SANJANAA SRIDHAR- PROJECT: REMOTE ACCESS USING RASPBERRYPI NAS

TABLE OF CONTENTS

**CHAPTERS CONTENTS PAGE NO.**

1. ABSTRACT

2. INTRODUCTION

3. REQUIREMENT ANALYSIS

4. ARCHITECTURE & DESIGN

5. IMPLEMENTATION

6. EXPERIMENT RESULTS & ANALYSIS

6.1. RESULTS

6.2. RESULT ANALYSIS

6.3. CONCLUSION & FUTURE WORK

7. REFERENCES

### 

### 1. ABSTRACT:

### The developing interest on the lookout for expanded capacity limit is basically a direct result of our reliance on Internet. Network Attached Storage (NAS) gives a committed record server to deal with a wide range of documents. It is an autonomous Storage gadget which is associated straightforwardly to the organization. Because of its accessibility on the organization it very well may be handily gotten to by quite a few heterogeneous clients. The Network Attached Storage gadgets promptly accessible in the market these days are profoundly overrated and don't give a lot of degree to upgrades and enhancements. Another perspective which is to be thought about is that in many cases these gadgets consume a considerable amount of power. This approach discussed in this paper targets giving a minimal expense NAS framework which is not difficult to utilize and design. It additionally involves added security highlights and Web Server capacities. It empowers you to have nonstop accessible stockpiling gadget which is helpful as well as power saving and permits openness to information on and off the organization.

### 2. INTRODUCTION:

The components/ softwares / protocols our NAS has are as follows:

* Raspberry pi
* Hard disk
* Ethernet cable
* HDMI cable
* Any desk software
* Open media vault
* SMB /samba protocol

#### Raspberry pi:

Raspberry Pi is a series of small single-board computers (SBCs) developed in the United Kingdom by the Raspberry Pi Foundation in association with Broadcom. The Raspberry Pi project originally leaned towards the promotion of teaching basic computer science in schools and in developing countries. The original model became more popular than anticipated, selling outside its target market for uses such as robotics. It is widely used in areas, such as weather monitoring, because of its low cost, modularity, and open design. It is typically used by computer and electronic hobbyists, due to its adoption of HDMI and USB devices.

****

#### Any desk software:

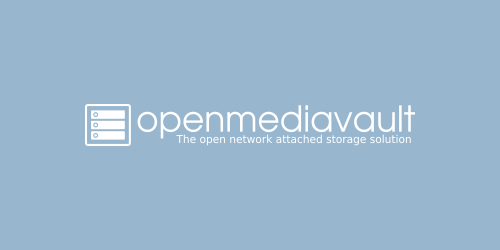
AnyDesk is a closed source remote desktop application distributed by AnyDesk Software GmbH. The proprietary software program provides platform independent remote access to personal computers and other devices running the host application. It offers remote control, file transfer, and VPN functionality.



#### Open media vault:

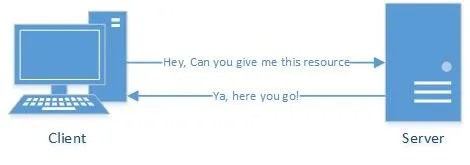
OpenMediaVault is the next generation network attached storage (NAS) solution based on Debian Linux. It contains services like SSH, (S)FTP, SMB/CIFS, DAAP media server, RSync, BitTorrent client and many more. Thanks to the modular design of the framework it can be enhanced via plugins.

OpenMediaVault is primarily designed to be used in small offices or home offices, but is not limited to those scenarios. It is a simple and easy to use out-of-the-box solution that will allow everyone to install and administrate a Network Attached Storage without deeper knowledge.

****

#### SMB protocol:

In computer networking, Server Message Block (SMB), one version of which was also known as Common Internet File System (CIFS), is a communication protocol for providing shared access to files, printers, and serial ports between nodes on a network. It also provides an authenticated inter-process communication (IPC) mechanism. Most usage of SMB involves computers running Microsoft Windows, where it was known as "Microsoft Windows Network" before the introduction of Active Directory. Corresponding Windows services are LAN Manager Server for the server component, and LAN Manager Workstation for the client component.

