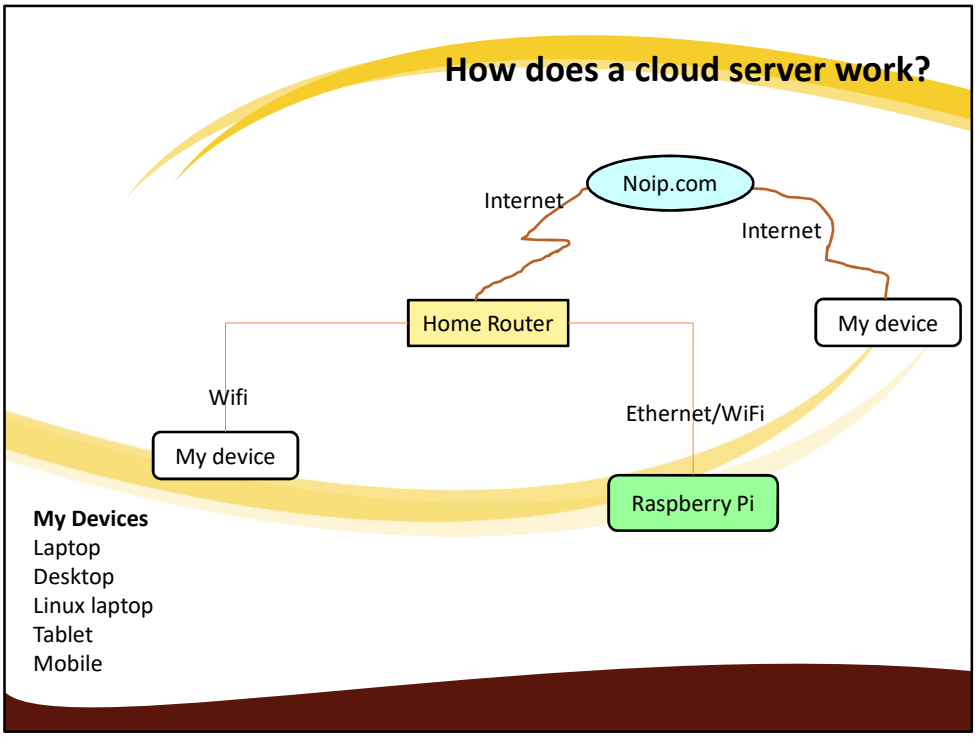
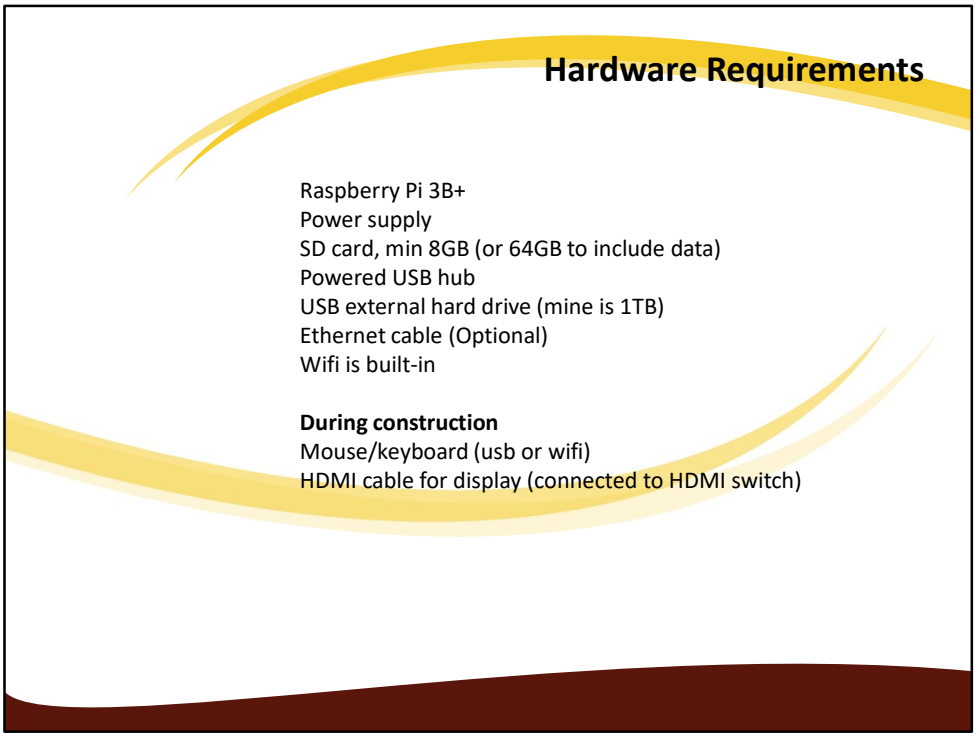
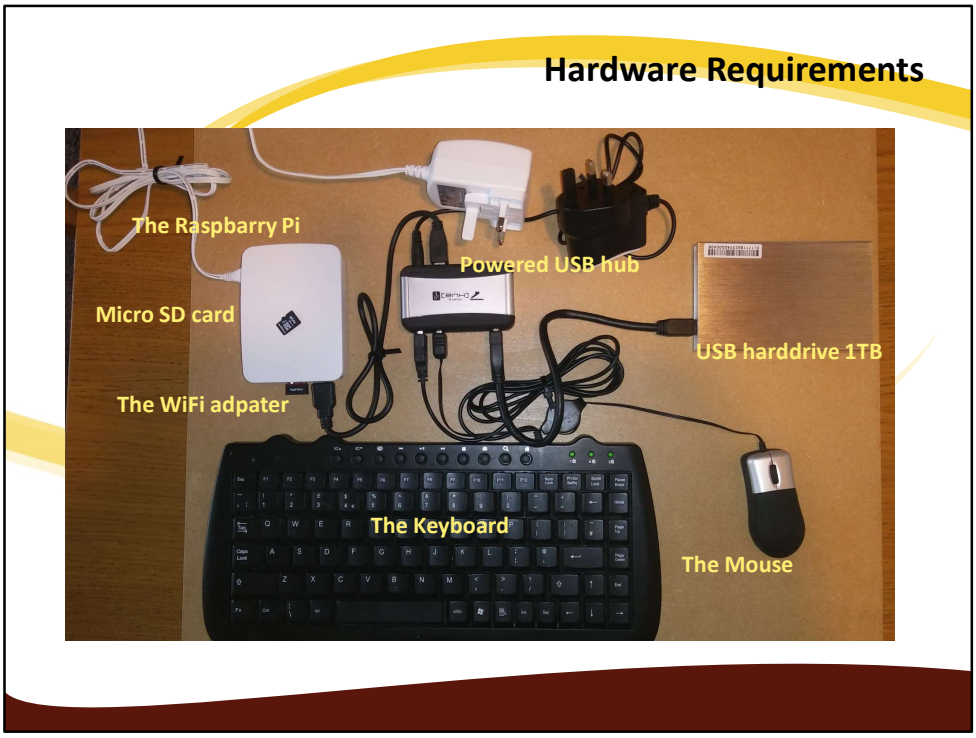
Having upgrader to an SSD for my main drive on my laptop, I have a problem because all my documents and media files are now on an external USB drive.
When outside of the my home I have no access to these files.
I am reluctant to use 'clouds' available from Microsoft, Google etc and want a solution which I own.
This would also make my files available to all my devices, wherever I am.



