

# Seedbox

- [How to Install qBittorrent on Ubuntu 18.04 Desktop or Server](#)

# How to Install qBittorrent on Ubuntu 18.04 Desktop or Server

This tutorial will be showing you how to install qBittorrent on Ubuntu 18.04. qBittorrent is a free, open-source, fast and lightweight P2P BitTorrent client written in C++ / Qt, available for Linux, FreeBSD, MacOS and Windows. It aims to provide an open-source alternative to [uTorrent](#), which is a very popular BitTorrent client among Windows users.

qBittorrent features:

- A uTorrent-like interface
- Supports main BitTorrent extensions like DHT, peer exchange, full encryption.
- It provides a good integrated search engine
- It also comes with UPnP port forwarding / NAT-PMP
- You can also remotely control it via the qBittorrent Web interface.

## Install qBittorrent on Ubuntu 18.04 Desktop via PPA

qBittorrent is included in the default Ubuntu software repository. You can easily install it on Ubuntu 18.04 desktop by running the following command in a terminal window.

```
sudo apt install qbittorrent
```

However, it is outdated (v4.0.3). The latest version 4.1.6 was released on May 6, 2019. To get the latest version of qBittorrent, you need to install it from the official qBittorrent PPA (Personal Package Archive). First, run the following command to add the PPA.

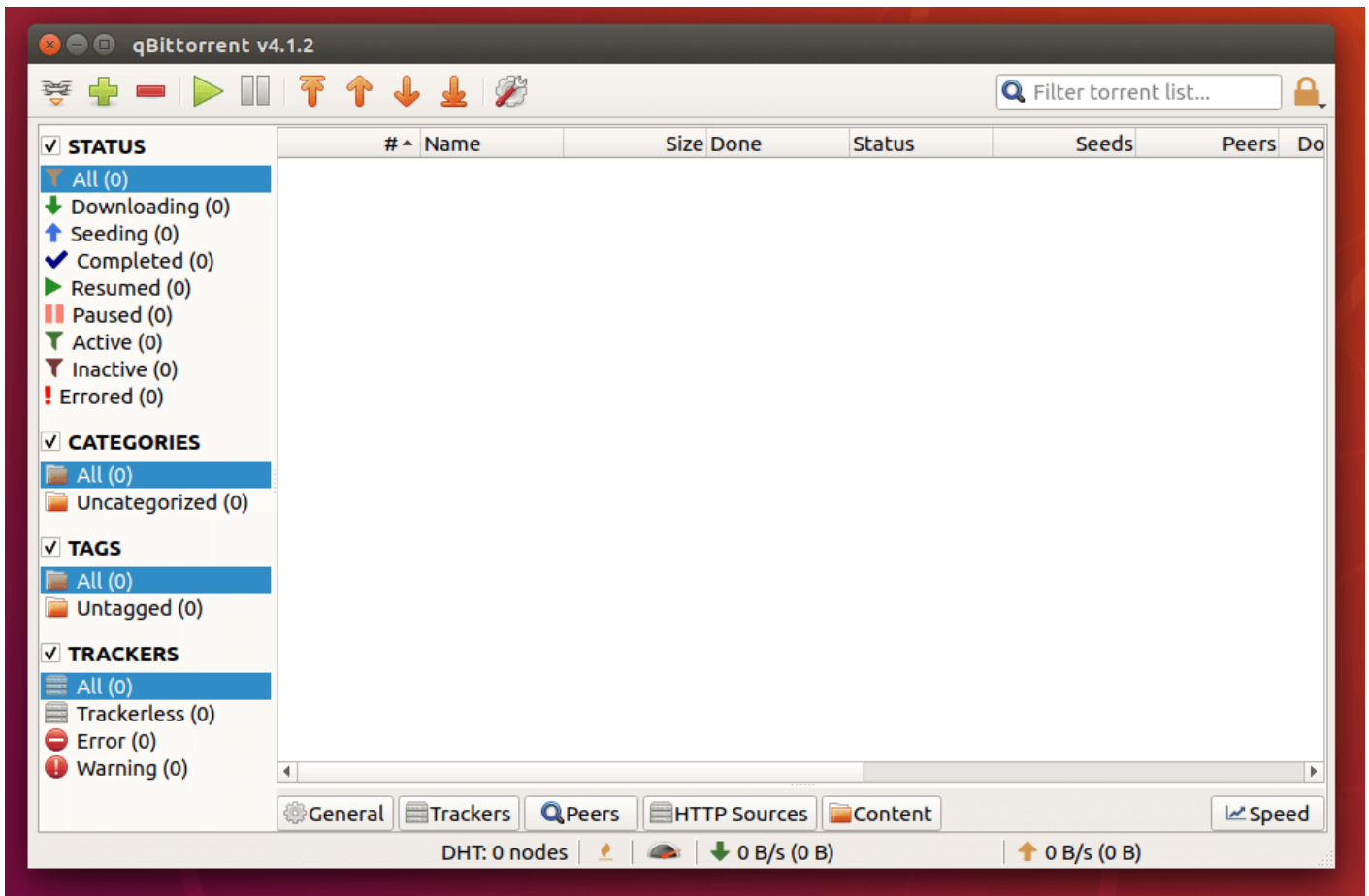
```
sudo add-apt-repository ppa:qbittorrent-team/qbittorrent-stable
```

```
linuxbabe@bionic: ~  
linuxbabe@bionic:~$ sudo add-apt-repository ppa:qbittorrent-team/qbittorrent-stable  
Packages for the stable series of qBittorrent  
More info: https://launchpad.net/~qbittorrent-team/+archive/ubuntu/qbittorrent-stable  
Press [ENTER] to continue or Ctrl-c to cancel adding it.  
[ENTER]
```

