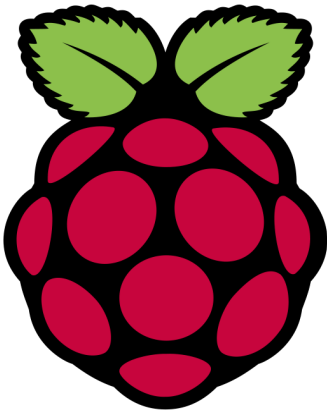


Raspberry Pi Server

Barry Lee
Chris Orcutt

CSE 321 - Realtime and Embedded Systems
SUNY University at Buffalo
bclee@buffalo.edu
crorcutt@buffalo.edu

1. Executive Summary

The Raspberry Pi server will be able to host web application and web content, such as text, images, videos, documents, and other files in on the server. The Raspberry Pi will be a cheap, portable, and easily manageable alternative to hosting your own server and purchasing processing time.

2. Project Objectives

The Raspberry Pi Server will meet the following objectives:

- Enable a locally hosted server in order to allow access to a small group of users
- Host user content such as images, videos, documents, and web applications
- Explore a little bit about how servers work and operate
- Be able to set up a server on the small portable device

The hardware required for this project includes:

- Raspberry Pi 3
- 32 GB MicroSD Card
- USB Keyboard
- USB Mouse
- HDMI Cord
- Display

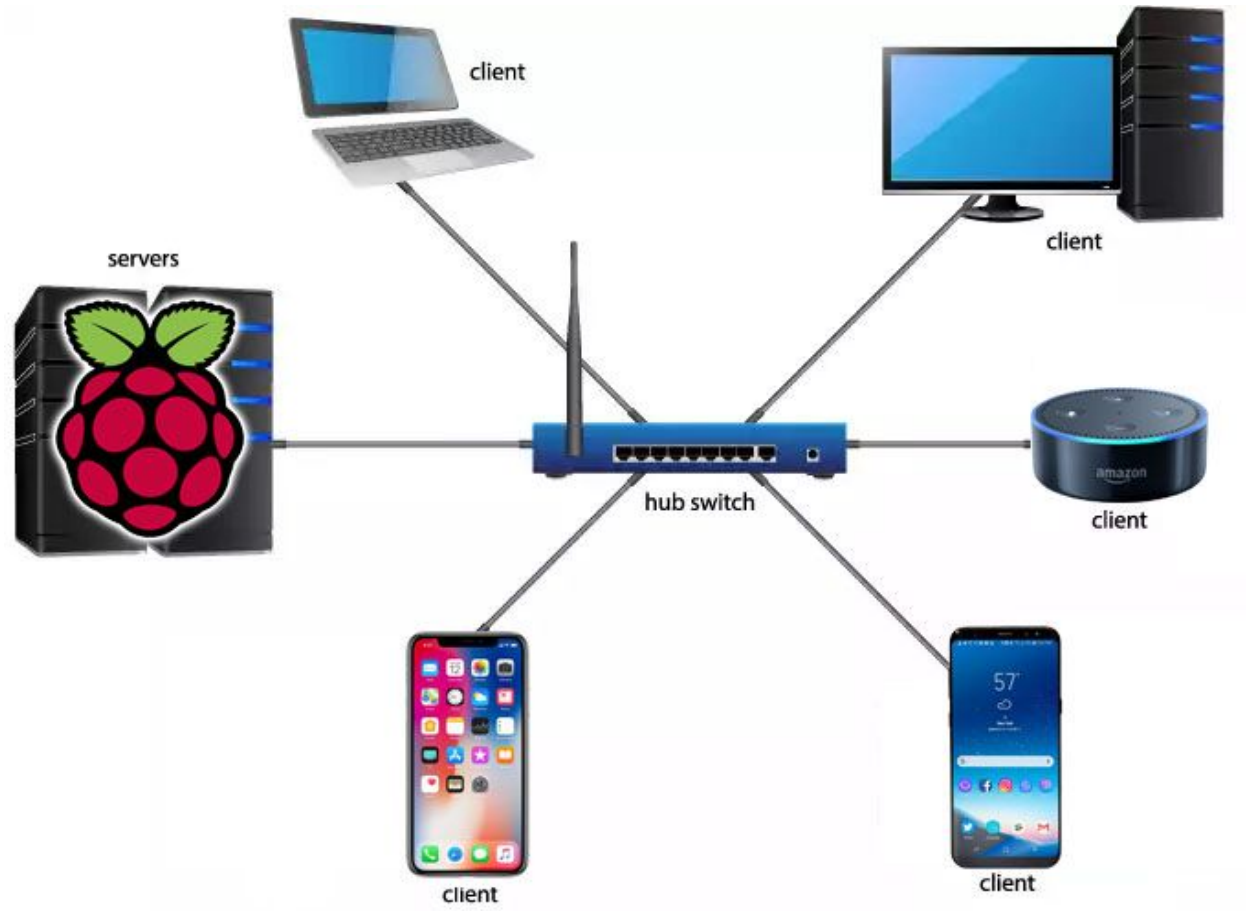
The software required for this project includes:

- Apache Web Server
- Win32 Disk Imager
- Raspbian “Wheezy”
- Text Editor
- Terminal
- Github

