NAS and Docker RASPBERRY PI

ABSTRACT

Our project deals with cloud services on the raspberry pi.

NAS has been setup using Samba server on the raspberry pi. A docker image has been installed which is not compatible with pi usually, but has been configured to work with raspberry pi. In addition to it, **Pi-hole, Tonido,** and **Torrentbox** has been setup on the pi.

RASPBERRY PI

The Raspberry Pi is a series of small single-board computers developed in the United Kingdom by the Raspberry Pi Foundation to promote the teaching of basic computer science in schools and in developing countries. The original model became far more popular than anticipated, selling outside of its target market for uses such as robotics. Peripherals (including keyboards, mice and cases) are not included with the Raspberry Pi. Some accessories however have been included in several official and unofficial bundles.

NAS

Network-attached storage (NAS) is a file-level computer data storage server connected to a computer network providing data access to a heterogeneous group of clients. NAS is specialized for serving files either by its hardware, software, or configuration. It is often manufactured as a computer appliance – a purpose-built specialized computer. NAS systems are networked appliances which contain one or more storage drives, often arranged into logical, redundant storage containers or RAID.

PI_HOLE

You have software to block ads on your computer, but if you want to block ads on all your devices—from your smartphone to your tablets—you'll need something a little stronger. Enter the Pi-Hole, a Raspberry Pi image that blocks ads of all sorts at the router level.

TONIDO

Tonido Server allows you to access all your files on your computer from a web browser, smartphone, tablet or even DLNA enabled devices.

Tonido is basically a simple platform to make all your files available over the internet remotely, turning your computer or device to a cloud platform.

DOCKER

Docker is an open-source project that automates the deployment of applications inside software containers. Docker provides an additional layer of abstraction and automation of operating-system-level virtualization on Windows and Linux. Docker uses the resource isolation features of the Linux kernel such as cgroups and kernel namespaces, and a union-capable file system such as OverlayFS and others to allow independent "containers" to run within a single Linux instance, avoiding the overhead of starting and maintaining virtual machines.

On the raspberry pi, docker is only supported till version 1.3.3. We installed and configured a docker image to version 1.10

TORRENTBOX

Torrentbox basically turns the raspberry pi into a box which will download all your torrents. It makes use of the transmission bit torrent client and a web interface which can be configured for remote access. This ensures that you can manage and download your files to the pi from anywhere.

IMPLEMENTATION

For NAS

First add support for ntfs by installing ntfs-3g, to make the hard drive available to all systems.

Check the drives attached to the pi using the 'sudo fdisk -l' command.

Create a folder for mounting the HDD to it.

```
sudo mkdir /media/USBHDD1
```

After creating the directory, mount the usb hdd to it

```
sudo mount -t auto /dev/sda1 /media/USBHDD1
```

Now install SAMBA:

```
sudo apt-get install samba samba-common-bin
```

In the samba config file located at /etc/samba/samba.conf, make the following addition to the bottom:

```
[Backup]
comment = Backup Folder
path = /media/USBHDD1/shares
valid users = @users
force group = users
create mask = 0660
directory mask = 0771
read only = no
```

Now restart the Samba daemon:

```
sudo /etc/init.d/samba restart
```

At this point we need to add in a user that can access the Pi's samba shares. We're going to make an account with the username backups and a password.

```
sudo useradd backups -m -G users
sudo passwd backups
```

You'll be prompted to type in the password twice to confirm. After confirming the password, it's time to add "backups" as a legitimate Samba user.

```
sudo smbpasswd -a backups
```

Now, just access the server from any machine on the network.

For pi-hole

Installation of pi-hole was done using the following command:

```
curl -sSL https://install.pi-hole.net | bash
```

This fetches the pi hole installation package from the web.

Configuration was done by setting up the pi-hole as our dns server by changing the dhcp options, so all traffic on the network will pass through it.

This will then block all advertisements on the network.