Press Enter to continue. It will update the software repository index. After that, run the following command to install the latest version of qBittorrent.

```
sudo apt install qbittorrent
```

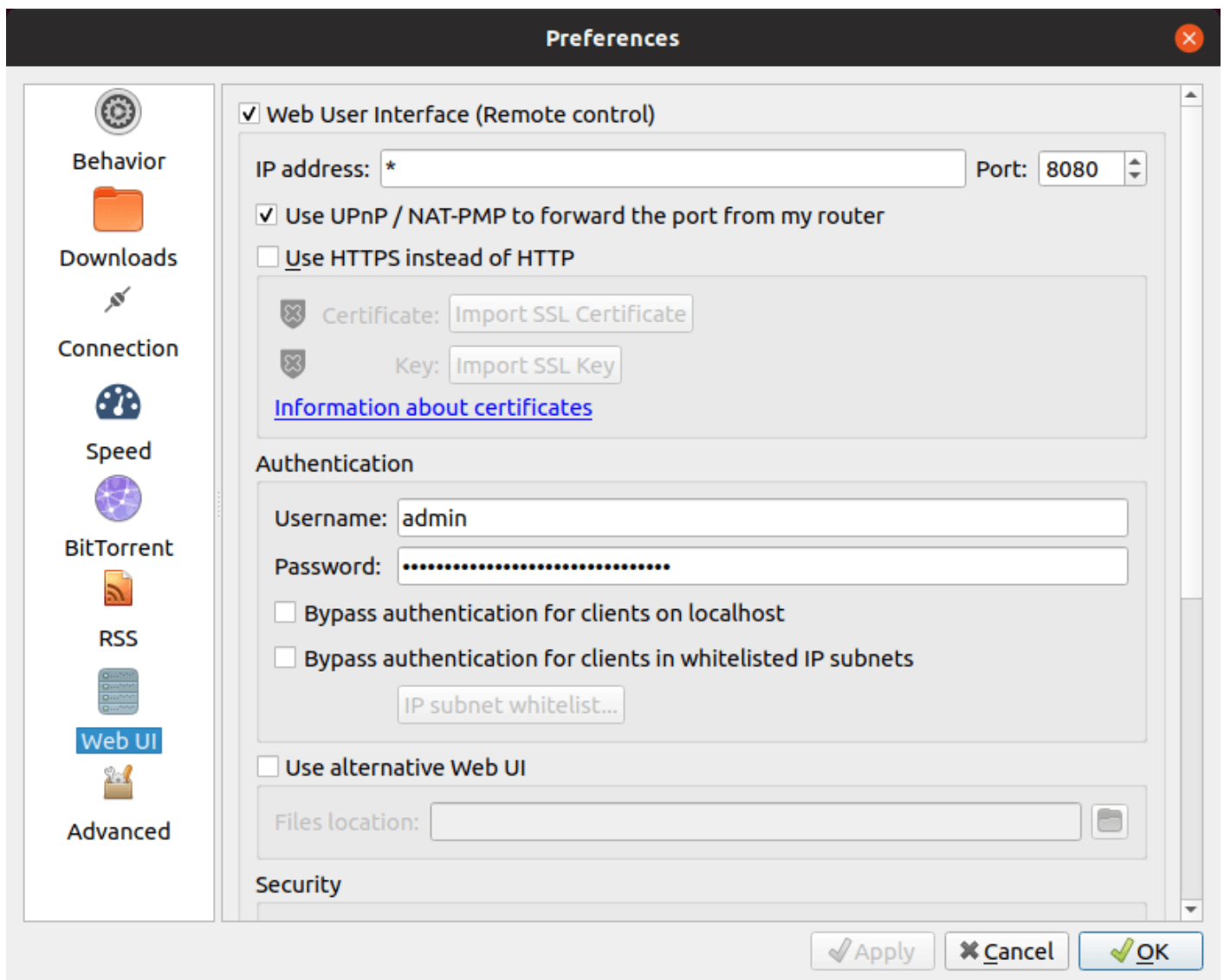
This PPA also works on other Linux distributions that are based on Ubuntu, such as Linux Mint and Elementary OS. If your system already has qBittorrent installed, then the above commands will update qBittorrent to the latest version. Don't worry, your existing torrents are safe. Once installed, you can start qBittorrent from application menu.



