

Torrent Client/Server

Headless Torrent Server/Client: Built on a Raspberry Pi

In this tutorial I will show you how to build a headless torrent server and client. I will be using the very popular Transmission as my torrent server and client. The first step is to install the necessary software:

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

After installing Transmission we need to stop the service and edit a few settings. Here is a list of some of the important settings. I have also annotated a copy of my settings.json file below.

- "download-dir" - This is the folder where you want to save your completed downloads (/mnt/torrents/complete)
- "incomplete-dir" - This is the folder where your partial downloads are saved until complete (/mnt/torrents/incomplete).
- "incomplete-dir-enabled" - Set to 'true' to enable the above 'incomplete' folder
- "rpc-whitelist" - Include the IP ranges you want to access the transmission web interface
- "rpc-whitelist-enabled" - Set this to 'true' to enforce the above whitelist.

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

```
{
"alt-speed-down": 50,
"alt-speed-enabled": false,
"alt-speed-time-begin": 540,
"alt-speed-time-day": 127,
"alt-speed-time-enabled": false,
"alt-speed-time-end": 1020,
"alt-speed-up": 50,
"bind-address-ipv4": "0.0.0.0",
"bind-address-ipv6": "::",
"blocklist-enabled": false,
"blocklist-url": "http://www.example.com/blocklist",
"cache-size-mb": 4,
"dht-enabled": true,
"download-dir": "/mnt/torrents/complete", (This is where your completed
downloads will be saved)
"download-limit": 100,
"download-limit-enabled": 0,
"download-queue-enabled": true,
"download-queue-size": 5,
"encryption": 1,
"idle-seeding-limit": 30,
"idle-seeding-limit-enabled": false,
"incomplete-dir": "/mnt/torrents/incomplete", (This is where your incomplete
```

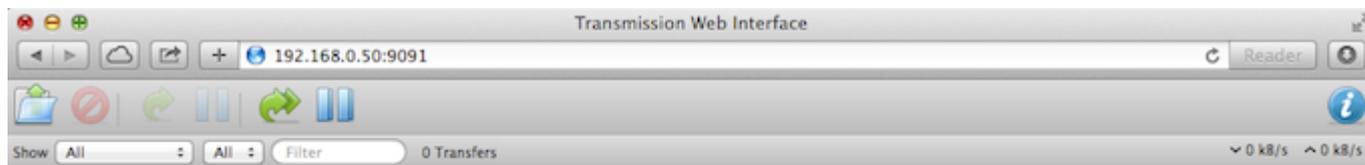
```
downloads will be saved)
"incomplete-dir-enabled": true,
"lpd-enabled": false,
"max-peers-global": 200,
"message-level": 2,
"peer-congestion-algorithm": "",
"peer-id-ttl-hours": 6,
"peer-limit-global": 200,
"peer-limit-per-torrent": 50,
"peer-port": 51413,
"peer-port-random-high": 65535,
"peer-port-random-low": 49152,
"peer-port-random-on-start": false,
"peer-socket-tos": "default",
"pex-enabled": true,
"port-forwarding-enabled": false,
"preallocation": 1,
"prefetch-enabled": 1,
"queue-stalled-enabled": true,
"queue-stalled-minutes": 30,
"ratio-limit": 2,
"ratio-limit-enabled": false,
"rename-partial-files": true,
"rpc-authentication-required": true,
"rpc-bind-address": "0.0.0.0",
"rpc-enabled": true,
"rpc-password": "Put your desired password here(will be encrypted after
transmission is started)",
"rpc-port": 9091,
"rpc-url": "/transmission/",
"rpc-username": "Put your desired username here",
"rpc-whitelist": "127.0.0.1, 192.168.0.*", (this allows everything on my
192.168.0.0/24 network to access the transmission web interface)
"rpc-whitelist-enabled": true,
"scrape-paused-torrents-enabled": true,
"script-torrent-done-enabled": false,
"script-torrent-done-filename": "",
"seed-queue-enabled": false,
"seed-queue-size": 10,
"speed-limit-down": 100,
"speed-limit-down-enabled": false,
"speed-limit-up": 100,
"speed-limit-up-enabled": false,
"start-added-torrents": true,
"trash-original-torrent-files": false,
"umask": 18,
"upload-limit": 100,
"upload-limit-enabled": 0,
"upload-slots-per-torrent": 14,
"utp-enabled": true
}
```

All that is left to do is start the transmission-daemon and test the transmission web interface

```
$ sudo service transmission-daemon start
```

To go to the transmission web interface just go to your web browser input your url:

<http://serverip:9091> It should look something like this:



From:

<http://it-joe.com/> - **iT-Joe**

Permanent link:

<http://it-joe.com/linux/torrent>

Last update: **2018/04/01 03:11**

