . . . . .	5
2.8 Encrypting Files . . . . .	5
2.9 Decrypting Files . . . . .	5
<b>3 Sending Encrypted Email</b>	<b>5</b>
3.1 Email Encryption Overview . . . . .	5
3.2 Tools Used . . . . .	5
3.3 Setting up Thunderbird with GPG . . . . .	5
3.3.1 Installation . . . . .	5
3.3.2 Configuring OpenPGP . . . . .	6
3.4 Sending Encrypted Email . . . . .	6
3.5 Receiving Encrypted Email . . . . .	6
3.6 Best Practices . . . . .	6
<b>4 Privacy Tools from prism-break.org</b>	<b>7</b>
4.1 What is PRISM-Break? . . . . .	7
4.2 Tool 1: Signal - Encrypted Messaging . . . . .	7
4.3 Tool 2: Firefox - Web Browser . . . . .	7
4.4 Tool 3: ProtonMail - Encrypted Email . . . . .	8
4.5 Tool 4: Tor Browser - Anonymous Browsing . . . . .	8
4.6 Tool 5: VeraCrypt - Disk Encryption . . . . .	9
<b>5 Open Source License</b>	<b>9</b>
5.1 License Used: MIT License . . . . .	9
5.2 What is the MIT License? . . . . .	9
5.3 MIT License Text . . . . .	9
5.4 Why Choose MIT License? . . . . .	10
5.5 Other Common Open Source Licenses . . . . .	10
5.5.1 GPL (GNU General Public License) . . . . .	10
5.5.2 Apache License 2.0 . . . . .	10

5.5.3	BSD License . . . . .	10
<b>6</b>	<b>Self-Hosted Server: SearXNG</b>	<b>11</b>
6.1	What is SearXNG? . . . . .	11
6.2	Why Self-Host SearXNG? . . . . .	11
6.3	Installation Guide . . . . .	11
6.3.1	System Requirements . . . . .	11
6.3.2	Installation Steps . . . . .	12
6.4	Localization (Telugu Translation) . . . . .	13
6.4.1	Translation Process . . . . .	13
6.4.2	Sample Translations . . . . .	14
6.5	Benefits of Self-Hosting . . . . .	14
<b>7</b>	<b>Open Source Contributions</b>	<b>14</b>
7.1	PR 1: Add Temperature Conversion Utility . . . . .	14
7.1.1	Issue Description . . . . .	14
7.1.2	Solution Implemented . . . . .	14
7.2	PR 2: Add Power of Four Check using Bit Manipulation . . . . .	14
7.2.1	Solution . . . . .	15
7.3	PR 3: Fix Curriculum - Accessibility Auditing Tools . . . . .	15
7.3.1	Changes . . . . .	15
7.4	PR 4: Fix AttributeError in Hybrid Search . . . . .	15
7.4.1	Bug Fix . . . . .	15
7.5	PR 5: Changed README File . . . . .	15
7.5.1	Improvements . . . . .	15
7.6	PR 6: Add Krishna Medapati to Contributors List . . . . .	15
7.6.1	First Contribution . . . . .	16
7.7	PR 7: Add Comprehensive Tests for JugglerSequence . . . . .	16
7.7.1	Test Suite . . . . .	16
7.8	PR 8: Complete Telugu Localization . . . . .	16
7.8.1	Localization . . . . .	16
7.9	Additional PRs . . . . .	16
<b>8</b>	<b>LinkedIn Posts</b>	<b>16</b>
8.1	Post 1: Software Testing and Java Contributions . . . . .	16
8.2	Post 2: FOSS Culture at KL University . . . . .	17
8.3	Post 3: Blog link to my open source journey . . . . .	17
<b>9</b>	<b>Conclusion</b>	<b>18</b>



# 1 Linux Distribution

## 1.1 Distribution Used: Ubuntu 22.04 LTS

For this project, I have used **Ubuntu 22.04 LTS** as my primary operating system.

