

# **REMOTE SERVER: Rudimentary Media Server**

A real-time research / societal related project submitted in partial  
fulfillment of the  
requirements for the award of the degree of

Bachelor of Technology

In

Computer Science and Engineering

By

**VENKAT MARAMREDDYPALLI (22011M2105)**

**SRIYA CHITTANANI (22011M2111)**

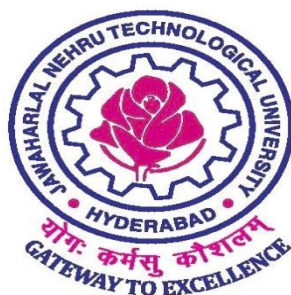
**DEEVI VENKATA NAGA SAI YASWANTH (22011M2113)**

**JASHITHA DESU (22011M2205)**

**Under The Guidance of**

**Dr. P.SAMMULAL**

**Professor**



**Department of Computer Science and Engineering,  
JNTUH University College of Engineering, Science & Technology  
Hyderabad  
Kukatpally, Hyderabad - 500 085.**

**July 2024**

**Department of Computer Science and Engineering,  
JNTUH University College of Engineering, Science & Technology  
Kukatpally, Hyderabad - 500 085.**

**July 2024**



**BONAFIDE CERTIFICATE**

Certified that this real-time/societal project report , “REMOTE SERVER”

Is a bonafide work of

**Venkat Maramreddypalli- Roll No: 22011M2105**

**Sriya Chittanani- Roll No: 22011M2111**

**Deevi Venkata Naga Sai Yaswanth - Roll No: 22011M2113**

**Jashitha Desu - Roll No: 22011M2205**

Who carried out the project work under my supervision.

SIGNATURE OF SUPERVISOR

**DR .P. SAMMULAL**

Professor

Computer Science and Engineering

SIGNATURE OF HOD

**DR.K.P.SUPREETHI**

Head of the department

Computer Science and Engineering

# ABSTRACT

This project focuses on the implementation and configuration of a remote server . The primary objective is to create a reliable and versatile server environment that facilitates efficient media management and secure file access

The problem addressed by this project is the need for a centralized system that can handle media streaming, while ensuring security and accessibility. Traditional standalone systems often lack the flexibility and integration capabilities required for modern, interconnected environments.

The method involves setting up an Ubuntu Server OS as the base system, chosen for its robustness, security features, and extensive support for open-source applications. Samba is configured to enable file sharing across different operating systems, allowing users to access and manage files remotely. Plex Media Server is installed to organize and stream media content to various devices, providing a comprehensive media management solution. OpenSSH is utilized to facilitate secure remote access and administration of the server, ensuring that all connections are encrypted and authenticated.

The results of this project demonstrate a functional remote server capable of efficiently handling file sharing, media streaming, and secure remote access. The integration of Samba, Plex, and OpenSSH on an Ubuntu Server OS provides a versatile platform that meets the needs of various users.

# ACKNOWLEDGEMENT

We wish to extend our sincere gratitude to our supervisor, **Dr.Sammulal.P**, Professor, Department of Computer Science and Engineering, for being a driving force all through the way. The process of making the project wouldn't be so smooth and interesting without his support and encouragement.

We wish to express our gratitude to **Dr.K.P.Supreethi**, Professor and Head of the Department of Computer Science and Engineering for providing necessary computing facilities. We are indebted to the Department of Computer Science and Engineering, JNTUH for providing us with all the required facilities to carry our work in a congenial environment. We also extend our gratitude to the CSE Department staff for providing necessary support from time to time whenever requested.

Above all, we are grateful to our parents and friends for their patience and continuous supply of inspirations and suggestions for our ever-growing performance.