****

### 3. REQUIREMENT ANALYSIS:

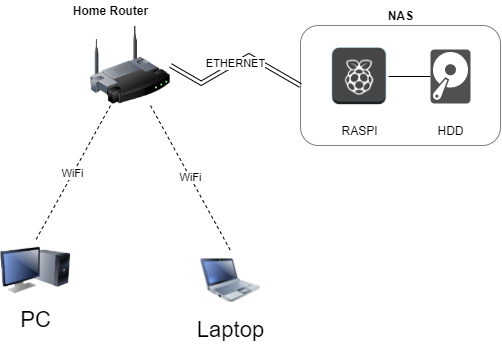
* **A Raspberry Pi with all the trimmings:** A Raspberry Pi for this project is the most important requirement, along with the requisite accessories:
* A 9V power supply
* A microSD card.
* SD card reader
* HDMI to mini-HDMI cable
* And a mouse, a keyboard, and a monitor for the initial setup.

* Any of the recent-model PIs should work for this project. We used RaspberryPi 4 Model B for this study.
* **A disc drive (or two, or three):** Except if it's sharing only a couple of records, the microSD card isn't sufficient capacity for a NAS. We will require a hard drive to upload music, movies, or different records we need to share between gadgets. A standard external drive will get the job done by and large, however for huge corporate networks we might require one that plugs into the wall independently or a controlled USB hub, since the PI will be unable to supply sufficient capacity to each of our drives. For a cleaner arrangement, we can utilize an inner drive intended for network connected capacity, as well, yet that would require a case. In this study, we have utilized a standard external hard drive attached directly to the Raspberry Pi via USB cable.

* **A NAS-friendly case (optional)**: If we want our system to have a clean look, it may behove to get an enclosure for our PI and drives, so it isn't just an octopus of wires and disks. For example, Geekworm makes a board called the [X825](https://zdcs.link/RxNdV) that allows you to dock a 2.5-inch internal hard drive, connect it to your PI, and mount it all in a [trim little case](https://zdcs.link/0N0Ve). (Just make sure you get one that's compatible with your PI—there are different cases for the PI 3, the PI 4, and so on.)
* For now, we are using a standard Raspberry Pi case with a USB external drive attached to it. Once you have all your components in hand, it's time to get your NAS up and running.

### 4. ARCHITECTURE & DESIGN:

The high level view of the architecture is as follows:



### As seen from above, the RaspberryPi and HDD together constitute the NAS. The NAS is connected to the router via ethernet cable and is hosted locally on the network. We can then access the NAS via any windows/linux device on the same network by first specifying a network location(i.e. IP address of the raspi) and then entering the correct credentials.

### 5. IMPLEMENTATION:

The NAS was implemented in a series of steps as follows:

1. **Installation of OpenMediaVault:**

OpenMediaVault is the NAS solution software that we used. OpenMediaVault(OMV) is what allows you to create a NAS, make shared folders, set permissions, enable sharing, etc. So we first installed OMV on our raspberry pi.

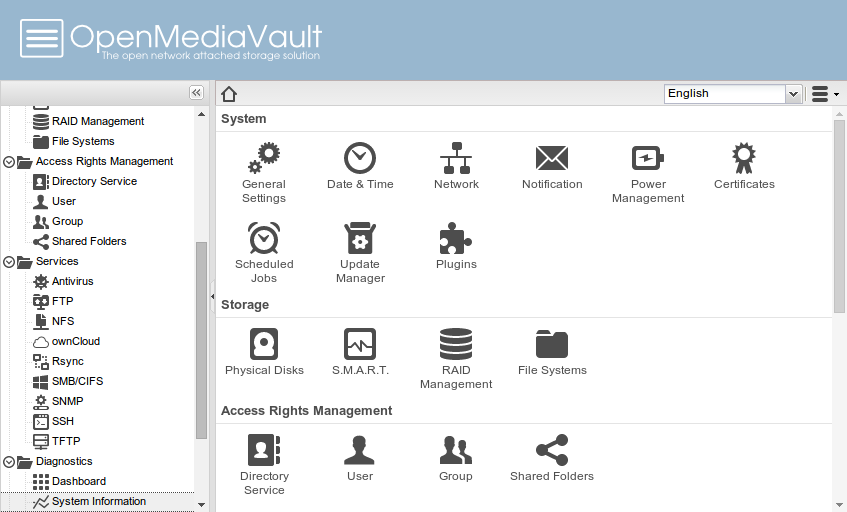
Command:

*wget -O - https://raw.githubusercontent.com/OpenMediaVault-Plugin-Developers/installScript/master/install | sudo bash*

Once the installation is complete, reboot the pi

*sudo reboot*

To access OpenMediaVault, open your browser and go to *http://[RaspberryPiIPAddress]*



1. **Mounting hdd and making a shared folder:**

The next step is to mount your hdd that’s connected to the raspberry pi and make a shared folder.

