

AvoidErrors

Turn Your Raspberry Pi 3 into a Personal Cloud Storage 2017

96
Shares

Share

Tweet

Share

Share

Our goal of installing ownCloud on the Raspberry pi is not just installing ownCloud but also mounting an external drive for larger capacity of data storage and to be able to access it from anywhere in the world as if you where using Dropbox or OneDrive.

OwnCloud is free robust open source software to set up your own syncing, Dropbox-like cloud storage..

Advantages: No monthly payments, added security and unlimited files storage. This process will always install the latest version of Owncloud.

This tutorial assumes that you have a fully functional **Raspberry pi-3** with **Raspbian (Full Desktop) installed** with internet access, and a **8 GB Micro**

Search the site



Newsletter

Get weekly Video Tutorials tips and trends via email!

[Join Now](#)

100% Privacy. We don't spam.

Popular Posts



Install ownCloud 10 on Raspberry PI 3 ...

October 6, 2017 • 44 Comments

SD card Minimum!. If you haven't installed Raspbian then check out my guide on [how to install Raspbian](#) via NOOBS.

There are many ways to install ownCloud onto a Raspberry Pi. If you don't have a Raspberry PI I would personally recommend buying the **Raspberry pi-3** kit with a **32 GB Micro SD card** so that you don't have to mount an external drive. Unless you want to store more than 32GB of data.

Turn your Raspberry PI 3 Into a Personal Cloud Storage



Setting up Raspberry Pi for Owncloud:

1. Turn on the Raspberry Pi. From the terminal or SSH enter the following command:

```
sudo raspi-config
```



The following changes needs to be made in the Raspberry Pi configuration



Install OwnCloud 9 on Raspberry Pi 3 ...

June 12, 2016 • 30 Comments



How to Restore Windows Photo Viewer Windows ...

September 7, 2015 • 22 Comments



Turn Your Raspberry Pi 3 into a ...

June 18, 2016 • 22 Comments



Reset Windows 7 Login Password Without Third ...

December 19, 2016 • 16 Comments



How To Burn an ISO Image In ...

March 4, 2011 • No Comment

a. Expand the root filesystem to have enough space for the cloud

Select “**Expand Filesystem**”

b. Change user password

“**For Security when accessing form the WAN**”

c. Change locale to en_US.UTF8

Select “**Internationalisation Options**”

d. Memory split, allocate 16M to video graphics

Select “**Advanced Options**” → “**Memory Split**”



Install Owncloud:

Go to owncloud.org and enter the commands as shown in the video. I would suggest to copy and paste the commands from owncloud so that you enter the following commands:

```
sh -c "echo 'deb
```

```
http://download.owncloud.org/download/repositories/stable/Debian_8.0/' >  
/etc/apt/sources.list.d/owncloud.list"
```

```
wget -nv
```

```
https://download.owncloud.org/download/repositories/stable/Debian_8.0/Release.key  
-O Release.key
```

```
apt-key add - < Release.key
```

```
apt-get update
```

```
apt-get install owncloud
```

Mount and Set up an External Hard Drive

These instructions are for mounting an NTFS formatted hard drive for maximum capacity storage, and allowing ownCloud to store files onto it.

1. For an NTFS drive we will need to install a NTFS package by entering the following:

```
sudo apt-get install ntfs-3g
```

2. Create a directory we can mount the drive to:

```
sudo mkdir /media/ownclouddrive
```

3. Create and add the www-data user to the www-data group:

```
sudo groupadd www-data
sudo usermod -a -G www-data www-data
```

4. Make the user www-data owner of the mounted drive and make its permissions read, write and execute:

```
sudo chown -R www-data:www-data /media/ownclouddrive
sudo chmod -R 775 /media/ownclouddrive
```

5. Now we need to get the gid, uid and the uuid as we will need to use them so the pi will remember it even if we plug it into a different USB port. Enter the following command for the gid:

```
id -g www-data
```

6. Now to get the uid enter the following command:

```
id -u www-data
```



7. We need to get the UUID of the attached external hard drive so the Pi can remember this drive even if you plug it into a different USB port.

```
ls -l /dev/disk/by-uuid
```