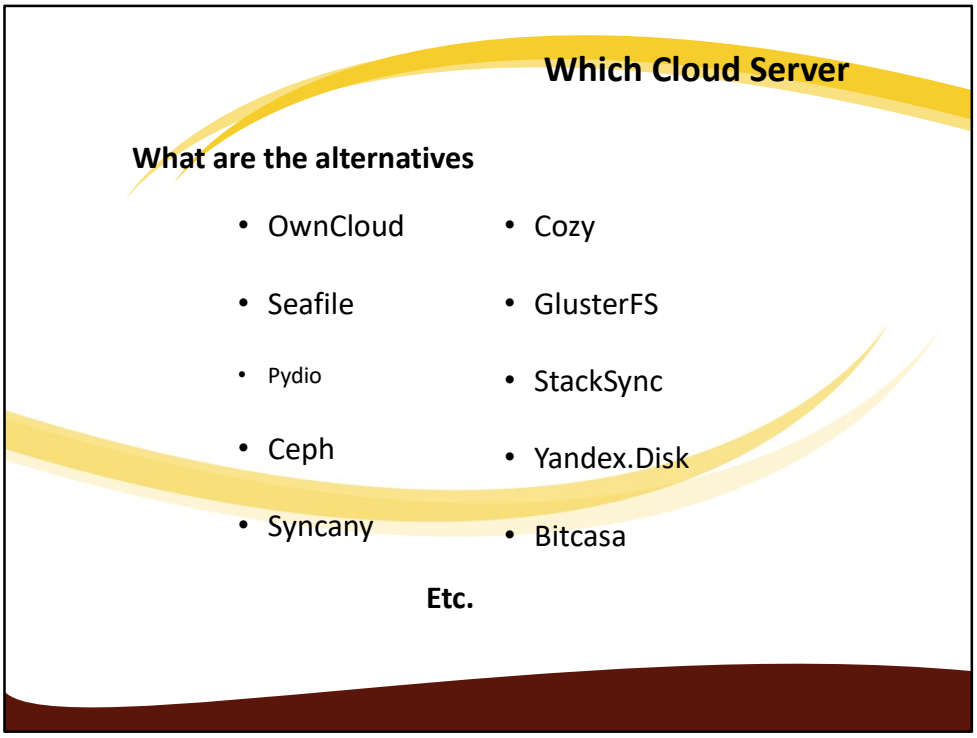
A cloud server is installed on my Raspberry Pi. This is left permanently on from when I get up until I go to bed.
Noip.com is required. It detects the external ip address currently being used by my router (dynamically provided by my internet provider).
The router is configured to allow access to the cloud server ip address.
When connected to the home network devices have direct access via the router.