Check the disk list with the command:

*sudo fdisk -l*

Locate the disk and mount it using the command:

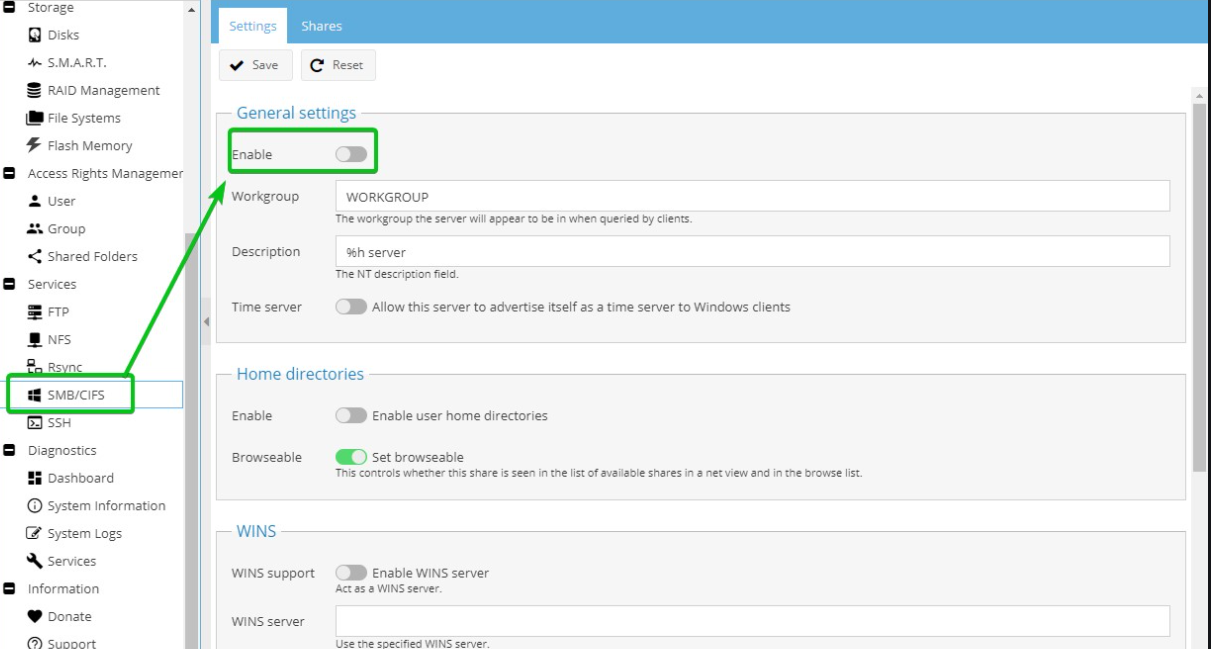
*sudo mount -t ntfs /dev/sdb1 /media*

1. **Making shared folder:**

We can either expose the entire hard drive to the network making it accessible to all, or just certain folders that we would like to. Here, we do the latter. Just make a new folder in the HDD that you wish to expose.