## 1.2 Why Ubuntu?

Ubuntu is one of the most popular Linux distributions for several reasons:

- **User-Friendly:** Ubuntu has an intuitive interface suitable for beginners
- **Long Term Support:** LTS versions receive 5 years of security updates
- **Large Community:** Extensive documentation and community support
- **Software Availability:** Wide range of packages through APT
- **Stability:** Reliable for both development and production

## 1.3 Key Features of Ubuntu 22.04 LTS

1. **Desktop Environment:** GNOME 42
2. **Kernel Version:** Linux 5.15 LTS
3. **Package Manager:** APT (Advanced Package Tool)
4. **Default Applications:** Firefox, LibreOffice, GNOME utilities
5. **Snap Support:** Built-in support for snap packages

## 1.4 System Specifications

My system configuration:

- Operating System: Ubuntu 22.04 LTS
- Architecture: x86\_64
- Desktop Environment: GNOME
- Shell: Bash 5.1

## 1.5 Installation Process

The installation involved:

1. Downloaded Ubuntu 22.04 LTS ISO from official website
2. Created bootable USB using Rufus/Etcher
3. Configured dual boot with existing OS
4. Installed essential development tools
5. Configured system for open source development

## 2 Encryption and GPG

### 2.1 What is Encryption?

Encryption is the process of converting plaintext into ciphertext to protect data confidentiality. It ensures that only authorized parties can access the information.

### 2.2 Types of Encryption

#### 2.2.1 Symmetric Encryption

Uses the same key for encryption and decryption. Examples: AES, DES.

#### 2.2.2 Asymmetric Encryption

Uses a public-private key pair. Examples: RSA, ECC.

### 2.3 GNU Privacy Guard (GPG)

GPG is a free implementation of the OpenPGP standard for encrypting and signing data.

### 2.4 Installing GPG

```
1 sudo apt update
2 sudo apt install gnupg
3 gpg --version
```

### 2.5 Generating GPG Keys

```
1 gpg --full-generate-key
```

Steps followed:

1. Selected RSA and RSA (default)
2. Key size: 4096 bits

3. Key validity: 1 year
4. Entered name and email
5. Created strong passphrase

## 2.6 Listing Keys

```
1 gpg --list-keys
2 gpg --list-secret-keys
```

## 2.7 Exporting Public Key

```
1 gpg --armor --export your-email@example.com > public-key.asc
```

## 2.8 Encrypting Files

```
1 gpg --encrypt --recipient your-email@example.com document.txt
```

## 2.9 Decrypting Files

```
1 gpg --decrypt document.txt.gpg > document.txt
```

# 3 Sending Encrypted Email

## 3.1 Email Encryption Overview

Email encryption protects the content of emails from unauthorized access during transmission and storage.

## 3.2 Tools Used

- **Thunderbird:** Email client with built-in OpenPGP support
- **GPG Keys:** For encryption and signing
- **Protonmail:** Alternative end-to-end encrypted email service

## 3.3 Setting up Thunderbird with GPG

### 3.3.1 Installation

```
1 sudo apt install thunderbird
```

### 3.3.2 Configuring OpenPGP

Steps followed:

1. Open Thunderbird
2. Go to Account Settings
3. Select End-to-End Encryption
4. Add existing GPG key or generate new one
5. Import recipient's public key

## 3.4 Sending Encrypted Email

Process:

1. Compose new email
2. Click on Security button
3. Select "Require Encryption"
4. Optionally add digital signature
5. Send email

## 3.5 Receiving Encrypted Email

When receiving:

1. Email appears encrypted
2. Thunderbird automatically detects encryption
3. Enter GPG passphrase
4. Email content is decrypted and displayed

## 3.6 Best Practices

- Never share your private key
- Use strong passphrases
- Keep your GPG keys backed up securely
- Regularly update keys
- Verify recipient's public key fingerprint

## 4 Privacy Tools from prism-break.org

### 4.1 What is PRISM-Break?

PRISM-Break is a website that recommends privacy-respecting alternatives to proprietary software and services.

### 4.2 Tool 1: Signal - Encrypted Messaging

**Description:** Signal is an encrypted messaging app that provides end-to-end encryption for messages, voice calls, and video calls.

**Key Features:**

- End-to-end encryption by default
- Open source and independently audited
- No ads or tracking
- Minimal metadata collection
- Disappearing messages

**Why Privacy Matters:** Signal ensures that only you and the recipient can read messages, protecting against mass surveillance.