Venkat Maramreddypalli Roll No: 22011M2105

Sriya Chittanani Roll No: 22011M2111

Deevi Venkata Naga Sai Yaswanth Roll No: 22011M2113

Jashitha Desu Roll No: 22011M2205

## **Table of Contents**

1.Introduction.....	6
1.1ProblemDefinition.....	7
1.2 Motivation.....	7
2. Proposed method and implementation.....	8
2.1 Requirement and tools.....	8
2.2Proposed design and troubleshooting.....	8
2.3Implementation.....	9
2.4 System architecture.....	11
3. Results and screenshots.....	12
4. Conclusion.....	16
5. Futurescope.....	16

# 1. INTRODUCTION

In today's interconnected world, individuals increasingly need privacy of their data due to the dawn of AI and also storage solutions with data protection, prompting the need for a dedicated space to store important files etc. Therefore, a remote physical server is a solution. A personal remote server, offers individuals dedicated hardware resources for managing their data, applications, and services. This project aims to develop a user-friendly and secure solution for managing personal data and enabling privacy

This server will provide a storage solution for minimal cost with almost no privacy concerns unlike cloud solutions like google one which costs an individual two dollars for storage with limited privacy and possible AI testing done by google. The dedicated nature of these servers means that the users are not sharing resources with other users, reducing the risk of data breaches and cyber-attacks. Additionally, users can implement their security protocols, firewalls, and intrusion detection systems, further enhancing the server's security posture.

One of the primary advantages of this physical server is their superior performance. The resources of the server—such as CPU, RAM, and storage—are dedicated to a single client, there are no competing workloads to slow down operations. This ensures consistent and predictable performance. Moreover, physical servers come with robust hardware configurations that can handle peak loads and provide the reliability that applications demand.

This server can also be used to share files among friends, store media and stream it. The server provides a quality of life improvement to stream downloaded media among a local network of computers. This eliminates the need of having to share the files with a physical storage device. This also reduces the duplication of already existing files since the server can be accessed by multiple users at once. The use-cases are endless, including hosting websites etc. This Server with Ubuntu server operating system has samba to access and share files among both windows and linux users. It also has plex integration which helps to stream media like music and movies.

## 1.1 PROBLEM DEFINITION:

To address issues like data privacy and breaching , a personal remote server is the solution. It provides a reliable and dedicated personal storage for all important files and media with backup

.

It can be accessed easily under the same network for file sharing and media streaming without third party operators and paying excessive costs .

This solution will enable user's flexibility and data protection and preservation with backup capabilities

## 1.2 MOTIVATION:

The motivation to build a remote server stems from the fact that in this increasingly inter-connected world every document and file is shared and stored only in software form with risks of data breaching and data being lost.

Therefore , the server is built to securely store sensitive files within a controlled environment.

Also due to the dawn of AI, our data is being used without permission by large corporations. This server will give data privacy since it can only be accessed by a specific set of users and devices.

It also gives greater flexibility and efficiency by being accessible by multiple devices to stream media unlike other platforms.

## 2. PROPOSED METHODS AND IMPLEMENTATION

To address the problems of data breaching, storage and media preservation, a remote physical server is proposed. A physical server provides more reliability and is more robust as opposed to virtual machines. It offers maximum security to the data and has excellent backup solution leading to preservation of important documents and media. It also offers file sharing among a single network with multiple devices.

## 2.1 REQUIREMENTS AND TOOLS:

### Hardware requirements:

- Should have intel or AMD processor
- Only supports 64 bit processors and 32 bit arm
- Minimum of 3 GB RAM to ensure smooth operation and memory upgrades
- An ethernet port
- Minimum storage should be 25 GB or more
- Is compatible with peripheral devices
- Adequate fans to prevent overheating

### Software tools:

- Operating system: Ubuntu server operating system
- Samba for file sharing.
- Plex for media sharing.

## 2.2 PROPOSED DESIGN AND TROUBLESHOOTING

The proposed method is to use ssh (secure shell), for remote access which offers a secure and efficient way to manage the server from any device in the network. We use OpenSSH based on secure shell protocol which provides access and safety over an unsecured network. It provides a large suite of secure tunneling and only provides access to specific users.

The proposed way was to use debian with light weight XFCE environment. The problem with this approach is that even though a light weight desktop environment, it is not suitable for long term use. The server may shutdown or crash after 2 days of continual usage with the GUI.



Hence, Ubuntu server edition is used because it doesn't have any GUI and takes less processing power. This improves the performance of the server and the temperature too.

Proposed method for file sharing is Samba which implements the SMB/CIFS networking protocol, it becomes a vital utility for cross-platform interaction. It's renowned to provide full networking capabilities to windows and is also extremely compatible with various linux and unix.

For media sharing, Plex is a media server software that allows user to stream their own movies. It allows users to create a client-server for movies, shows and music.