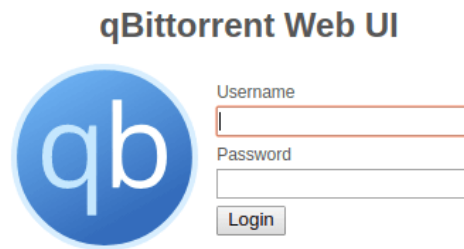
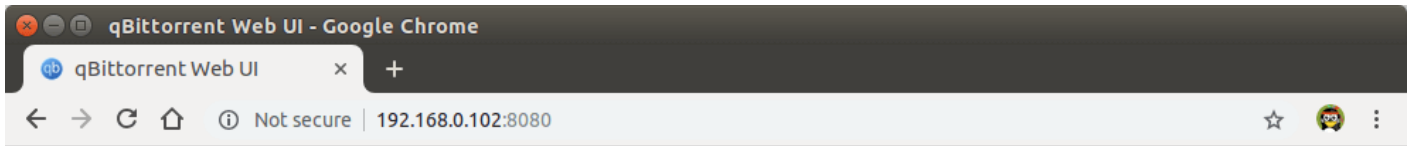
# How to Enable qBittorrent Web UI on Ubuntu 18.04 Desktop

The web UI allows you to remotely control qBittorrent in a web browser.

1. In the qBittorrent application window, go to **tools** > **Preferences** > **Web UI**.
2. Tick on the **Web User Interface (Remote Control)** option.
3. You can also choose to use UPnP/NAT-PMP to forward the port from your router, so that you will be able access qBittorrent from outside network.
4. By default, the web UI is accessible on port 8080. If there's another service using port 8080, you need to change the port number in qBittorrent.
5. Set a username and password. (By default "admin" and "adminadmin" respectively).
6. At the bottom, you can also choose to use one of two dynamic DNS services: DynDNS and No-IP.
7. Click **Apply** button.



Now in my local network, I can access qBittorrent web interface via **192.168.0.102:8080**.