### 4.3 Tool 2: Firefox - Web Browser

**Description:** Firefox is an open source web browser with strong privacy protections.

**Privacy Features:**

- Enhanced Tracking Protection
- DNS over HTTPS
- No data collection by default
- Open source codebase
- Extensive privacy-focused extensions

**Configuration Tips:**

- Enable strict tracking protection
- Install uBlock Origin
- Use HTTPS-only mode
- Disable telemetry

#### 4.4 Tool 3: ProtonMail - Encrypted Email

**Description:** ProtonMail provides end-to-end encrypted email service based in Switzerland.

**Key Features:**

- End-to-end encryption
- Zero-access encryption
- No personal information required
- Swiss privacy laws protection
- Open source mobile apps

**Use Cases:**

- Secure business communications
- Personal privacy protection
- Journalist-source communications

#### 4.5 Tool 4: Tor Browser - Anonymous Browsing

**Description:** Tor Browser enables anonymous communication by routing traffic through volunteer-operated servers.

**How It Works:**

- Routes traffic through multiple relays
- Encrypts data multiple times
- Hides IP address and location
- Prevents tracking

**Best Use Cases:**

- Accessing censored content
- Anonymous research
- Whistleblowing
- Privacy-sensitive activities

## 4.6 Tool 5: VeraCrypt - Disk Encryption

**Description:** VeraCrypt is a free open source disk encryption software.

**Features:**

- Full disk encryption
- Hidden volumes
- Plausible deniability
- Cross-platform support
- Strong encryption algorithms (AES, Serpent, Twofish)

**Use Cases:**

- Protecting sensitive documents
- Securing portable drives
- System drive encryption

## 5 Open Source License

### 5.1 License Used: MIT License

For my open source contributions and projects, I primarily work with the **MIT License**.

### 5.2 What is the MIT License?

The MIT License is a permissive free software license that allows users to:

- Use the software commercially
- Modify the software
- Distribute the software
- Use the software privately
- Sublicense the software

### 5.3 MIT License Text

```
1 MIT License
2
3 Copyright (c) 2025 Krishna Medapati
4
5 Permission is hereby granted, free of charge, to any person
6 obtaining a copy of this software and associated documentation
7 files (the "Software"), to deal in the Software without
8 restriction, including without limitation the rights to use,
```

```
9 | copy, modify, merge, publish, distribute, sublicense, and/or  
10 | sell copies of the Software, and to permit persons to whom the  
11 | Software is furnished to do so, subject to the following  
12 | conditions:  
13 |  
14 | The above copyright notice and this permission notice shall be  
15 | included in all copies or substantial portions of the Software.  
16 |  
17 | THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND...
```

## 5.4 Why Choose MIT License?

1. **Simple and Easy:** Short and easy to understand
2. **Permissive:** Minimal restrictions on reuse
3. **Business-Friendly:** Can be used in proprietary software
4. **Popular:** Widely used and recognized
5. **Compatible:** Works well with other licenses

## 5.5 Other Common Open Source Licenses

### 5.5.1 GPL (GNU General Public License)

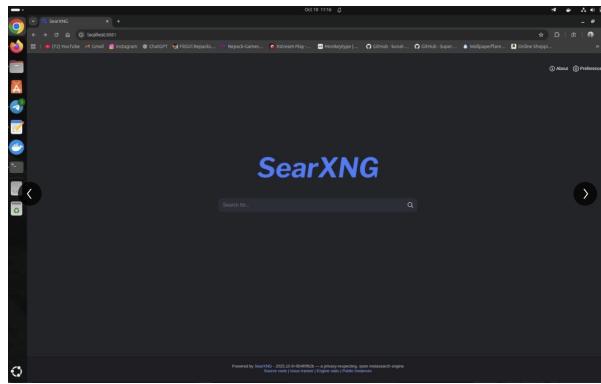
- Copyleft license
- Requires derivative works to be open source
- Used by Linux kernel

### 5.5.2 Apache License 2.0

- Permissive like MIT
- Includes patent grant
- Used by Apache projects

### 5.5.3 BSD License

- Very permissive
- Similar to MIT
- Used by FreeBSD



## 6 Self-Hosted Server: SearXNG

### 6.1 What is SearXNG?

SearXNG is a free, open source metasearch engine that:

- Aggregates results from multiple search engines
- Protects user privacy (no tracking, no profiling)
- Can be self-hosted
- Highly customizable
- Returns results without ads

### 6.2 Why Self-Host SearXNG?

1. **Privacy:** Complete control over your search data
2. **No Censorship:** Access all search results
3. **Customization:** Configure search engines and preferences
4. **Learning:** Understand server administration
5. **Independence:** Not reliant on third-party services