For backup, we can use the option of safe boot when booting up the server. Safe boot option lets you access the data without properly saving the new data. This option is mostly used to copy the pre-existing data on the server to a portable drive before nuking the server for re-implementation.

## 2.3 IMPLEMENTATION:

We take a PC with an OS already installed on it. We then take a portable storage drive and plug into the PC with OS. Now, the .iso file for Ubuntu LTS 24.02 Server Edition is to be downloaded from the official website. This OS is booted on to the portable storage drive with the help of applications like BalenaEtcher or Rufus. Now that the OS is booted into the portable storage media, The actual server or the PC which will act as server for the project is taken and it is connected to the portable storage media. While booting the system, go to the BIOS settings and boot it using the storage media. This step varies vastly from PC to PC since the BIOS settings are not the same from manufacturer to manufacturer. During installation there are some important things to consider:

i) It is preferable to use a LAN connection rather than wireless options because the port can be easily recognized and we need the data about port for maintenance.

ii) While partitioning SSD or HDD it is important to uncheck the option "set up this disk as an LVM group". You can think of LVM as "dynamic partitions", meaning that you can create/resize/delete LVM "partitions" from the command line while your Linux system is running: no need to reboot the system to make the kernel aware of the newly-created or resized partitions. But the initial setup of LVM is more complex than just partitioning a disk, and you

will definitely need to understand the LVM terminology and model (Logical Volumes, Physical Volumes, Volume Groups) before you can start using it. Also, you lose all the data if one of the drives fail.

iii) Also give username and password you remember. Without them you can't access the server.

iv) Not necessary but pre installing open SSH makes the process convenient.

Now that the server OS is setup, we need to access the server from other device to make things easier. Run the command to find the ipv4 address. Note down the ipv4 address and open the terminal of a system in the same network. Connect to server using SSH command to connect. After the remote access setup, we need to install some important applications to run this server as intended. Before downloading anything, we need to make directory which only the user's password so that the media in there is protected.

First application we need to download is samba db. Install it by downloading its .deb file and unpacking it by using dpkg command. Now, go to the smb.config file and make some change with vim or nano since the terminal has no GUI based text editors. Also put smb password. Install Plex media server on the server in similar fashion and enable Plex to run as soon as the system starts.

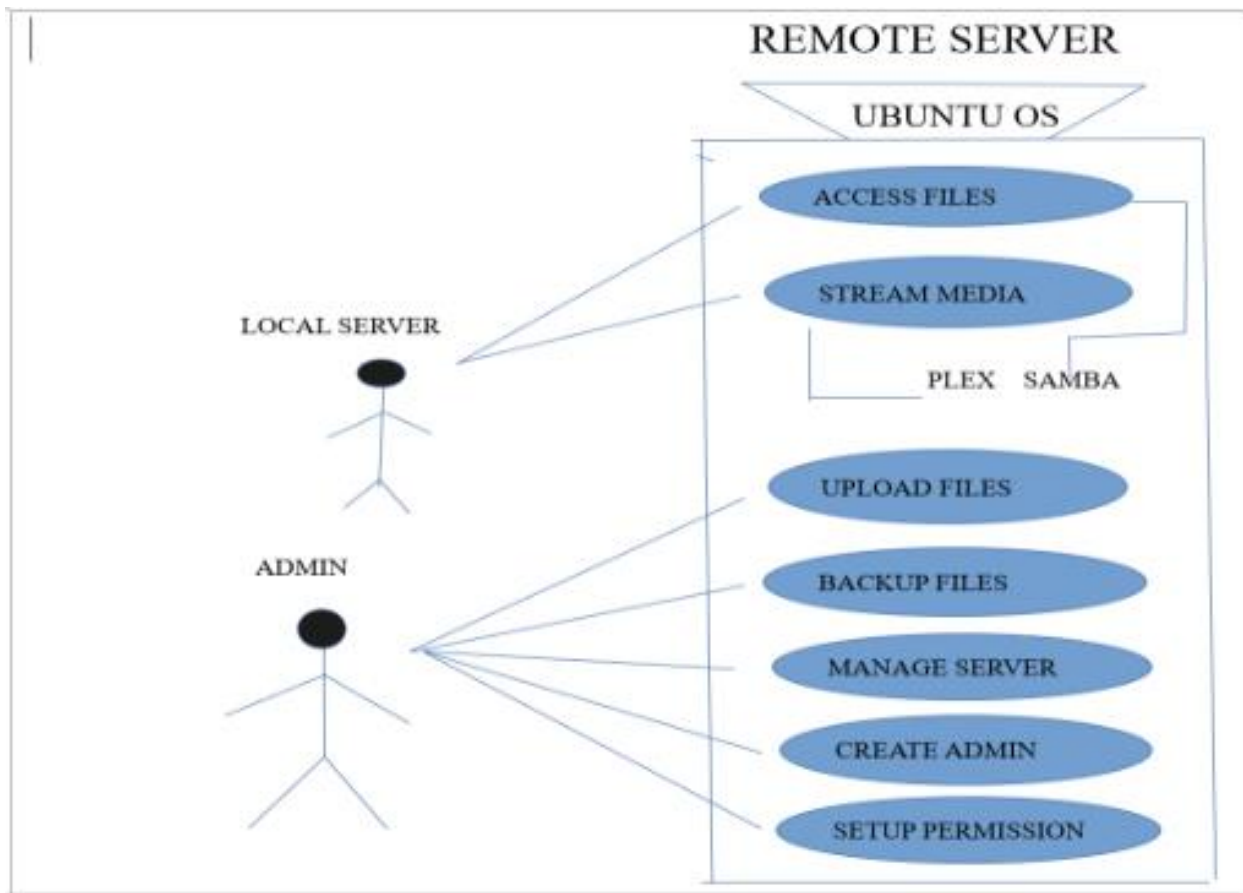
Extra:

we can map a network drive to the server for easier transfer of files.

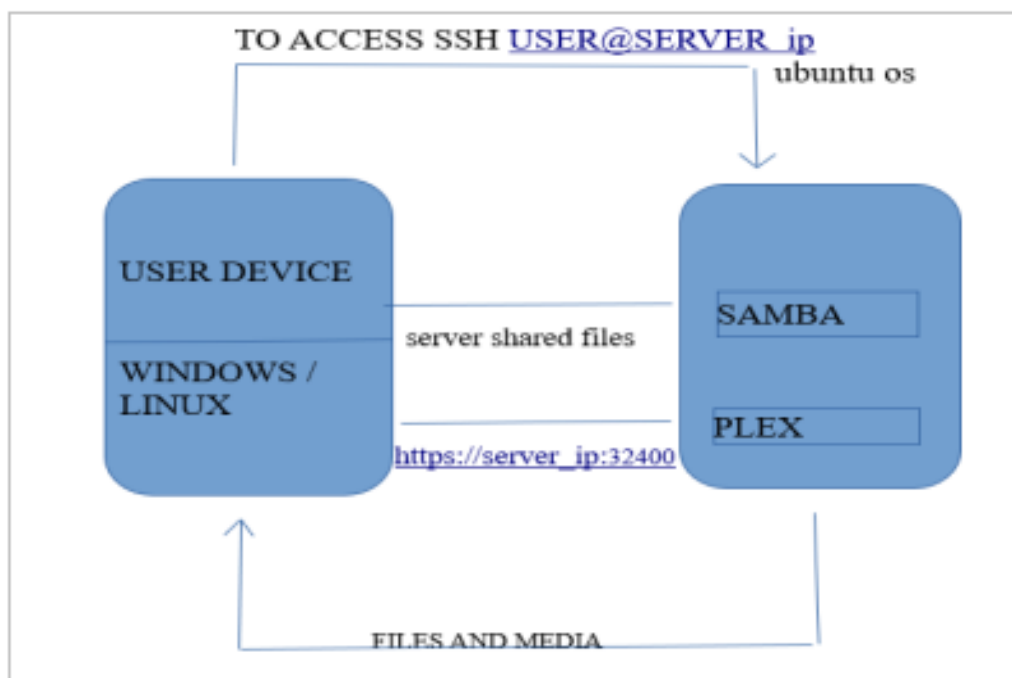
Now we can go to the plex media website by using ipv4:32400 to access the contents of the servers. The Plex service also reads meta data of the files and make it prettier to look. Here we can set up the folder in server from which the data is taken from.