The USB drive is not used in this first stage of development. At a later stage the Nextcloud data directory will be moved here together with a web server documents directory.
During development I use a wiFi connection to the router but in production this will be replaced by an ethernet cable to provide greater speed.
In production the Pi will have no connection to a mouse, keyboard or monitor. It will be operated remotely via Teamviewer or similar remote control program.

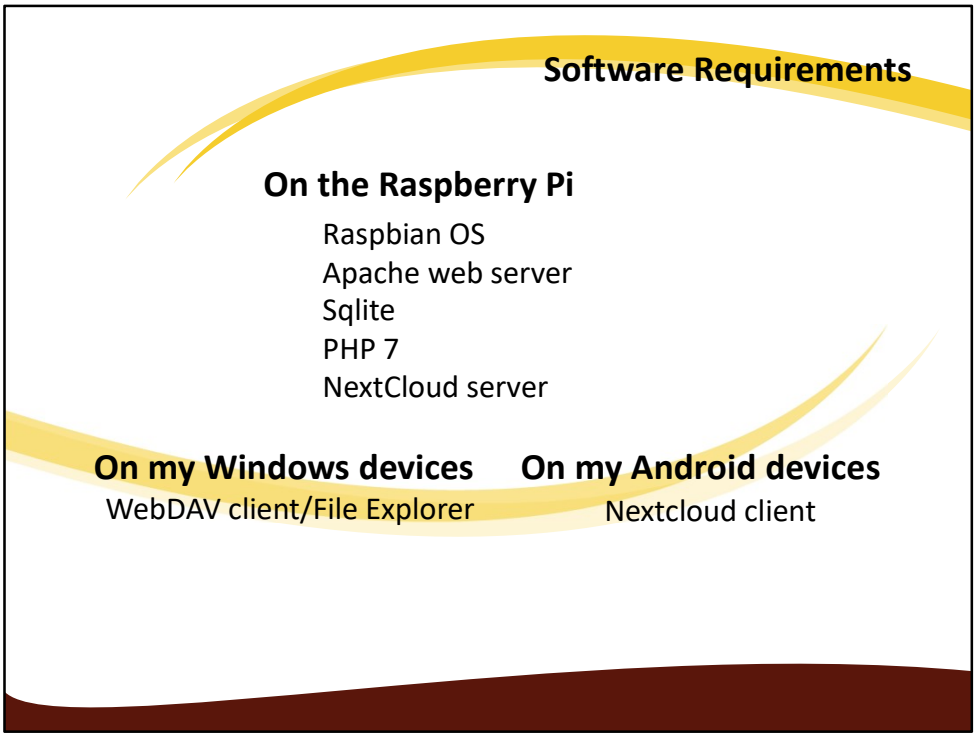


PI model 3B+ and 64GB SD card
A powered USB hub is required as the PI cannot power the hard drive plus keyboard, mouse, and Wi-Fi adapter
Keyboard and mouse only required during set up. The PI can be accessed via Wi-Fi from my laptop or desktop machines
1TB usb hard drive (previously in my laptop but replaced by an SSD unit last summer)
Not shown is the HDMI connection to my monitor – again only necessary during construction
The Pi, Usb HUB, USB hard drive and power connector will be mounted in a small box placed near the router on the floor behind my armchair.