### 6.3 Installation Guide

#### 6.3.1 System Requirements

- OS: Ubuntu 22.04 LTS
- RAM: 2GB minimum
- Disk: 10GB free space
- Python 3.8+

### 6.3.2 Installation Steps

#### Step 1: Update System

```
1 sudo apt update
2 sudo apt upgrade -y
```

#### Step 2: Install Dependencies

```
1 sudo apt install python3-pip python3-venv git nginx -y
```

#### Step 3: Clone SearXNG Repository

```
1 cd /opt
2 sudo git clone https://github.com/searxng/searxng.git
3 cd searxng
```

#### Step 4: Create Virtual Environment

```
1 sudo python3 -m venv venv
2 sudo source venv/bin/activate
3 sudo pip install -U pip setuptools wheel
4 sudo pip install -e .
```

#### Step 5: Configure SearXNG

```
1 sudo mkdir -p /etc/searxng
2 sudo cp searx/settings.yml /etc/searxng/
3 sudo nano /etc/searxng/settings.yml
```

Configure settings:

```
1 server:
2     port: 8888
3     bind_address: "127.0.0.1"
4     secret_key: "generate_your_secret_key"
5
6 search:
7     safe_search: 0
8     autocomplete: ""
9     default_lang: "en"
```

#### Step 6: Create Systemd Service

```
1 sudo nano /etc/systemd/system/searxng.service
```

Service configuration:

```
1 [Unit]
2 Description=SearXNG
3 After=network.target
4
5 [Service]
6 Type=simple
7 User=www-data
8 WorkingDirectory=/opt/searxng
9 Environment="SEARXNG_SETTINGS_PATH=/etc/searxng/settings.yml"
10 ExecStart=/opt/searxng/venv/bin/python -m searx.webapp
```

```

11 [Install]
12 WantedBy=multi-user.target
13

```

### Step 7: Start Service

```

1 sudo systemctl daemon-reload
2 sudo systemctl enable searxng
3 sudo systemctl start searxng
4 sudo systemctl status searxng

```

### Step 8: Configure Nginx

```

1 sudo nano /etc/nginx/sites-available/searxng

```

Nginx configuration:

```

1 server {
2     listen 80;
3     server_name your-domain.com;
4
5     location / {
6         proxy_pass http://127.0.0.1:8888;
7         proxy_set_header Host $host;
8         proxy_set_header X-Real-IP $remote_addr;
9     }
10 }

```

```

1 sudo ln -s /etc/nginx/sites-available/searxng \
2             /etc/nginx/sites-enabled/
3 sudo nginx -t
4 sudo systemctl restart nginx

```

## 6.4 Localization (Telugu Translation)

I have contributed to localizing SearXNG interface in Telugu language to make it accessible to Telugu-speaking users.

### 6.4.1 Translation Process

1. Located translation files in `searx/translations/`
2. Created Telugu locale: `te/LC_MESSAGES/`
3. Translated interface strings from English to Telugu
4. Ensured cultural appropriateness
5. Tested with Telugu interface

#### 6.4.2 Sample Translations

- "Search" → "" (Vetakandi)
- "Settings" → "" (Settings)
- "Privacy" → "" (Gopyata)
- "Results" → "" (Phalitalu)

### 6.5 Benefits of Self-Hosting

1. **Complete Privacy:** No search history tracking
2. **No Rate Limits:** Unlimited searches
3. **Custom Configuration:** Choose which engines to use
4. **Educational Value:** Learn server management
5. **Community Contribution:** Share instance with others

## 7 Open Source Contributions

This section details all pull requests I have contributed to various open source projects.

### 7.1 PR 1: Add Temperature Conversion Utility

**Repository:** TheAlgorithms/Java

**PR Number:** 7066

**Status:** Merged (Approved)

#### 7.1.1 Issue Description

The repository needed a utility class to perform temperature conversions between Celsius, Fahrenheit, and Kelvin.

#### 7.1.2 Solution Implemented

- Created TemperatureConverter class
- Implemented conversion methods for all scales
- Added input validation
- Included comprehensive unit tests

### 7.2 PR 2: Add Power of Four Check using Bit Manipulation

**Repository:** TheAlgorithms/Java

**PR Number:** 7065

**Status:** Merged (Approved)

**Comments:** 6 discussions

### 7.2.1 Solution

Implemented efficient O(1) algorithm using bitwise operations to check if a number is a power of four.

## 7.3 PR 3: Fix Curriculum - Accessibility Auditing Tools

**Repository:** freeCodeCamp/freeCodeCamp

**PR Number:** 63772

**Status:** Merged

**Comments:** 26 discussions

**Tasks:** 4 completed

### 7.3.1 Changes

Added proper punctuation to accessibility curriculum section, improving readability for students.