For torrentbox

First, install the transmission client by using:

```
sudo apt-get install transmission-daemon
```

Then, to combine with the NAS, make folders on the external HDD drive to act as destination for the torrents.

```
sudo mkdir -p /media/NASHDD1/torrent-inprogress
sudo mkdir -p /media/NASHDD1/torrent-complete
```

Now, make some changes to the settings file of the transmission daemon.

```
sudo nano /etc/transmission-daemon/settings.json
```

Access the above file, and make the following changes, adding a username and a password to prevent unidentified access to the torrents, and also refer to the above made directories.
Whitelist is the ip addresses which can access the torrentbox via the web.

```
"incomplete-dir": "/media/NASHDD1/torrent-inprogress",
"incomplete-dir-enabled": true,
"download-dir": "/media/NASHDD1/torrent_complete",
"rpc-password": "Your_Password",
"rpc-username": "Your_Username",
"rpc-whitelist": "192.168.*.*",
```

For Tonido

To download and run Tonido on the Pi:

```
sudo mkdir /usr/local/tonido
cd /usr/local/tonido
sudo wget http://patch.codelathe.com/tonido/live/installer/armv6l-rpi/tonido.tar.gz
sudo tar -zxvf tonido.tar.gz
./tonido.sh start
```

Open Tonido Website using the address http://<RaspberryPi IP Address>:10001 to begin the setup process.

so for example: http://192.168.0.104:10001

This will redirect to the tonido login page at first.

Now, create a tonido account for allowing remote logins.

After providing the email address and password, configure it to give access to particular files and folders.

Port forwarding can be setup to 10001 to setup remote access to Tonido and all your files over the internet, thus making the Pi as our very own cloud server.

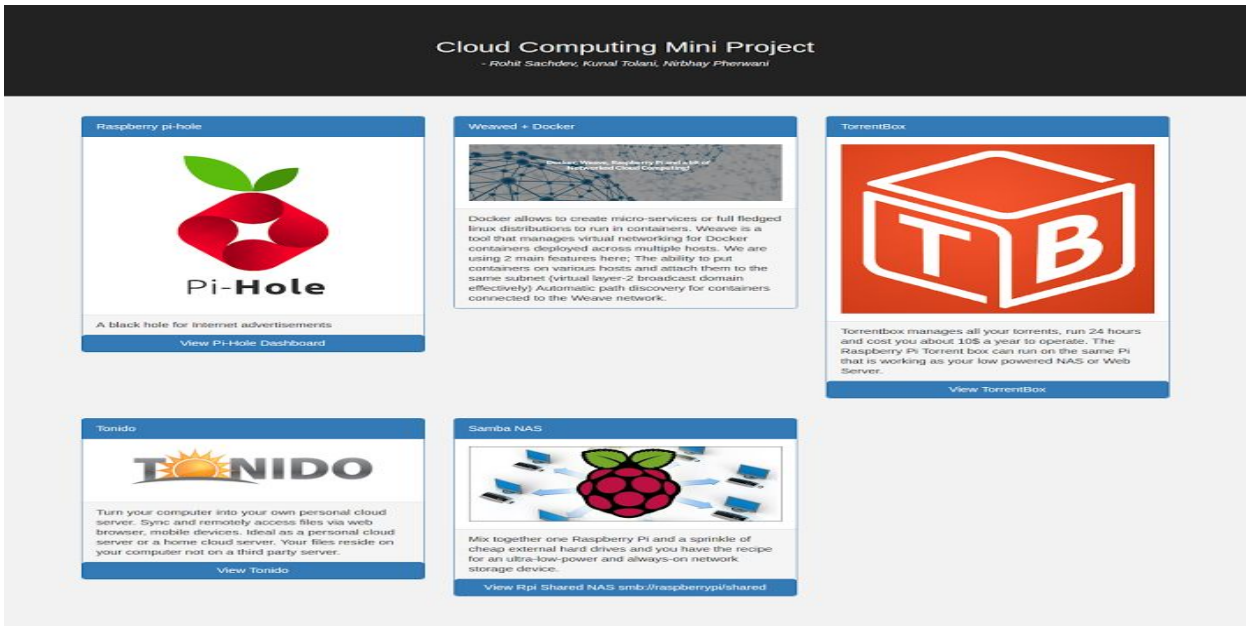
For Docker

We used a configured docker image from HypriotOS to install docker version 1.10 on the pi, which is normally not supported on the Pi.