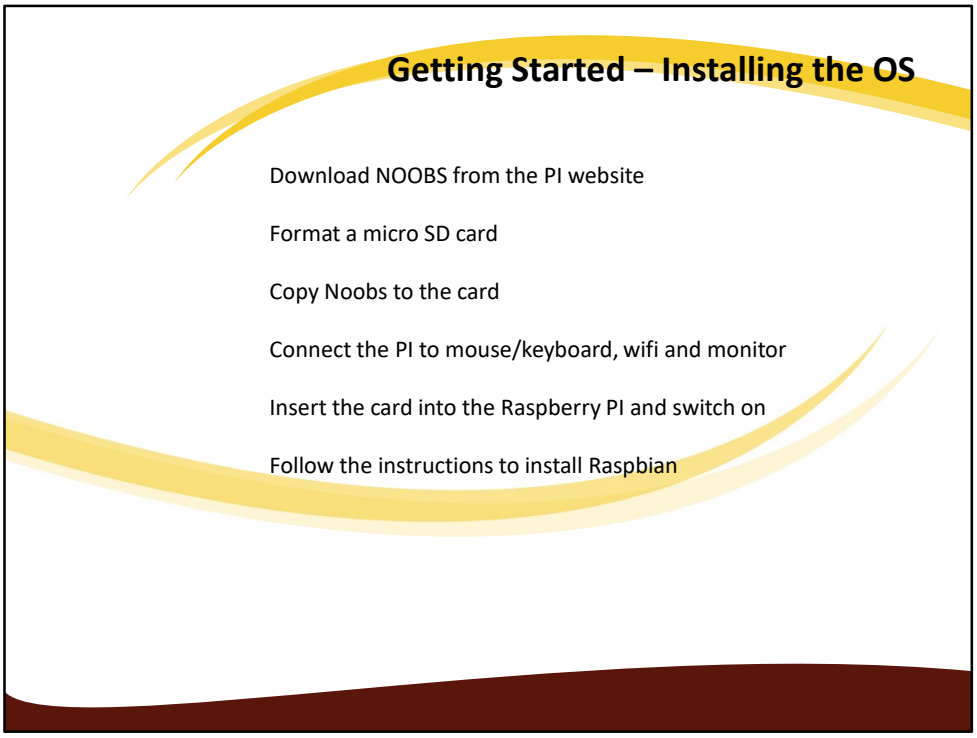
OwnCloud A Dropbox replacement for Linux users, giving many functionalities which are similar to that of DropBox, ownCloud is a self-hosted file sync and share server.
Seafile. file hosting software system which exploits open source property to avail its users with all advantages they expect from a good cloud storage software system.
Pydio. Pydio is a freeware aiming to provide file hosting, sharing and syncing.
Ceph. Didn't really understand the description – very technical. network file system which aims for high performance and large data storage.
Syncany. Only available as a command line tool so forget it
Cozy. Cozy team has introduced Cozy Light which performs well even on cheap hardware's like: Rasberry Pi
GlusterFS. GlusterFS is a network attached file storage system.
StackSync. StackSync is a Dropbox like tool running on top of OpenStack swift which is specially designed to tackle the needs of organizations to sync their data at one place
Yandex.Disk. Yandex.Disk is a cloud storage and synchronization service. Various features provided by Yandex.Disk to its users are: built-in flash player that lets people preview songs, sharing of files with others by sharing download links, synchronization of files between different devices of same user, unlimited storage, WebDAV support allowing easy management of files by any application supporting WebDAV protocol.
Bitcasa. another solution for open source Cloud Storage and synchronization

I also found NextCloud which is a development of Owncloud, but simpler to install.



Rasbian is a Linux distro produced especially to suit the Raspberry Pi. It is a fully working desktop with common applications such Libre Office.
Apache web server is a well known server often used in serious commercial environments.
Sqlite is a simple SQL managed database without all the bells and whistles of a full MYSql database. – also easier to install.
PHP is the standard server side scripting language (for non-Microsoft based servers) It is used to create html pages and communicate with the database. Version 7 is the latest.
NextCloud is the actual server software contained in a Linux 'package' (a zip file).

WebDAV is a protocol for communication via the internet. It can be implemented in apps which provide other functions including set-up. There are apps available for Windows, Linux, Apple and Android systems.



Noobs stands for **New Out Of the Box Software** It contains the OS set up files for Raspbian and several other OS's suitable for the Pi.
I chose 64GB at this stage to make setting up easier.
The complete OS installation process took about 1 hour including fetching updates and installing them

Installing Apache, PHP and Nextcloud

The Linux Way of doing things

All installation etc is done by the root user, when installing Raspbian you do not set a root user so all command in Terminal have to be preceded by 'sudo' this is gives temporary root user rights.

Folders and files are essentially the same as far as the OS is concerned, but in the file manager they behave as folders and files (so why do we have to know this?

Root level folders and documents are not available to a user, authorisation is by inheritance or assignment.

This takes me back to the very early 90's and working with the DOS command line. But much more complicated.

Installing Apache, PHP and Nextcloud

How to install software.

Method 1

Use the built in add/remove app.

Method 2

Use the Terminal and command line.

The Add/remove app in Raspbian is almost unusable and the scripts I have found use the terminal.

Investigated 7 or 8 instructions for installing Owncloud and Nextcloud.
Some were very detailed but out of date so the scripts did not work Others were incomplete and assumed you were a linux export.
I tried 3 different owncloud scripts, and failed at various stages installing the MySQL database was the hardest part as what actually happened did not match the screenshots.
Nextcloud was much simpler as the database was installed automatically as part of the 'Finishing' process.

Installing Apache, PHP and Nextcloud

Typical command line instructions:

```
1. sudo apt-get install apache2

2. sudo apt-get install php7.0 php7.0-gd sqlite php7.0-sqlite
php7.0-curl php7.0-zip php7.0-xml php7.0-mbstring

3. curl https://download.nextcloud.com/server/releases/nextcloud-
13.0.4.tar.bz2 | sudo tar -jxv

4. sudo mkdir -p /var/www/html/nextcloud/data
   sudo chown -R www-data:www-data /var/www/html/nextcloud/
   sudo chmod 750 /var/www/html/nextcloud/data
```

Apache2 installation was a doddle, quick and simple.
PHP was much more complicated
After restart the apache2 server the installation could be tested in the browser with 'localhost/
' to get a default apache2 server page and 'localhost/info.php' to display a page of php7.0 configuration details.