8. Then copy the light blue letters and numbers of the **sda1** entry usually located on the bottom. It should look something like (**numbers&letters** -> **../sda1**). See picture below:

```
pi@raspberrypi:~ $ sudo ls -l /dev/disk/by-uuid
total 0
lrwxrwxrwx 1 root root 15 Jun 15 01:36 45FD-4023 -> ../../mmcblk0p1
lrwxrwxrwx 1 root root 15 Jun 15 01:36 5CA5-3ACE -> ../../mmcblk0p6
lrwxrwxrwx 1 root root 15 Jun 15 01:36 6e910782-ffa9-4a27-b398-030be505d14b -> ../../mmcblk0p5
lrwxrwxrwx 1 root root 10 Jun 15 01:36 86B2A30DB2A300B3 -> ../sda1
lrwxrwxrwx 1 root root 15 Jun 15 01:36 1137c73c-bd30-41ee-a5d1-719946401c91 -> ../../mmcblk0p7
pi@raspberrypi:~ $
```

Auto mount the external drive and then check if it is accessible at /media/owncloudrive:

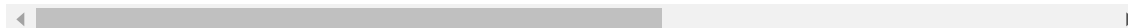
9. Add your drive into the fstab file so it'll boot with the proper permissions.

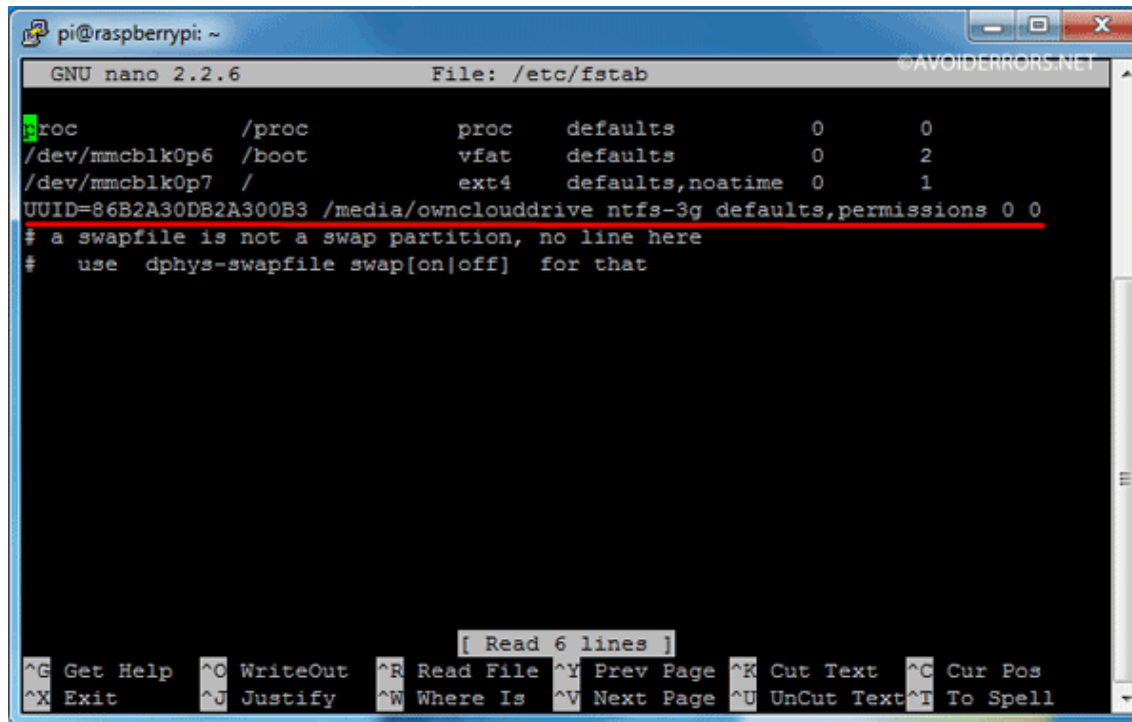
```
sudo nano /etc/fstab
```



10. Add the following line to the bottom of the file, updating the **UUID** with the values you got from the above step 5. (It should all be a single line)

```
UUID=xxxxxxxx /media/owncloudrive auto nofail,uid=33,g
```





```
pi@raspberrypi: ~
GNU nano 2.2.6      File: /etc/fstab
proc                /proc              proc               defaults           0                 0
/dev/mmcblk0p6      /boot              vfat               defaults           0                 2