Weaved was run in a docker container, which enables connection between docker containers on different hosts by creating a virtual network.

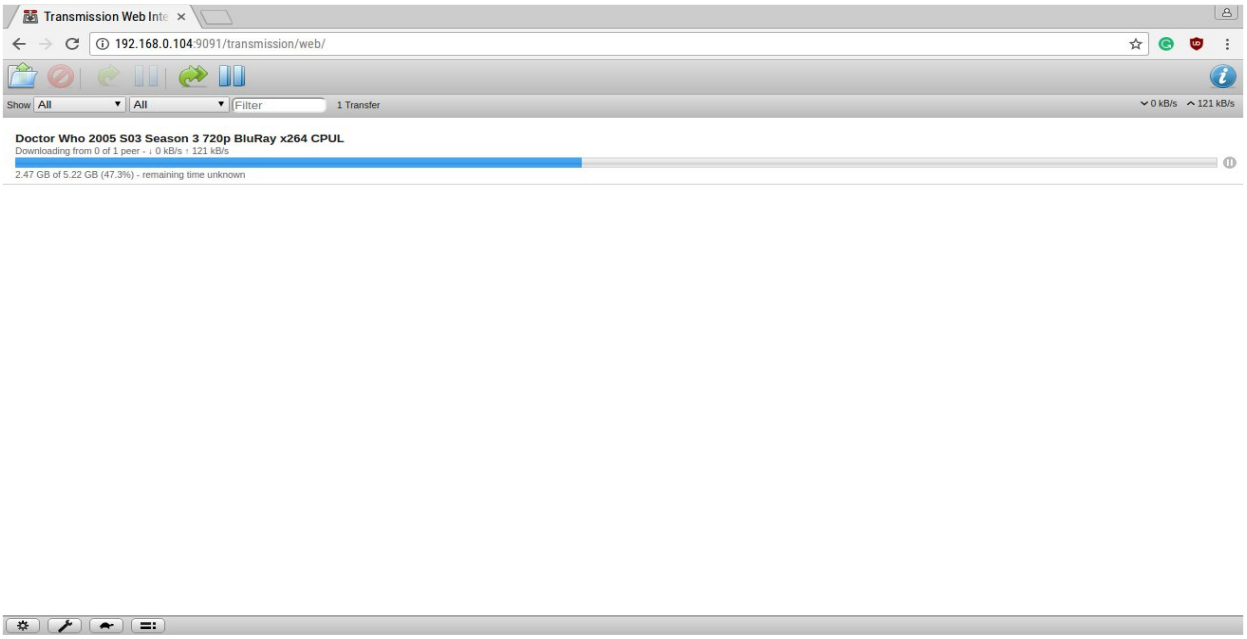
RESULT AND SNAPSHOTS