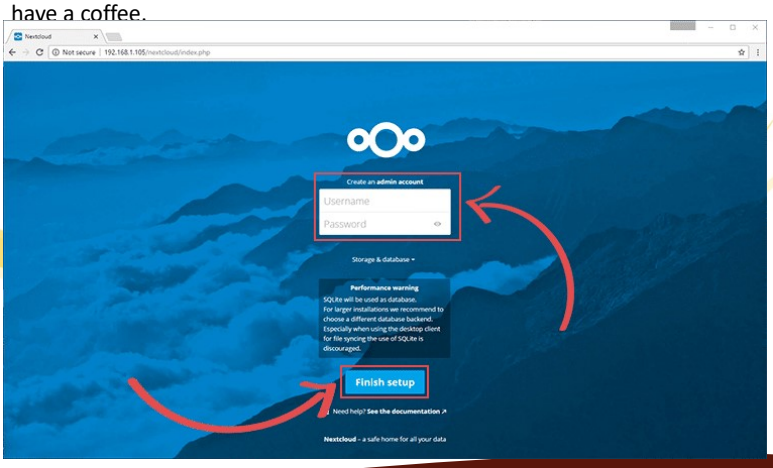
Curl is some form of installation app which downloads a 'package file' unzips it and extracts the files correctly. No idea what jzv stands for some sort of parameters.

The last 3 command create a data directory, assign ownership to user www-data belonging to user group www-data and then assign read-write privileges.
I late found that these privilege levels did not work and had to increase to FULL (777).

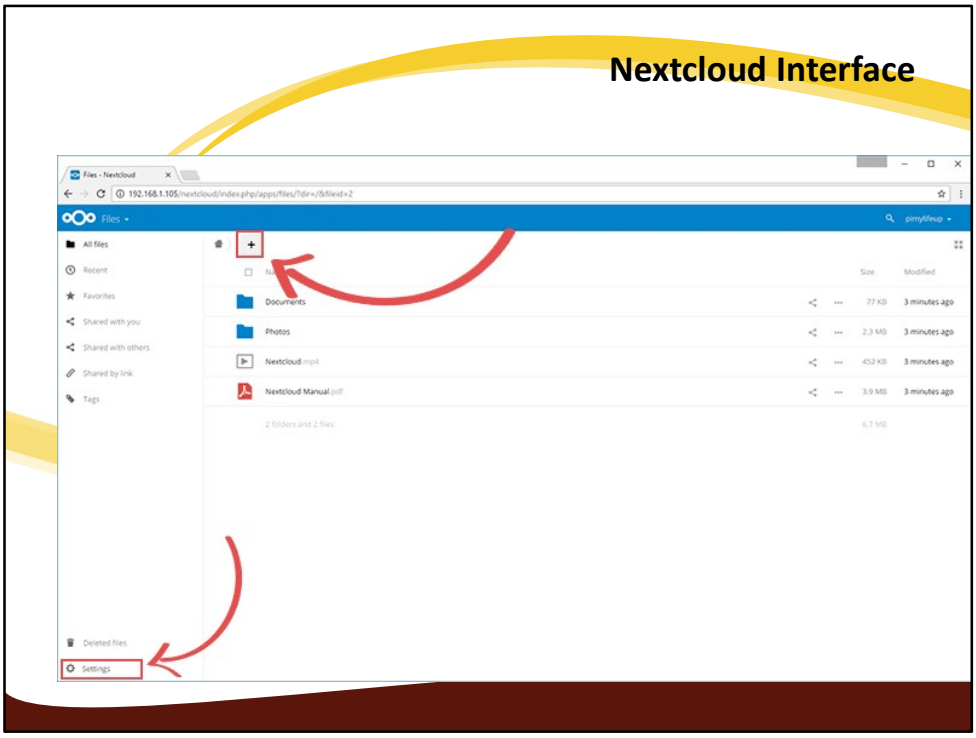
That is the end of the installation.

Nextcloud Set-up

First decide on your user name and password then open Nextcloud in the browser. To do this you need the assigned IP address for the Pi. Or use localhost. 'localhost/nextcloud'. Login and click the Finish button Go and have a coffee.



In this set up Nextcloud installs the Sqlite database , creates the necessary database and content structure. I have no idea what this looks like.
Sqlite is a simple SQL database which is OK for small private clouds. It can only be accessed via the terminal. However I will explore its use on a web server source later.
Currently I have no means of managing the database other than via the terminal command line . I cursory reading of the documentations says that there is a way to import table structures and data from an SQLdump file.
I will investigate this later when I try to extend the pi to being a full web server also.



After closing the Welcome splash screen the actual user interface is displayed.
It is basically a file manager. You upload and down load files from the server. You can give your documents tags, share them with other registered users.
I am the cloud admin so I have additional functions for registering and administering users.
The top arrow indicates the ‘Add file’ button and the bottom arrow the setting button.