Accessing for regular use: Since server is switched on for a long time, the transfer of files can be done using either network drive or portable storage media directly connected to server. Once the files are put in server, the files can be accessed by all computers in the same network.

## 2.4 SYSTEM ARCHITECTURE:

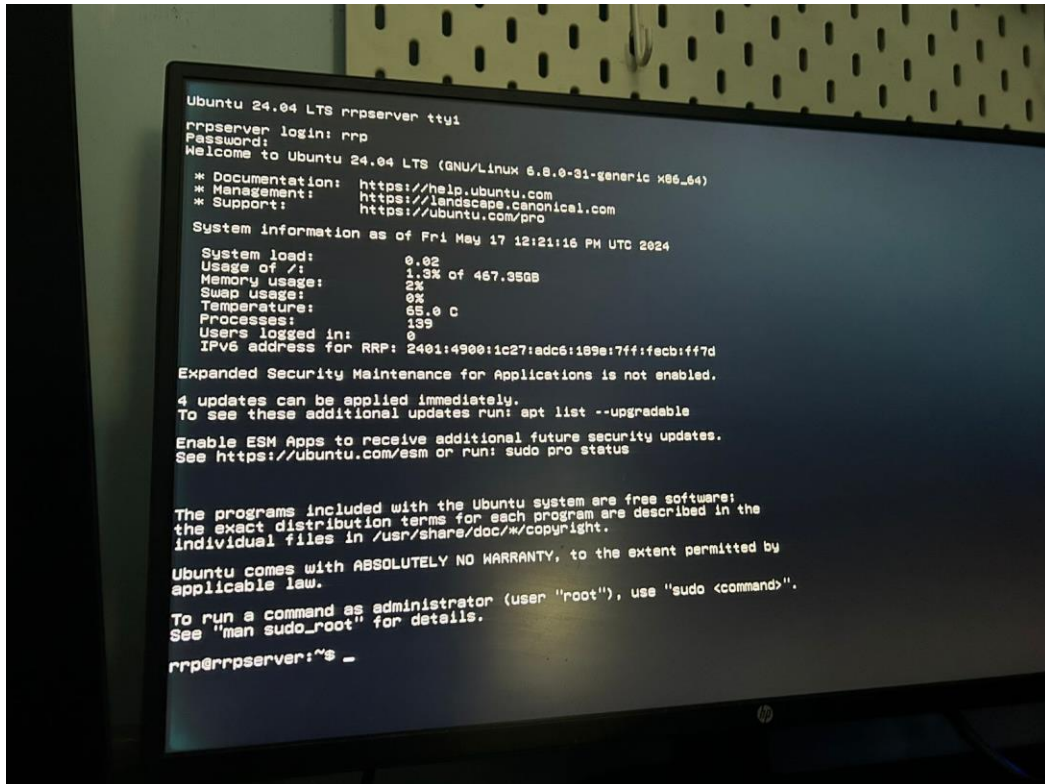


USE CASE UML DIAGRAM

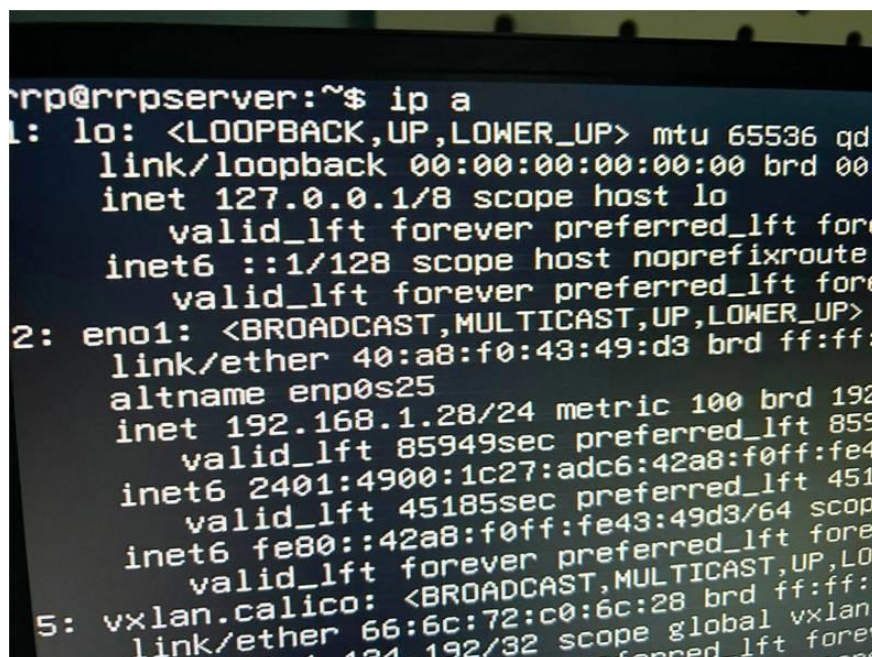


## DEPLOYMENT UML DIAGRAM

## 3.RESULTS AND SCREENSHOTS:



## SERVER ACCESS



```
Terminal - rrp@rrpserver: ~
File Edit View Terminal Tabs Help
dvns@192:~$ ssh rrp@192.168.1.25
The authenticity of host '192.168.1.25 (192.168.1.25)' can't be established.
ED25519 key fingerprint is SHA256:8sX+sUcKcbE2aWNTq304hI7Vtc4yC+hAExD2SyHkyW0.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.25' (ED25519) to the list of known hosts.
rrp@192.168.1.25's password:
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-35-generic x86_64)

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

System information as of Thu Jun 27 03:40:04 PM UTC 2024

System load:          0.44