Web view of the cloud based service



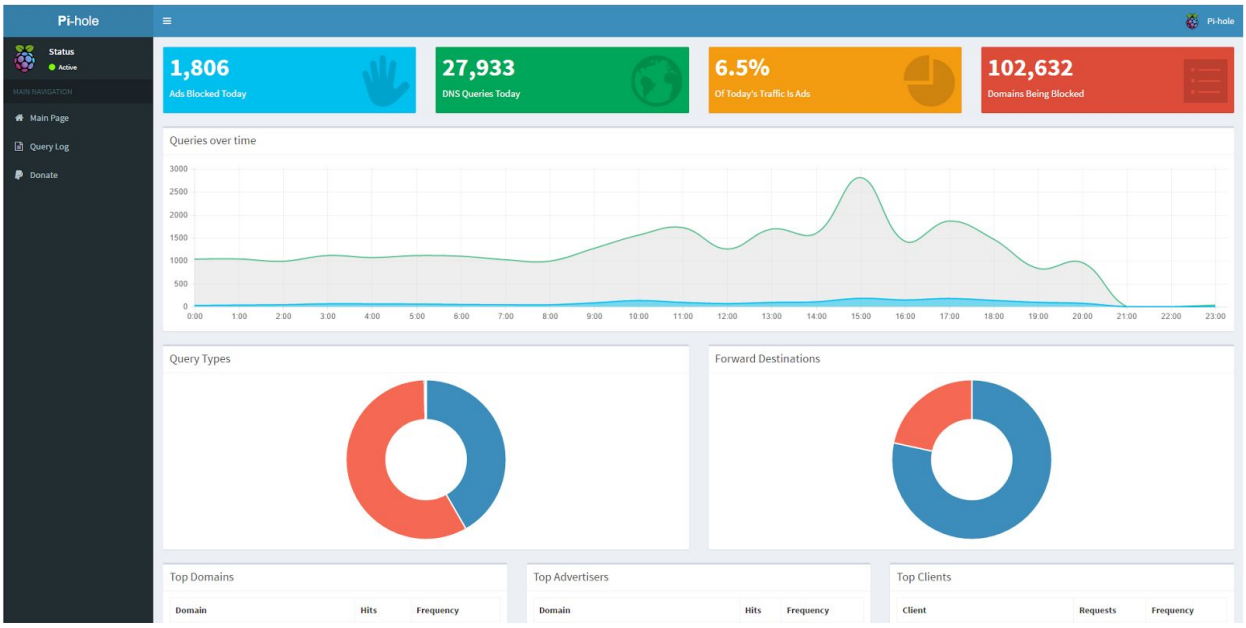
(Fig. 1. Web page)

Torrentbox has been setup, and is running



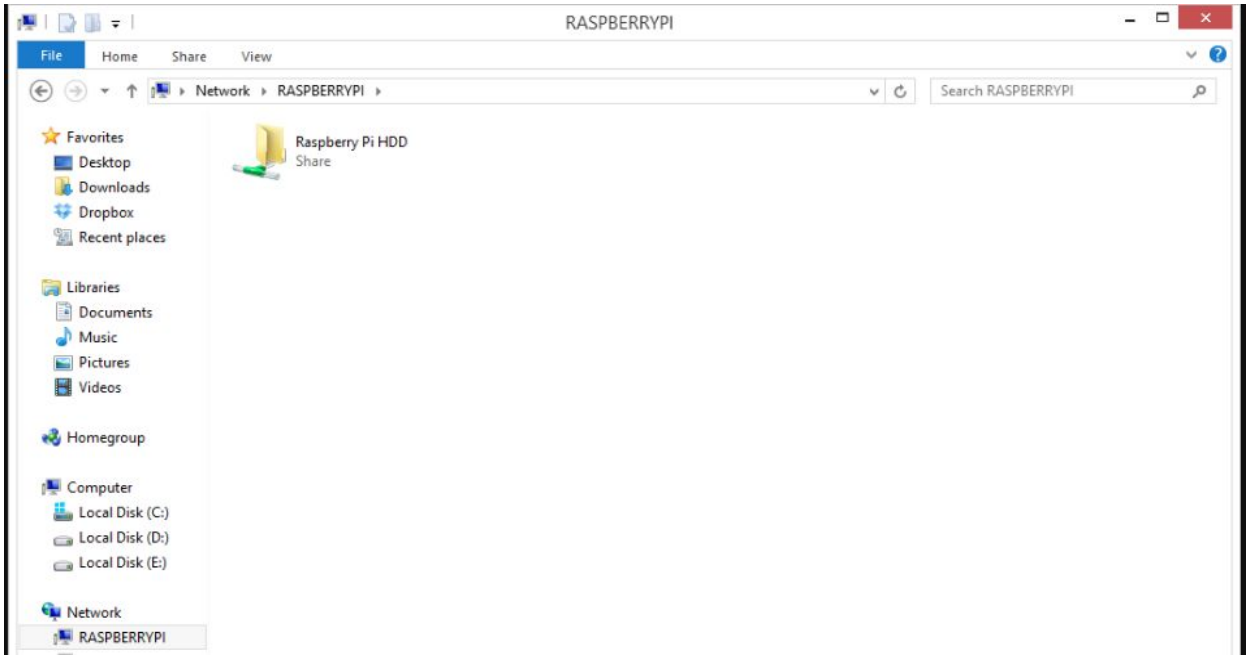
(Fig. 2. Torrentbox)