3. Project Approach

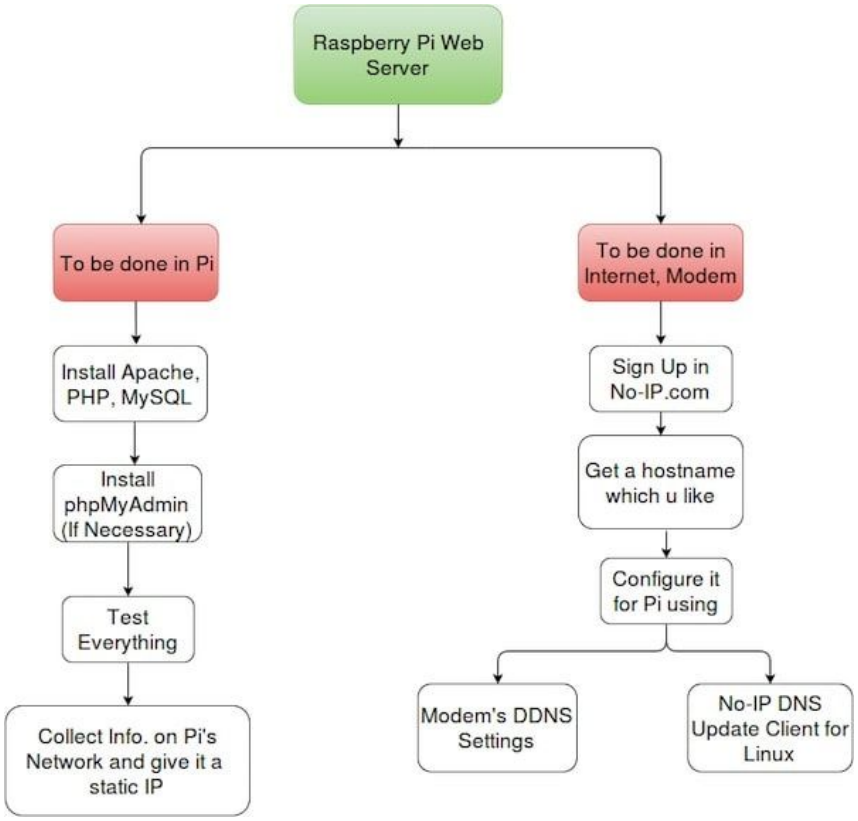
We will set up an Apache server on a Raspberry Pi which allows you to access your files from any part of the world using the internet or locally. We will be using Apache, PHP, and MySQL to build this Raspberry Pi Web Server. We will be using a few command lines to setup and install the server. This setup will be easy to replicate anywhere given that the users have all the equivalent software and hardware. Users will be able to host anything they desire on to the server and it will be able to be sharable across the network.

4. Project Description



As graduating computer science students, employers find that having websites to not only show off your skills but also have a central hub to show off some other projects is vital. Thus sparked the idea to create our own web server using a Raspberry Pi. When creating our project we ran into one problem, connecting the web server to the internet. This problem was solely due to the fact that we did not have direct access to the university's routers to create port forwarding for the Pi to be able to host our web server to the internet. Although we could not connect directly to the internet, having a local area network server has many practical applications all in itself. Whether a student wants to practice some of their skills and not make their content public, or for small businesses to use for their day to day operations.

5. Design Details



The hardware required for this project includes:

- Raspberry Pi 3
- 32 GB MicroSD Card
- USB Keyboard
- USB Mouse
- HDMI Cord
- Display

The software required for this project includes:

- Apache Web Server
- Win32 Disk Imager
- Raspbian “Wheezy”
- Text Editor
- Terminal
- Github

6. User's Manual

Users can access the server by typing the IP address of the Raspberry Pi into the address bar. The user can make this easier by editing the /etc/hosts and changing the IP address to map to a more user-friendly address. In our term project, we named it "termproject.com". It is up to the server host to make the content as user-friendly as possible. To modify, remove, or add any files, you must update /var/www/html and change index.html.

7. Programmer's Manual (For Local Host)

1. sudo apt-get update
2. sudo bash
3. apt-get install apache2 apache2-doc apache2-utils
4. apt-get install apache2 php7.0 php7.0-curl php7.0-gd php7.0-imap php7.0-json php7.0-mcrypt php7.0-mysql php7.0-opcache php7.0-xmlrpc libapache2-mod-php7.0
5. apt-get install php7.0-mysql
6. apt-get install mysql-server mysql-client
7. Go to localhost or '127.0.0.1' to check if you can connect

8. References

- https://diyhacking.com/raspberry-pi-web-server/
- http://www.instructables.com/id/Turning-your-Raspberry-Pi-into-a-personal-web-serv/
- https://www.raspberrypi.org/downloads/
- https://www.cnet.com/how-to/install-raspbian-on-a-raspberry-pi-without-noobs/
- https://www.stewright.me/2016/03/turn-raspberry-pi-3-php-7-powered-web-server/

9. Project Presentation Details

- Setup the Raspberry Pi
- Connect additional devices (with browsers) to access point/hotspot
- Demonstrate that multiple devices can connect to the server
- Explore the server's content
- Explain the possible applications for this technology:
    - Private and local group projects
    - Small businesses to maintain their database without having to pass data through the internet
    - Better security, prevents unauthorized remote accesses (local access)