/dev/mmcblk0p7      /                  ext4               defaults,noatime   0                 1
UUID=86B2A30DB2A300B3 /media/owncloudrive ntfs-3g defaults,permissions 0 0
# a swapfile is not a swap partition, no line here
#   use dphys-swapfile swap[on|off] for that

[ Read 6 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

11. Reboot the Raspberry Pi:

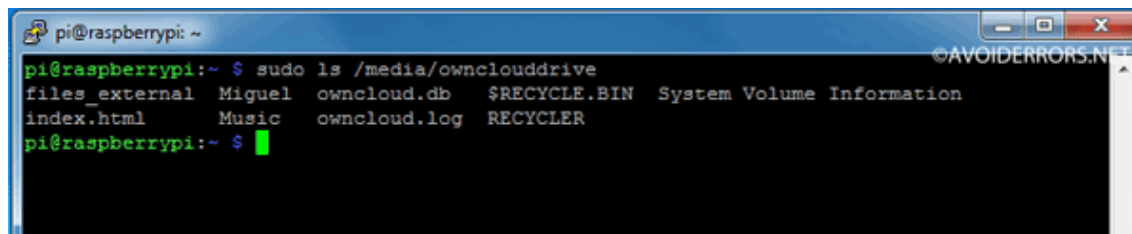
```
sudo reboot
```

</>

12. Now the external hard drive should automatically be mounted. If mounted we're all good to go. To check it enter:

```
sudo ls /media/owncloudrive
```

</>



```
pi@raspberrypi: ~
pi@raspberrypi:~ $ sudo ls /media/owncloudrive
files_external Miguel owncloud.db $RECYCLE.BIN System Volume Information
index.html Music owncloud.log RECYCLER
pi@raspberrypi:~ $
```

If there are files inside the drive you should be able to see them here. If it's an empty drive you might not see anything so don't be surprised if nothing shows up.

If you want to be absolutely positive that the drive is properly mounted, unplug the drive from the RPI, plug it onto a PC since it's formatted NTFS it should be easily detected by the PC, open it, create a .txt file name it test, then eject it and plug it back to the RPI, reboot the RPI, and re-run the command:

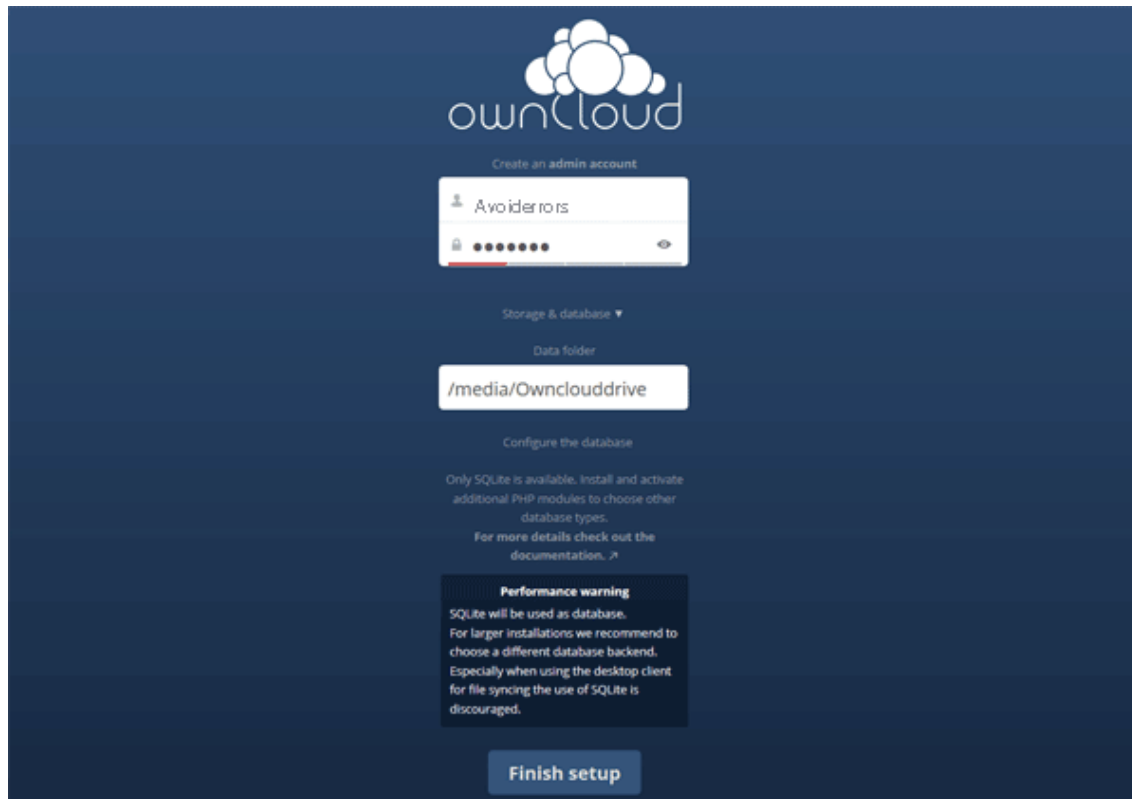
```
sudo ls /media/owncloudrive
```