Pi hole:



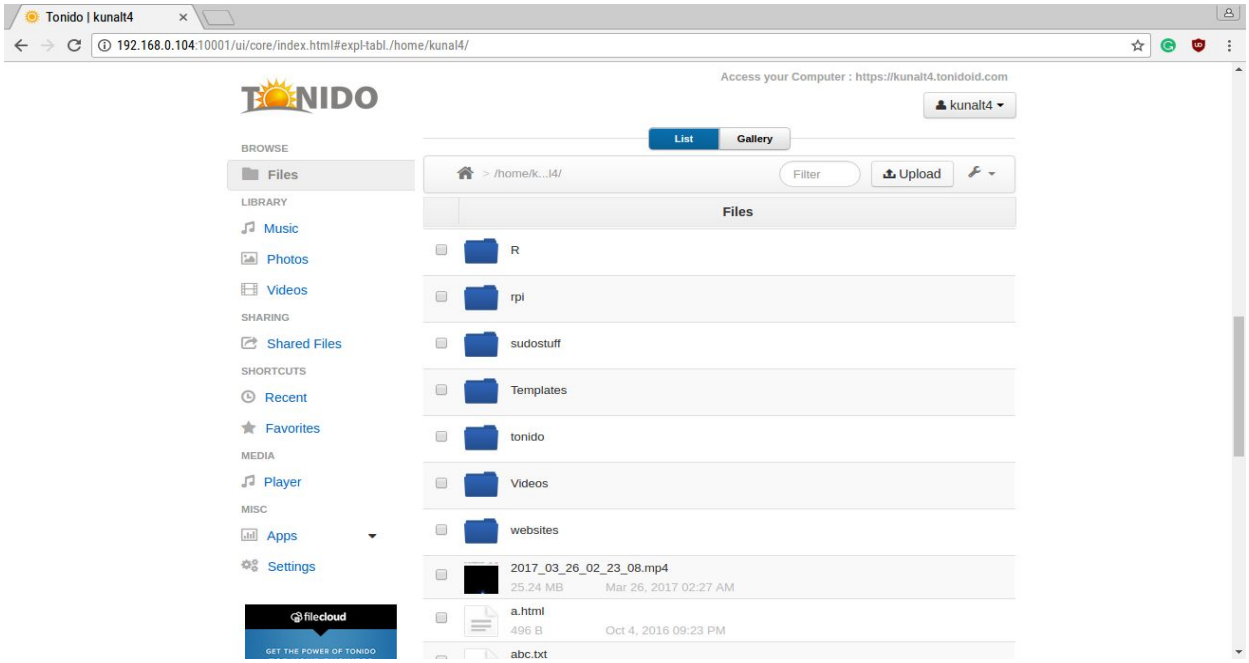
(Fig. 3. Pi-hole)

Accessing the NAS server from a Windows machine on the same network:



(Fig. 4. NAS)

Tonido:



(Fig. 5. Tonido)

Conclusion

We setup a multi-purpose cloud service on the raspberry pi.

Docker, Torrentbox, Pihole, Tonido and NAS using Samba, all have been setup and running on the Pi.

References

1. http://www.tonido.com/support/display/docs/Installation
2. http://www.tonido.com/support/display/docs/Configuration
3. https://pimylifeup.com/raspberry-pi-torrentbox/
4. http://blog.hypriot.com/getting-started-with-docker-on-your-arm-device/
5. https://pi-hole.net/
6. https://pimylifeup.com/raspberry-pi-nas/