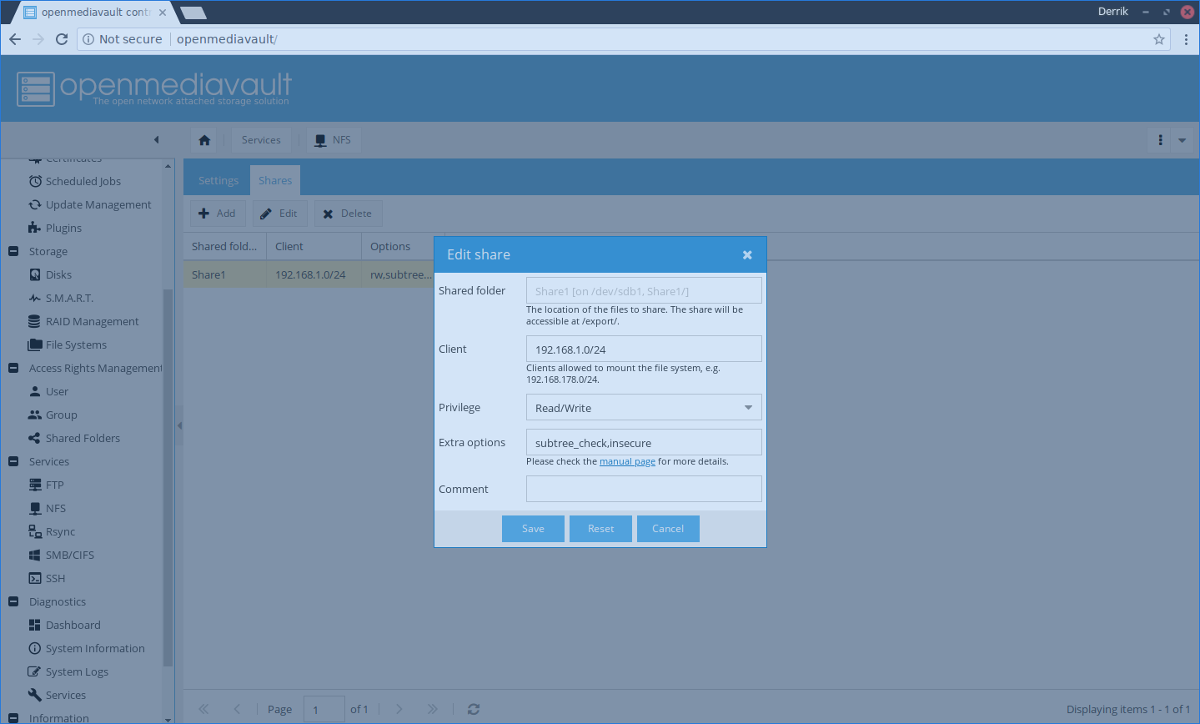
1. **Enabling SMB sharing:**

After making the shared folder, we need to enable SMB sharing so that windows devices can access this folder via SMB protocol. We simply go to “Services” in OMV, enable SMB, then add our shared folder to it.



1. **Enabling NFS:**

Like windows uses SMB, Linux devices use NFS protocol for file sharing. So we then need to enable NFS in OMV and add our shared folder there too.

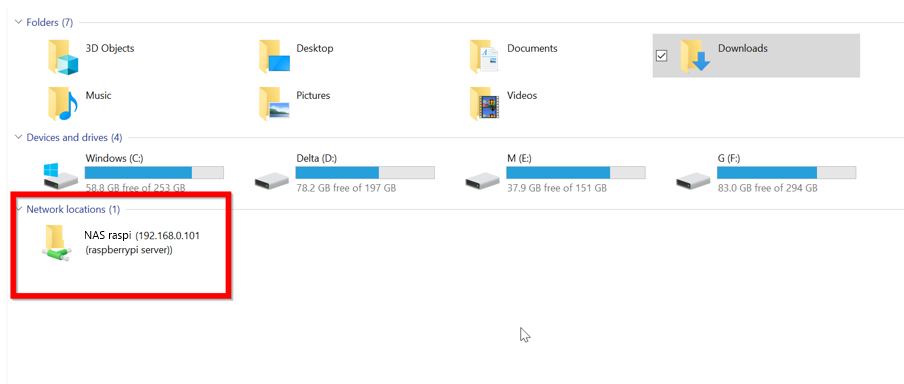


1. **Setting permissions:**

Now that we have set up our shared folder and the services we define which users are allowed to access this share. For this project, we have disabled guests and allowed only the “pi” user full read/write access to the share.

1. **Setup complete:**

Now we have everything setup. We can access our shared folder from other devices by simply adding a “Network location”(i.e. Our Raspi’s IP address) to our pc in the file explorer. After adding the network locat ion it’ll prompt you for the credentials of the raspberry pi. You can then access the NAS by entering the correct credentials.