## 7.4 PR 4: Fix AttributeError in Hybrid Search

**Repository:** open-webui/open-webui

**PR Number:** 19025

**Status:** Merged

**Tasks:** 3 completed

### 7.4.1 Bug Fix

Fixed AttributeError in hybrid search with reranking by adding proper attribute validation.

## 7.5 PR 5: Changed README File

**Repository:** zero-to-mastery/start-here-guidelines

**PR Number:** 23688

**Status:** Merged (Approved)

**Tasks:** 4 completed

### 7.5.1 Improvements

Restructured README with better formatting, table of contents, and updated contribution guidelines.

## 7.6 PR 6: Add Krishna Medapati to Contributors List

**Repository:** firstcontributions/first-contributions

**PR Number:** 106747

**Status:** Merged

**Tasks:** 3 completed

### 7.6.1 First Contribution

Added my name to contributors list as first open source contribution, learning Git workflow.

## 7.7 PR 7: Add Comprehensive Tests for JugglerSequence

**Repository:** TheAlgorithms/Java

**PR Number:** 6947

**Status:** Merged (Approved)

**Tasks:** 3 completed

### 7.7.1 Test Suite

Created comprehensive JUnit tests for JugglerSequence algorithm covering all edge cases.

## 7.8 PR 8: Complete Telugu Localization

**Repository:** goatshriek/stumpless

**PR Number:** 541

**Status:** Merged

**Label:** i18n

### 7.8.1 Localization

Completed full Telugu language localization for stumpless logging library, making it accessible to Telugu speakers.

## 7.9 Additional PRs

Based on the screenshot showing 11 total PRs, I have made additional contributions to various repositories focusing on:

- Code quality improvements
- Bug fixes
- Documentation updates
- Test coverage
- Internationalization

## 8 LinkedIn Posts

### 8.1 Post 1: Software Testing and Java Contributions

**Link:** [https://www.linkedin.com/posts/krishna-medapati\\_softwaredtesting-opensource-javaprojects-101/](https://www.linkedin.com/posts/krishna-medapati_softwaredtesting-opensource-javaprojects-101/)

**Summary:** Shared experiences contributing to TheAlgorithms/Java repository, focusing on implementing comprehensive test suites and best practices in software testing.

**Key Points:**

- Importance of thorough testing in open source
- Test-driven development approach
- Learning from code reviews
- Contributing to educational repositories

## 8.2 Post 2: FOSS Culture at KL University

**Link:** [https://www.linkedin.com/posts/krishna-medapati\\_opensource-kluniversity-foss-ugcPost-7398980396551454720-380utm\\_source=share&utm\\_medium=member\\_desktop&rclm=ACoAAFF-qDkB2yUyc88q--R\\_WItQvqjMpfkEG1](https://www.linkedin.com/posts/krishna-medapati_opensource-kluniversity-foss-ugcPost-7398980396551454720-380utm_source=share&utm_medium=member_desktop&rclm=ACoAAFF-qDkB2yUyc88q--R_WItQvqjMpfkEG1)

**Summary:** Discussed the Free and Open Source Software culture at KL University and encouraged peers to participate in open source.

**Highlights:**

- FOSS courses and initiatives at university
- Personal journey in open source
- Community-driven development model
- Real-world learning experience

## 8.3 Post 3: Blog link to my open source journey

**Link:** [https://www.linkedin.com/posts/krishna-medapati\\_ugcPost-7398980396551454720-380utm\\_source=share&utm\\_medium=member\\_desktop&rclm=ACoAAFF-qDkB2yUyc88q--R\\_WItQvqjMpfkEG1](https://www.linkedin.com/posts/krishna-medapati_ugcPost-7398980396551454720-380utm_source=share&utm_medium=member_desktop&rclm=ACoAAFF-qDkB2yUyc88q--R_WItQvqjMpfkEG1)

## 9 Conclusion

This report documents my comprehensive journey in open source software development, including:

- Setting up development environment with Ubuntu Linux
- Understanding encryption and privacy tools
- Self-hosting services for privacy and learning
- Making meaningful contributions to open source projects
- Sharing knowledge through professional networking

Through these experiences, I have gained:

1. Technical skills in multiple programming languages
2. Understanding of collaborative development
3. Experience with version control and code review
4. Appreciation for open source philosophy
5. Professional networking abilities

Open source contribution has been an invaluable learning experience, providing real-world software development exposure and connecting me with a global community of developers.