Accessing the cloud

Devices inside the home network can access through their browser or android app etc. Directly

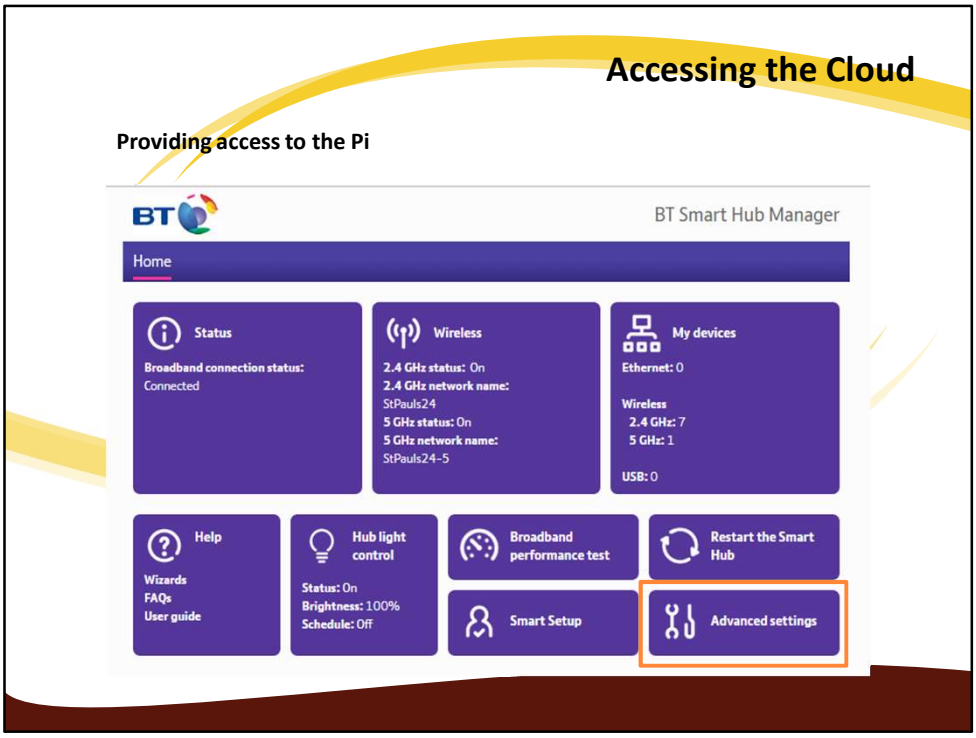
Devises OUTSIDE must use the internet. This requires the server to have a **Domain Name**

Using No-Ip to produce a domain name linked dynamically to my router and thus the cloud server

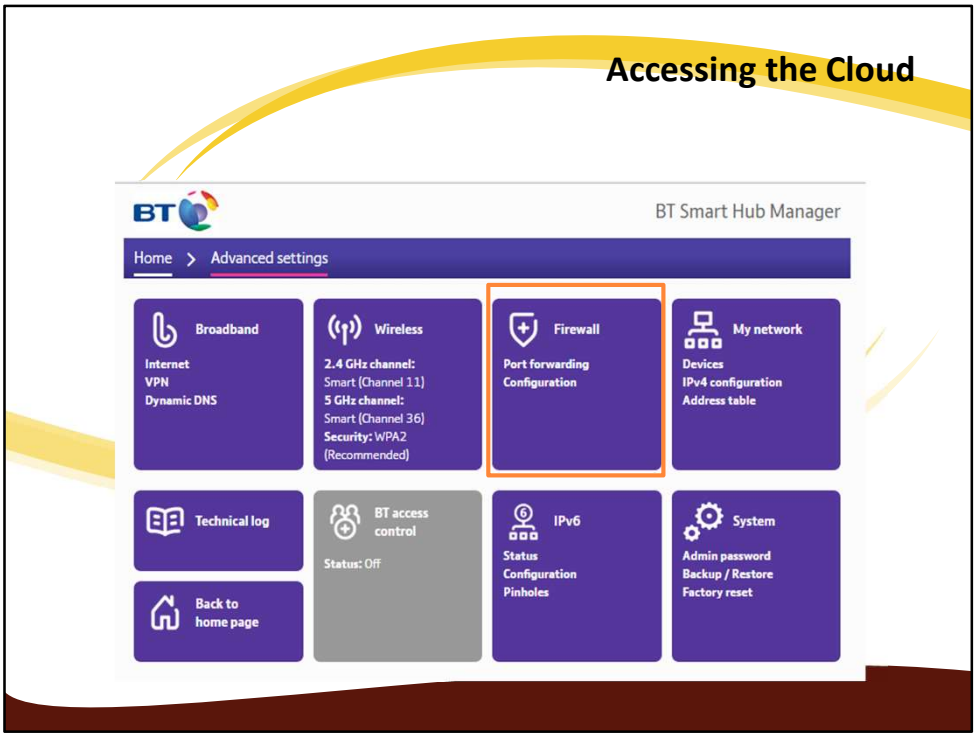
1, Create a no-ip account
2. Set up a domain name and link to my current external IP address
3. Down load the NOIP DUC software which constantly updates No-ip with my current external IP address