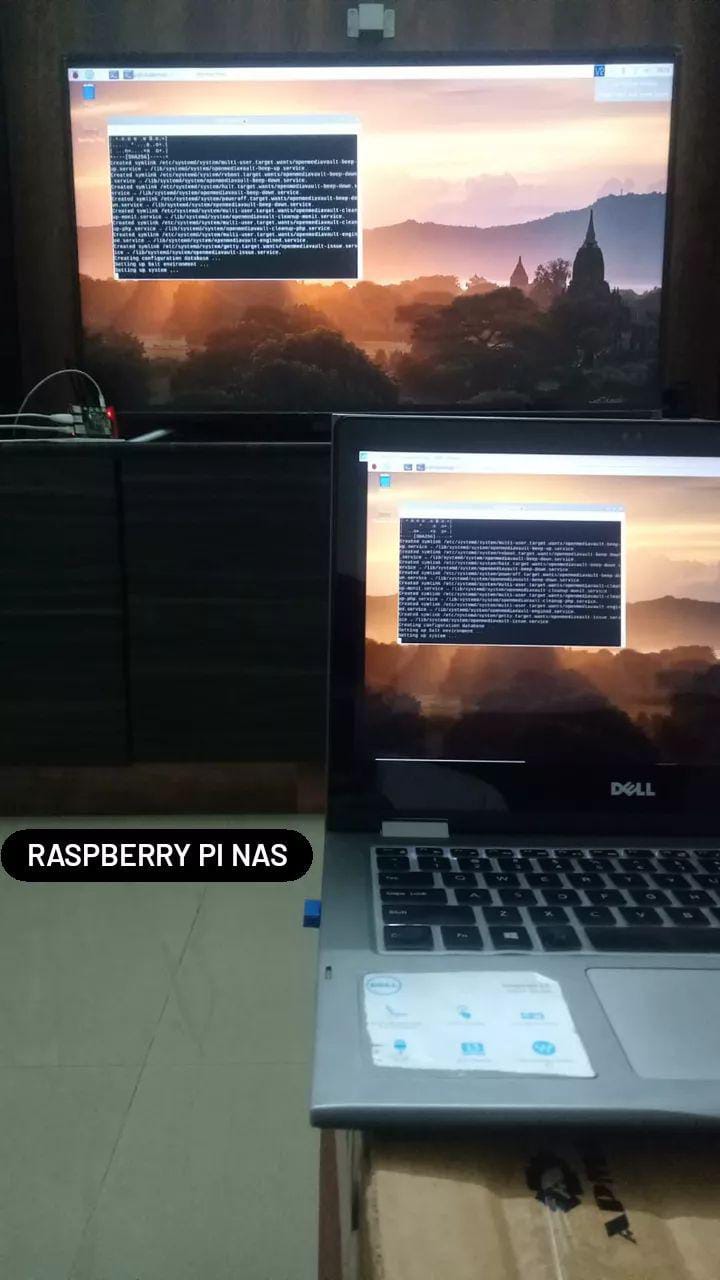
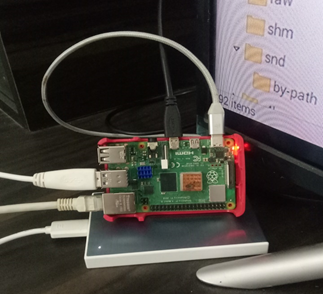


### 6. EXPERIMENT RESULTS & ANALYSIS:

#### 6.1. RESULTS:

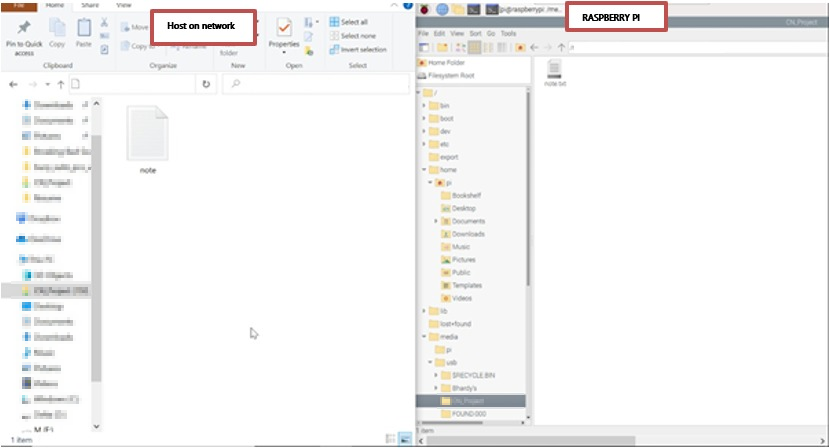
We have successfully implemented and established the Network Attached Storage using Raspberry PI.