```
</>
```

The .txt file you created should be there.

Basic First Access Setup

1. Open your browser and enter the IP address provided, in my case is 107.170.39.236 you'll be directed to your ownCloud storage server.
2. Because we installed SSL for more security you might get a certificate error, simply add the certificate to your exception list to proceed.
3. You should be presented with a simple setup screen, Here enter a username and password to create an admin account.
4. Click on **Storage & database** dropdown and enter your external hard drive: **/media/owncloudrive** (Skip this if you didn't setup an external drive).
5. Click **Finish setup**.



Done! !! all is left is to configure your devices to access your ownCloud storage.

Enable SSL for Secure External Access

This process only applies if you are planning on connecting to the Raspberry PI ownCloud from the WAN (outside the network) using mobile devices or tablet. Enable and configure SSL on Owncloud 9 in Raspberry PI ownCloud.

1. Open the terminal and run the following command:

```
sudo a2enmod ssl
```



2. Create a new directory for the self signed certificate:


```
sudo mkdir /etc/apache2/ssl
```

</>

3. Create the self signed certificate and the server key that protects it, and placing both of them into the new directory:

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048
```

Note: The above certificate is valid for 365 days. We have mentioned validity days in the command.

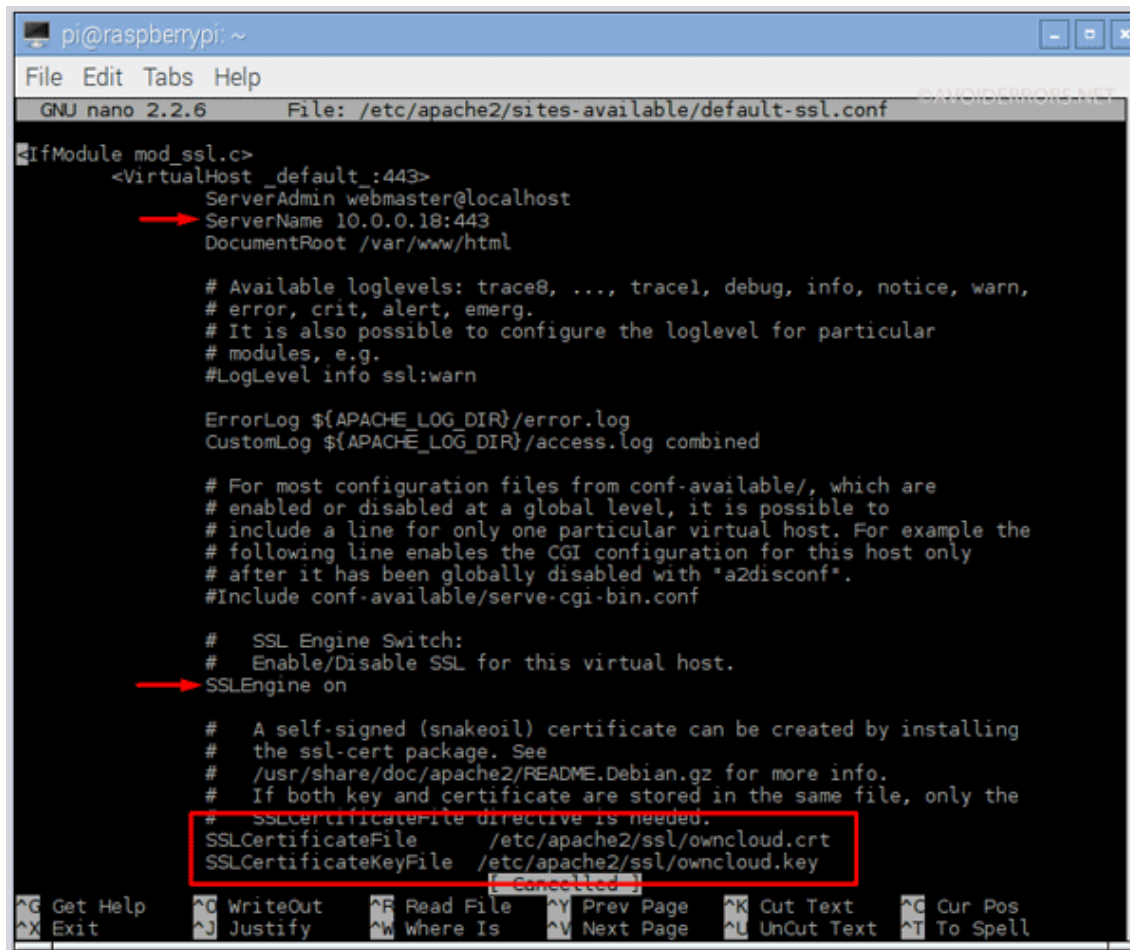
4. Now we need to setup the certificate:

```
sudo nano /etc/apache2/sites-available/default-ssl.conf
```