Usage of /:            3.5% of 467.35GB
Memory usage:         10%
Swap usage:           0%
Temperature:          42.0 C
Processes:            201
Users logged in:      0
IPv4 address for eno1: 192.168.1.25
IPv6 address for eno1: 2401:4900:1c26:5b62:42a8:f0ff:fe43:49d3

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Sat Jun  8 14:34:25 2024 from 192.168.1.26
rrp@rrpserver:~$
rrp@rrpserver:~$
```

## SERVER

```
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.1) ...
Scanning processes...
Scanning processor microcode...
Scanning linux images...

Running kernel seems to be up-to-date.

The processor microcode seems to be up-to-date.

No services need to be restarted.

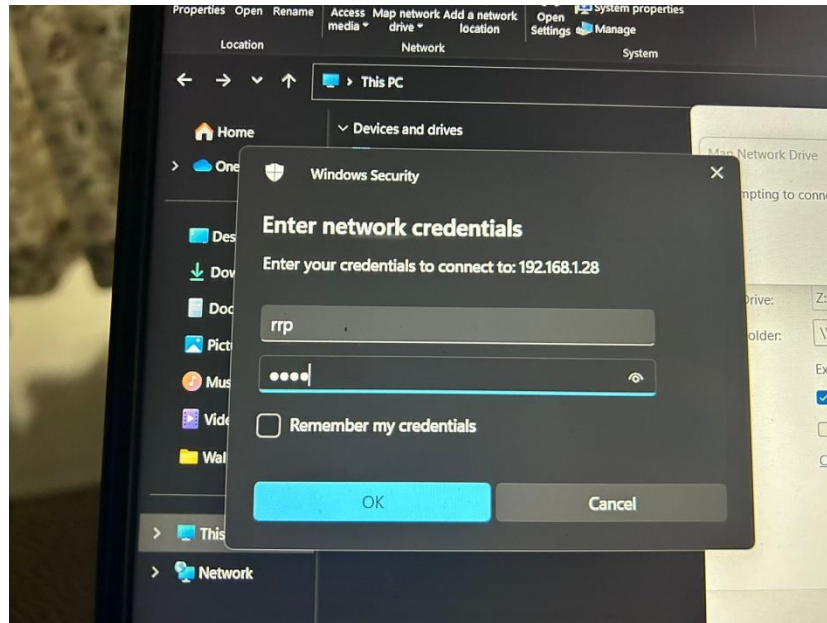
No containers need to be restarted.

No user sessions are running outdated binaries.

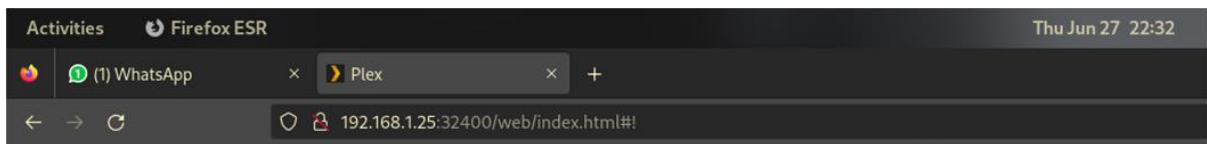
No VM guests are running outdated hypervisor (qemu) binaries...
rrp@rrpserver:~/testing$ sudo mkdir /media/myfiles
rrp@rrpserver:~/testing$ sudo chown $USER: /media/myfiles
rrp@rrpserver:~/testing$ sudo nano /etc/samba/smb.conf
```