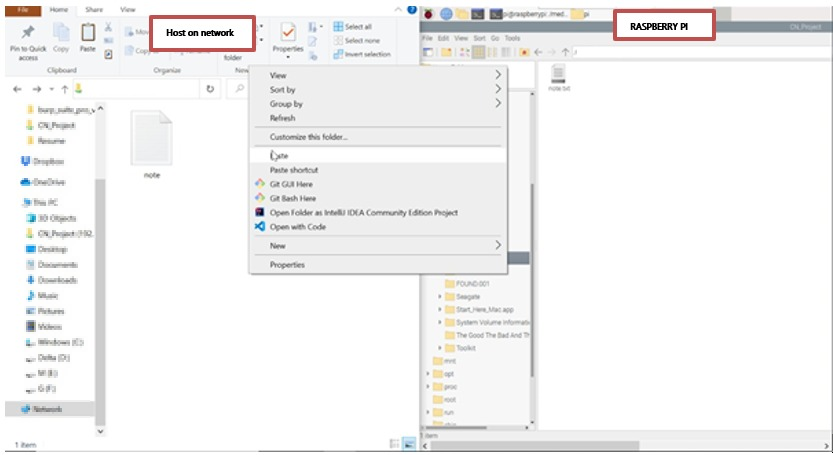
#### SETUP:

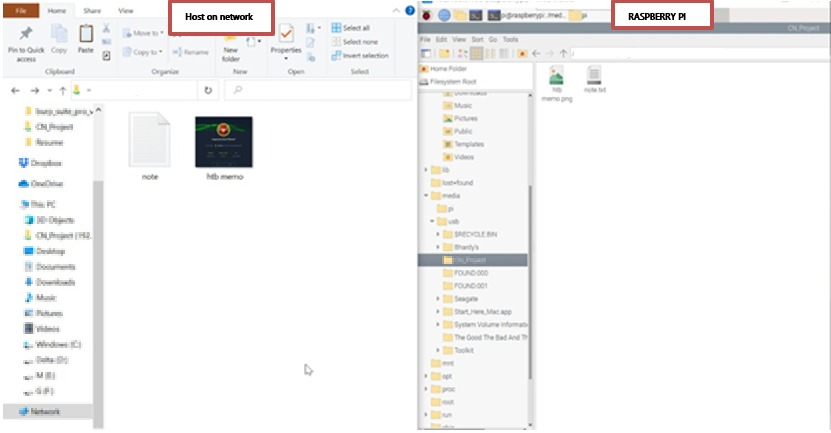
** **

#### 6.2. RESULT ANALYSIS:

#### 





****

#### 6.3. CONCLUSION AND FUTURE WORK:

High-performance, low-latency networking is essential to achieving the potential of scalable network-attached storage. Software for the critical data in a LAN was developed. The software could perform backing up of files in remote nodes, deletion of files from remote nodes, retrieve files from remote nodes, join the group and unsubscribe from the group. The main objective of this project is to get thorough knowledge of the Raspberry PI board and about implementing NAS using commands in Linux or Windows operating systems.

**Future Scope:**

* The number of businesses using voice over wireless ethernet LANs is set to triple in the next two years due to open-source IP phones.
* Concentrates on the security issues which are handled.
* Secure communication.
* **Graphical user interface design:** As its clear from the screenshots or the demonstration of the web application (here we used OpenMediaVault), we must develop a much appealing web interface with better use of colours and more interactive tables.
* **Drag and drop:** Further down the line, this feature to be implemented as drag and drop functionality in the web interface will allow files and folders to be copied and moved on the file system with ease. These operations are quite complex to implement and require a large amount of JavaScript experience which we can achieve in the future.

### 7. References:

* <https://medium.com/theteammavericks/wireless-nas-using-raspberry-pi-669d81883fdb>
* <https://www.ionos.com/digitalguide/server/configuration/raspberry-pi-nas/>
* <https://www.youtube.com/watch?v=gyMpI8csWis>