5. The lines that need changing are the following:

```
ServerName RPI IP :443
SSLEngine on
SSLCertificateFile /etc/apache2/ssl/owncloud.crt
SSLCertificateKeyFile /etc/apache2/ssl/owncloud.key
```

</>



```
pi@raspberrypi: ~
File Edit Tabs Help
GNU nano 2.2.6 File: /etc/apache2/sites-available/default-ssl.conf
<IfModule mod_ssl.c>
  <VirtualHost _default_:443>
    ServerAdmin webmaster@localhost
    ServerName 10.0.0.18:443
    DocumentRoot /var/www/html

    # Available loglevels: trace8, ..., tracel, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf

    #
    # SSL Engine Switch:
    # Enable/Disable SSL for this virtual host.
    SSLEngine on

    #
    # A self-signed (snakeoil) certificate can be created by installing
    # the ssl-cert package. See
    # /usr/share/doc/apache2/README.Debian.gz for more info.
    # If both key and certificate are stored in the same file, only the
    # SSLCertificateFile directive is needed.
    SSLCertificateFile /etc/apache2/ssl/owncloud.crt
    SSLCertificateKeyFile /etc/apache2/ssl/owncloud.key
  
```

6. Activate the new virtual host

```
sudo a2ensite default-ssl
```

</>

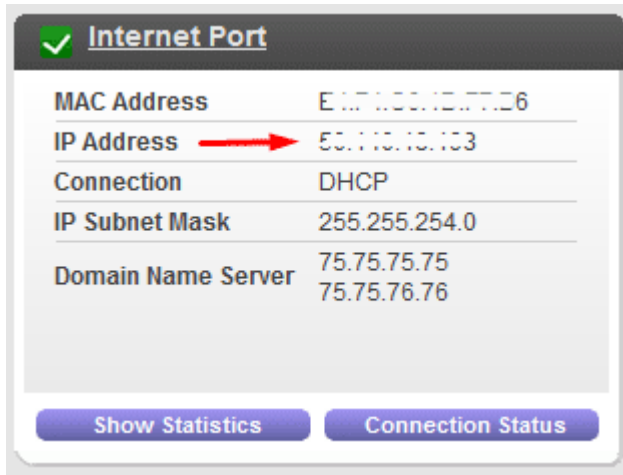
7. Restart apache

```
sudo service apache2 restart
```