## SAMBA

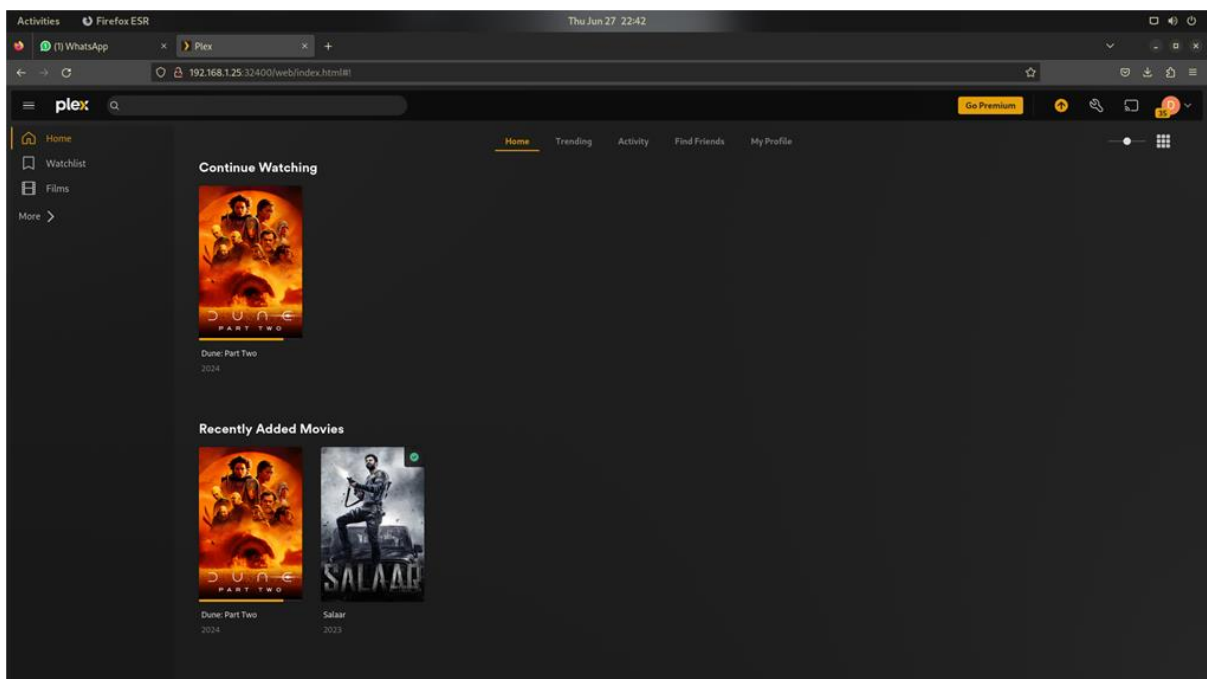




CONNECT TO SERVER



PLEX ACCESS



## 4. CONCLUSION:

In conclusion, remote server is designed for streamlining file sharing and streaming media under the same network, that can be accessed by multiple users. By leveraging this software, we can ensure protection of important data and preservation of it in the future. This project demonstrates a functional remote server capable of efficiently handling file sharing, media streaming, and secure remote access. The integration of Samba, Plex, and OpenSSH on an Ubuntu Server OS provides a versatile platform that meets the needs of various users.

## 5. FUTURE SCOPE

For further scaling , we can cluster a few servers together to enable increased demand and maintain peak performance.

For much wider use cases , we can make the server accessible to different networks, which will achieve greater flexibility and efficiency . It will act as a cloud storage without any third party service providers greatly saving costs.