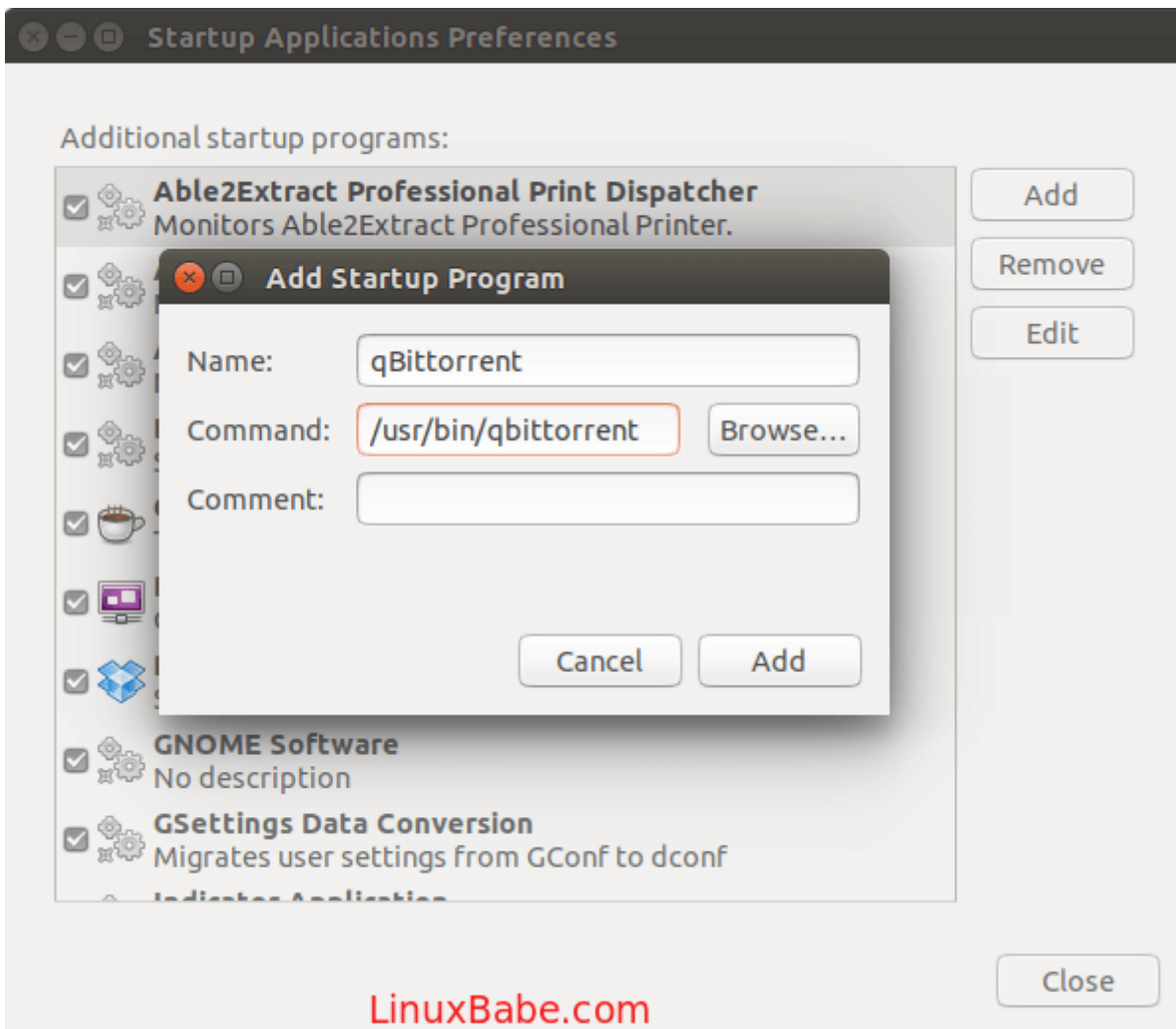


# Make qBittorrent Automatically Start at System Boot Time

To enable autostart, open the `Startup Applications` from your applications menu. It can also be launched by issuing the following command in the terminal window.

```
gnome-session-properties
```

Then click `Add` button to add a new startup program. In the Name field, you can enter something like "qBittorrent". In the Command field, enter `/usr/bin/qbittorrent`. You can leave the comment field blank. After that, click `Add` button.



# How to Install qBittorrent on Ubuntu 18.04 Server

You can install qBittorrent command line client on a headless Ubuntu 18.04 server and manage it via the qBittorrent web interface (You control it in a web browser). SSH into your Ubuntu 18.04 server and use the same PPA to install qBittorrent daemon.

```
sudo add-apt-repository ppa:qbittorrent-team/qbittorrent-stable
```

```
sudo apt install qbittorrent-nox
```

Note that we need to install `qbittorrent-nox` (without X), instead of `qbittorrent`. qBittorrent-nox is meant to be controlled via its feature-rich Web UI which is accessible as a default on `http://localhost:8080`. The Web UI access is secured and the default account username is “`admin`” with “`adminadmin`” as a password. You can start qBittorrent-nox with:

```
qbittorrent-nox
```

However, starting qBittorrent-nox this way isn't recommended, because you can't run other command while it's running. Press `Ctrl+C` to quit it now. We can create a systemd service unit so it can run in the background and also start at system boot time.