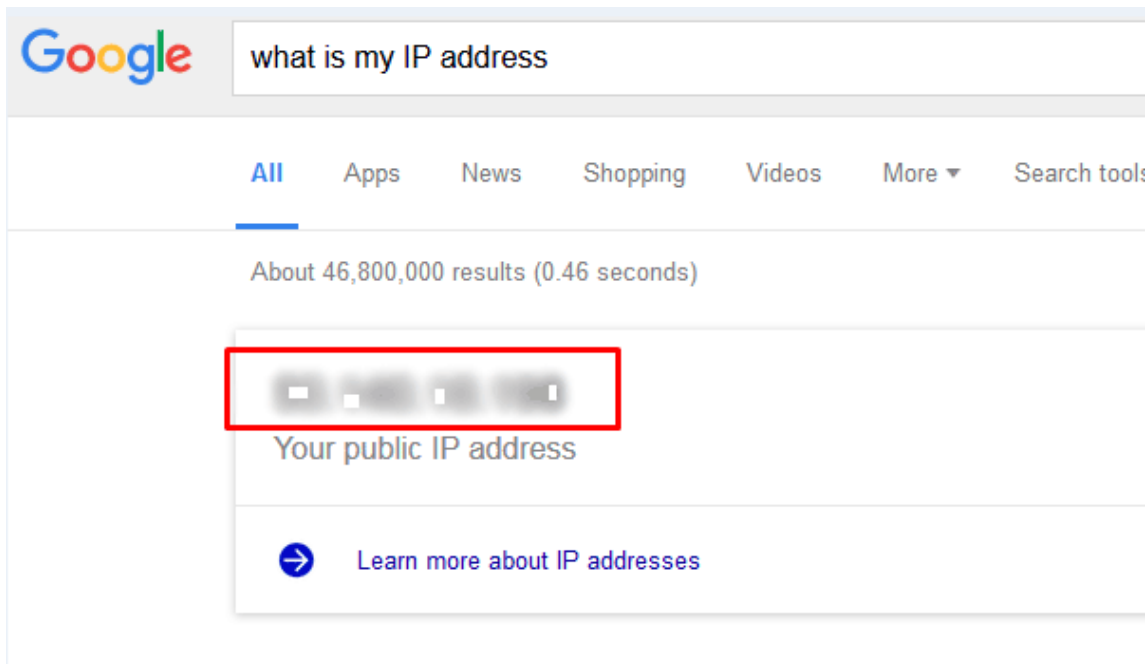
</>

Port Forward for External Access

Log into your router and get the WAN IP address:



Or Google what is my IP address?



Now we need to add the WAN IP to your trusted IP list and not to be overwritten by ownCloud. To do this open the Owncloud config file, enter:

```
sudo nano /var/www/owncloud/config/config.php
```

</>

Here add the **WAN IP** (External IP address) you just got from the router or Google to the trusted domains array. Your new entry should look something like this:

```
1 => 'xxx.xxx.xxx.xxx',
```

</>

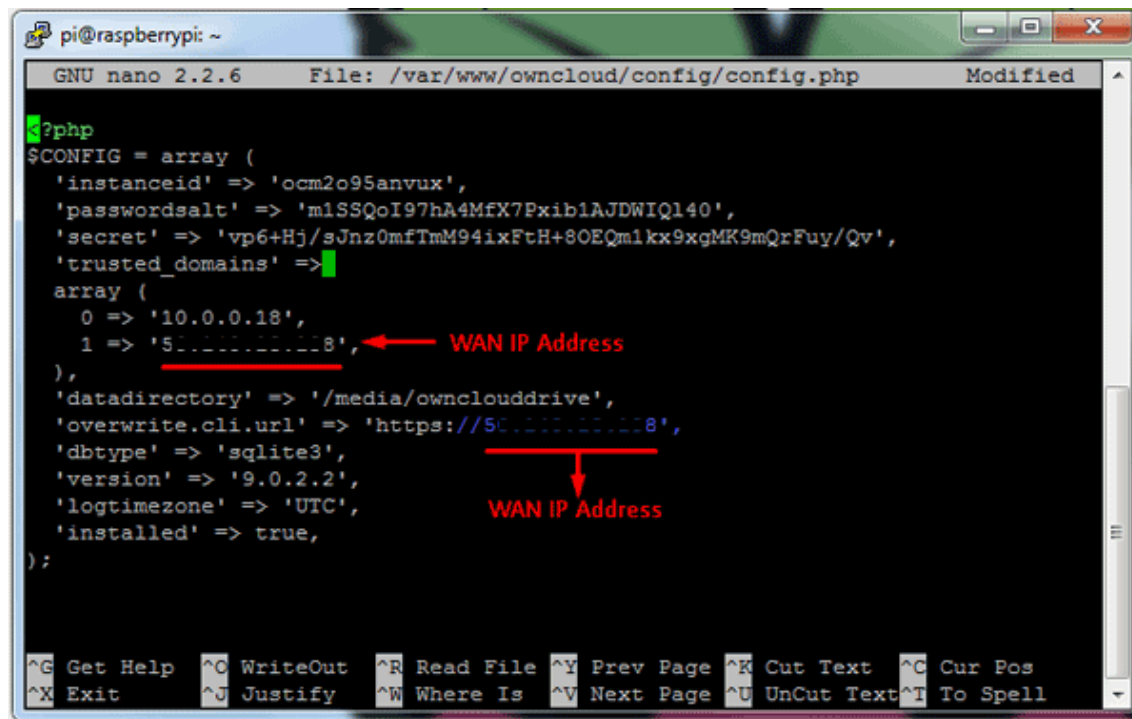
X are just placeholders. Replace the **X**'s with the WAN IP Address.