Provides access from the internet to the router

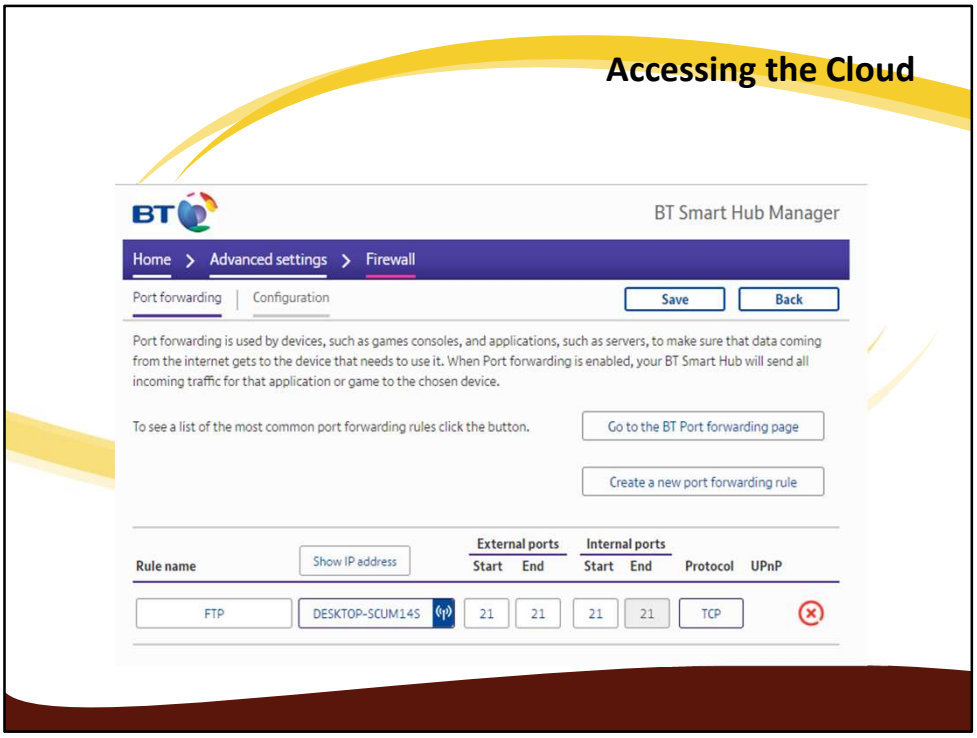
I do not have an official domain name so I need to set one up using a domain name provider. I found No-Ip which provide this service free for simple one-off installations. The account has to be actively renewed every month. (by logging in and clicking a button) The current external IP address is found by asking Google. The DUC software can be installed on the Pi or any other machine on the same WiFi network. As the Pi is constantly on this is the best place.
Generally the external IP address only changes if you restart the router for some reason.



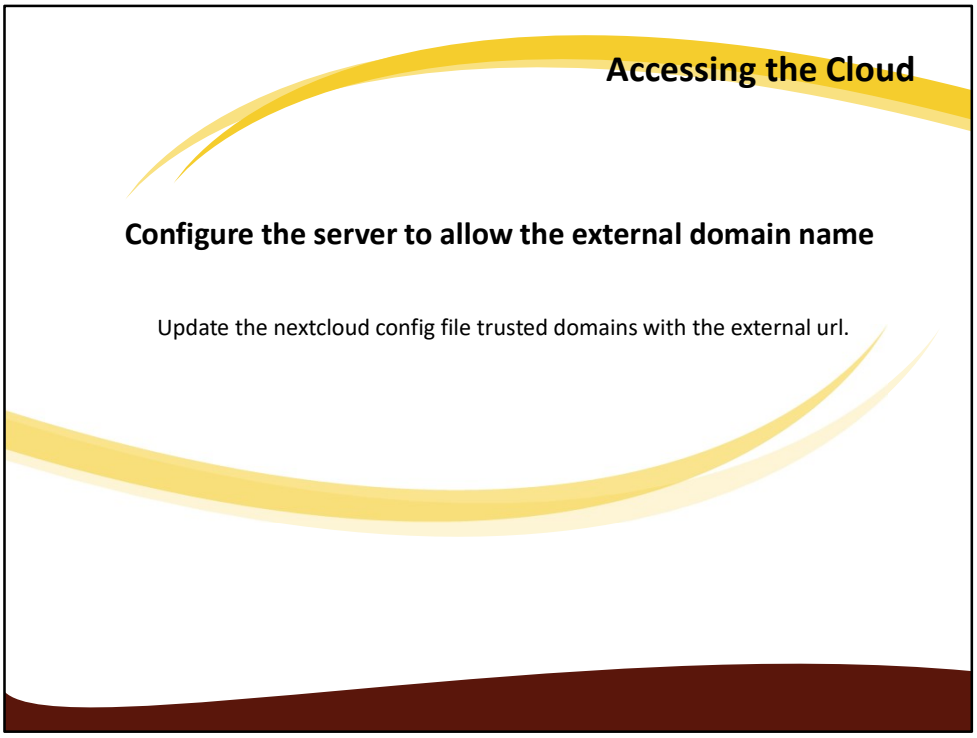
This is the BT Home hub 5, but there are similar functions available on other routers. For best results the Pi should be given a fixed IP address (function NOT available on TalkTalk routers) I am considering the purchase of a Netgear router for the production environment (also better performance overall).
Port Forwarding is found under advanced settings



Port Forwarding appears to be part of the Firewall configuration



Having done this there is now access from the internet to the cloud server, the rest can be done using the phone. Turn off Wifi and enter the address stpaulsview,hopto.org/nextcloud – you get the login page. (You wont because you are not a registered user)



The Nextcloud server is now up and running.
There is however one last configuration required, This is to tell the web server to allow the domain access as a trusted domain.