Create the `qbittorrent-nox` user and group so that it can run as an unprivileged user, which will increase your server's security.

```
sudo adduser --system --group qbittorrent-nox
```

The `--system` flag means we are creating a system user instead of normal user. A system user doesn't have password and can't login, which is what you would want for a torrent client. A home directory `/home/qbittorrent-nox` will be created for this user. You might want to add your user account to group `qbittorrent-nox` with the following command so that the user account has access to the files downloaded by qBittorrent-nox. Files are downloaded to `/home/qbittorrent-nox/Downloads/` by default. Note that you need to re-login for the groups change to take effect.

```
sudo adduser your-username qbittorrent-nox
```

Then create a systemd service file for qbittorrent-nox with your favourite text editor such as nano.

```
sudo nano /etc/systemd/system/qbittorrent-nox.service
```

Copy and paste the following lines into the file. If there's another service using port 8080, then you need to change the port number for qBittorrent to something else like 8081. Also note that the `-d` (daemonize) option is needed in this systemd service unit.

```
[Unit]
Description=qBittorrent Command Line Client
After=network.target

[Service]
#Do not change to "simple"
Type=forking
User=qbittorrent-nox
Group=qbittorrent-nox
UMask=007
ExecStart=/usr/bin/qbittorrent-nox -d --webui-port=8080
Restart=on-failure

[Install]
WantedBy=multi-user.target
```

To save a file in Nano text editor, press `Ctrl+O`, then press `Enter` to confirm. Press `Ctrl+X` to exit. Now start qBittorrent-nox with the following command.

```
sudo systemctl start qbittorrent-nox
```

Note that if you change a systemd service file, you need to reload the systemd daemon for the change to take effect.

```
sudo systemctl daemon-reload
```

You might also want to enable auto start at system boot time.

```
sudo systemctl enable qbittorrent-nox
```

Check its status:

```
systemctl status qbittorrent-nox
```

You can see that qBittorrent-nox is running and auto start at boot time is enabled.

# Accessing qBittorrent Web UI

To access the qBittorrent Web UI from local network, enter the Ubuntu server's private IP address followed by the port number like below.

```
192.168.0.102:8080
```

Username is admin. Default password is "adminadmin".

The default web interface.

qbittorrent as a service ubuntu

It's strongly recommended to change the default username and password. Go to **Tools** > **Options** and select the Web UI tab. Under the Authentication section, change both username and password.

qbittorrent-nox config file

And now you can start downloading torrents on your Ubuntu 18.04 server. You have the option to upload local torrents or add magnet links. Files are downloaded to `/home/qbittorrent-nox/Downloads/` by default.



# Using External Hard Drive as the Download Destination Folder

You can change the default download destination folder in `Tools` -> `Options` -> `Downloads`. In order to download torrent, the `qbittorrent-nox` user needs to have write permission to the download destination folder. If your external hard drive is mounted at `/external-hdd`, then run the following command grant the `qbittorrent-nox` user write permission to the download destination folder.

```
sudo setfacl -R -m "u:qbittorrent-nox:rwX" /external-hdd
```

In my opinion, using `setfacl` (set file access control list) rather than `chown` is a more flexible and better approach to manager permissions. `setfacl` will not change the ownership of a folder, it just grant permissions to additional users.

You may need to delete your torrent and re-add it after your run the `setfacl` command.

## Accessing qBittorrent Web Interface Outside of Your Home Network

If you want to access qBittorrent Web UI remotely from your mobile phone or public Wi-fi, then you need to set up port forwarding in your router. Also I recommend setting up Nginx reverse proxy and enable HTTPS protocol to encrypt the communication.