Now update the URL of the **overwrite.cli.url** line with your WAN IP Address. It should look something like this:

```
'overwrite.cli.url' => 'https://xxx.xxx.xxx.xxx',
```

</>

Here is an example of the completed config.txt file.



```
pi@raspberrypi: ~
GNU nano 2.2.6 File: /var/www/owncloud/config/config.php Modified
?php
$CONFIG = array (
  'instanceid' => 'ocm2o95anvux',
  'passwordsalt' => 'm1SSQoI97hA4MfX7Pxib1AJDWIQ140',
  'secret' => 'vp6+Hj/sJnz0mfTmM94ixFtH+8OEQm1kx9xgMK9mQrFuy/Qv',
  'trusted_domains' =>
    array (
      0 => '10.0.0.18',
      1 => '50.117.100.118', ← WAN IP Address
    ),
  'datadirectory' => '/media/ownclouddrive',
  'overwrite.cli.url' => 'https://50.117.100.118',
  'dbtype' => 'sqlite3',
  'version' => '9.0.2.2',
  'logtimezone' => 'UTC',
  'installed' => true,
);

^G Get Help ^C WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

Once done save and restart the nginx service:

```
sudo service nginx restart
```

</>

Now log into your router and navigate to the port forward section.

NETGEAR genie

Nighthawk R7000

Firmware Version V1.0.6.28_1

English

BASIC **ADVANCED**

ADVANCED Home
Setup Wizard
WPS Wizard
Setup
USB Functions
NETGEAR Downloader (BETA)
Security
Administration
Advanced Setup

Wireless Settings
Router / AP / Bridge / Repeating Mode
Port Forwarding / Port Triggering
Dynamic DNS
VPN Service
Static Routes
Remote Management
UPnP
IPv6
Traffic Meter

Port Forwarding / Port Triggering

Please select the service type.

☒ Port Forwarding
☐ Port Triggering

Service Name: FTP Server IP Address: 10.0.0.0 + Add

#	Service Name	External Start Port	Internal Start Port	Internal IP address
<p>Edit Service Delete Service + Add Custom Service Arrange by Internal IP</p>				

Now port forward SSL port **443** to the Raspberry pi internal IP (LAN IP) address and save settings.

Ports - Custom Services

Apply Cancel

Service Name: ownCloud

Protocol: TCP/UDP

External Port Range: 443 → **SSL Port** (1~65534)
(Specify ports and/or port ranges spilit by commas, example: 30, 50-60, 65500-65510)

☒ Use the same port range for Internal port

Internal Port Range: 443 (1~65534)
(Specify ports and/or port ranges spilit by commas, example: 30, 50-60, 65500-65510)

Internal IP address: 10.0.0.18 → **RPI IP**

Now your RPI ownCloud is ready to be accessed externally (WAN) and from your devices just download the ownCloud App and enter: “***https:// WAN IP Address***” on the address bar or devices. below is an example:



About The Author



Miguel

I started this tech blog back in 2011 as a place to write down processes I took to fix my client systems and network. Now I write some tips and tricks to help others with the tech issues that one might encounter.

Leave a Reply

Comment

Comment Text*

Name*

Email*

Post Comment