Refinements

The data directory

This should be moved to another position so that it is not directly attached to the web server. Not a very difficult task takes only a couple of minutes.

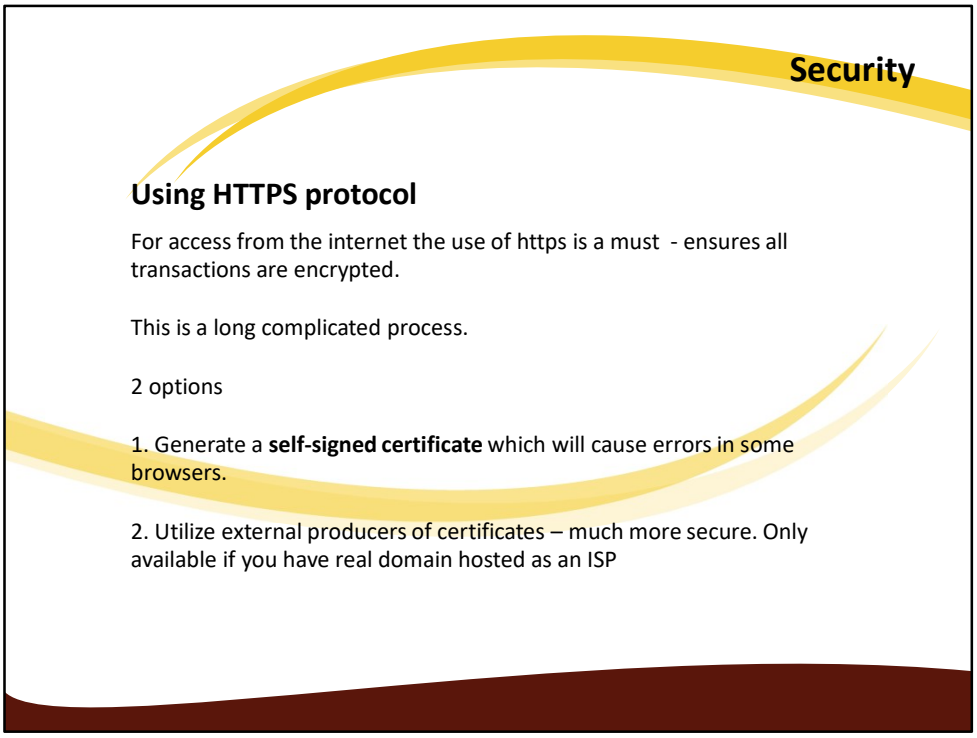
File sizes

The PHP default max file size is only 2MB. This needs to be increased to 1024MB for example

The .htaccess file

Access to this file by apache is required

Having moved the data directory the config file needs updating with the new address.
The file size parameters are in the PHP.ini file and only takes a couple of minutes to find and update.
As always the apache2 server need restarting.
What the htaccess file is I have no idea but the apache2 config file needs to be updated to allow access. Once more restart the server.



There are instruction with Nextcloud on how to set up https using a self-generated certificate.
Let's Encrypt: This site/company can provide proper certificates using SSH. Requires the SSH service and the Acme protocol to be installed on the server. This issues requests for a certificate to Lets Encrypt.
In addition the web server needs to be configured to only accept https requests. Code is available for Nextcloud.

Accessing the server from devices outside the router.

Phones and tablets

There a multitude of apps available on Google store or Apple store.

Windows machines

There are a number of WebDAV clients available or you can map directly from File Explorer

The android phone app works a treat. Having trouble getting the Windows one to work So far have not found a solution. Do not have iOS devices so cannot test these.

Final Words

This has been an interesting, if frustrating project.

The final installation and set up took about 5 hours, although experiments, research, dead-ends etc took up several months (not full time)

Currently I have a functioning cloud server which can only be accessed via my phone or tablet. I need to sort out how to access via my Windows machines.

I also need to fix the security with https.

That done I can construct the box and I am up and running.

References

I looked at the following documents.

Raspberry Pi OwnCloud: Your Own Personal Cloud Storage

OwnCloud on Raspberry Pi

How to set up a Raspberry Pi ownCloud server

Install ownCloud 10 on Raspberry PI 3 with Raspbian Stretch Installed

How to Setup a Raspberry Pi Nextcloud Server

(The last one was the one I finally used)

The NOIP web site