## Set Up Nginx Reverse Proxy for qBittorrent WebUI

A [reverse proxy](#) is a proxy for another web server, in this case the qBittorrent WebUI. Setting up Nginx reverse proxy makes it easy to enable HTTPS protocol. Install Nginx on Ubuntu 18.04.

```
sudo apt install nginx
```

Start Nginx.

```
sudo systemctl start nginx
```

Then create a Nginx server block file for qBittorrent WebUI.

```
sudo nano /etc/nginx/conf.d/qbittorrent-webui.conf
```

Copy and paste the following texts into the file. Replace the red-colored text with your own domain name and the IP address of your Ubuntu server. You should also set the A record for your domain name. If you don't have a domain name yet, then I recommend buying domain name from [Namecheap](#) because their price is lower than GoDaddy and they give you whois privacy protection free of charge.

```
server {
    listen 80;
    server_name torrent.your-domain.com;

    access_log /var/log/nginx/torrent.your-domain.com.access;
    error_log /var/log/nginx/torrent.your-domain.com.error;

    location / {
        proxy_pass          http://your-server-ip:8080/;
        proxy_set_header    X-Forwarded-Host    $server_name:$server_port;
        proxy_hide_header    Referer;
        proxy_hide_header    Origin;
        proxy_set_header    Referer            "";
        proxy_set_header    Origin             "";
        add_header           X-Frame-Options    "SAMEORIGIN";
    }
}
```

## MA CONFIG AVEC RUNCLOUD

```
location ~ ^/(?<url>.*) {
    proxy_pass http://195.154.235.32:8080/$url;
    proxy_hide_header    Referer;
    proxy_hide_header    Origin;
    proxy_set_header    Referer            "";
    proxy_set_header    Origin             "";
    proxy_set_header    Host                $host;
    proxy_set_header    X-Forwarded-Host    $server_name:$server_port;
    proxy_set_header    X-Real-IP          $remote_addr;
    proxy_set_header    X-Forwarded-For    $proxy_add_x_forwarded_for;
    proxy_set_header    X-Forwarded-Proto  $scheme;
```

```
add_header      Front-End-Https      on;

}
```

```
sudo nginx -t
```

If the test is successful, reload Nginx

```
sudo systemctl reload nginx
```

Now in your router, forward HTTP request (port 80) to the IP address of your Ubuntu server. After that, you can access qBittorrent WebUI via your domain name ( `torrent.your-domain.com` ).

**Note:** If your ISP gives you a NAT-ed IP address, i.e. not a public IP address, then port forwarding won't work. Instead, you need to [use PageKite to expose local web server to the Internet](#).

# Enable HTTPS to Encrypt Communications

To secure the Web UI, you can install a free TLS certificate issued by Let's Encrypt. First you need to install the Let's Encrypt client (certbot) on Ubuntu 18.04 server.

```
sudo apt install software-properties-common

sudo add-apt-repository ppa:certbot/certbot

sudo apt install certbot python3-certbot-nginx
```

`Python3-certbot-nginx` is the Certbot Nginx plugin. After they are installed, run the following command to automatically obtain and install Let's Encrypt certificate.

```
sudo certbot --nginx --redirect --agree-tos --hsts --staple-ocsp --email your-email-address -d torrent.your-domain.com
```

Once that's done, refresh the qBittorrent Web UI. It will be automatically redirected to HTTPS connection.

# Troubleshooting A Problem in Version 4.1.2

There's a problem in the old version 4.1.2. I write it down here just for note-taking purpose.

By default, qBittorrent-nox 4.1.2 listens on TCP port 8080 of 0.0.0.42, as can be seen with the following command. Port 8999 is used to connect to peers.

```
sudo netstat -lnt | grep qbittorrent
```

This IP address (0.0.0.42) is very strange and prevents me from accessing the web interface. If I run the following command,

```
curl http://localhost:8080
```

I get this error.

```
curl: (7) Failed to connect to localhost port 8080: Connection refused
```

I don't know why 0.0.0.42 is set as a default. To access the web interface, I created the qBittorrent configuration file.

```
sudo nano /home/qbittorrent-nox/.config/qBittorrent/qBittorrent.conf
```

Copy and paste the following text into the file. Change 12.34.56.78 to the IP address of your Ubuntu server.

```
[LegalNotice]
Accepted=true

[Network]
Cookies=@Invalid()

[Preferences]
WebUI\Address=12.34.56.78
```

Save and close the file. Then restart qBittorrent-nox.

```
sudo systemctl restart qbittorrent-nox
```

# Wrapping Up

I hope this tutorial helped you install qBittorrent on Ubuntu 18.04 desktop or server. Take care.

Rate this tutorial

[Total: 9 Average: 4.9]