Diesel's Installing Profit Trailer on a VPS

This is a guide to installing and running Profit Trailer on a Ubuntu VPS. It utilizes NodeJS, NPM, JAVA jdk and PM2. The basic installation should work on any system, with minor deviations - this is Ubuntu, installations on Windows, OSX and RaspPI will be different, but the applications are the same.

If you haven't gotten a license for Profit Trailer yet or for more info, contact me on Telegram: @Another_Diesel

Get a VPS.

I recommend Westhost.com, but there are many good ones. It doesn't need to be too heft, I use an $8/month system with 4ghz cpu, 1024gb ram and 3TB data transfer/month.
Get your VPS with Ubuntu. If you are comfortable with other flavors of Linux, or want to pay extra for Windows, fine. This tutorial is designed for Ubuntu, but the differences are minimal

Connect to your new VPS

I use WinSCP to connect to my VPS. It runs on Windows. You can use anything you are comfortable with, but WinSCP allows me to connect to the VPS, FTPp files, edit files and has PuTTY integrated into the app. So I can move files, edit files and also connect to the terminal in one nice little free app. https://winscp.net/eng/index.php

Your provider will have given you access information: an IP Address, username and password. In my case, Westhost gave me root user, which is fine. Use this info to open a terminal window to your new VPS. In my case I open Winscp and set up a new connection:
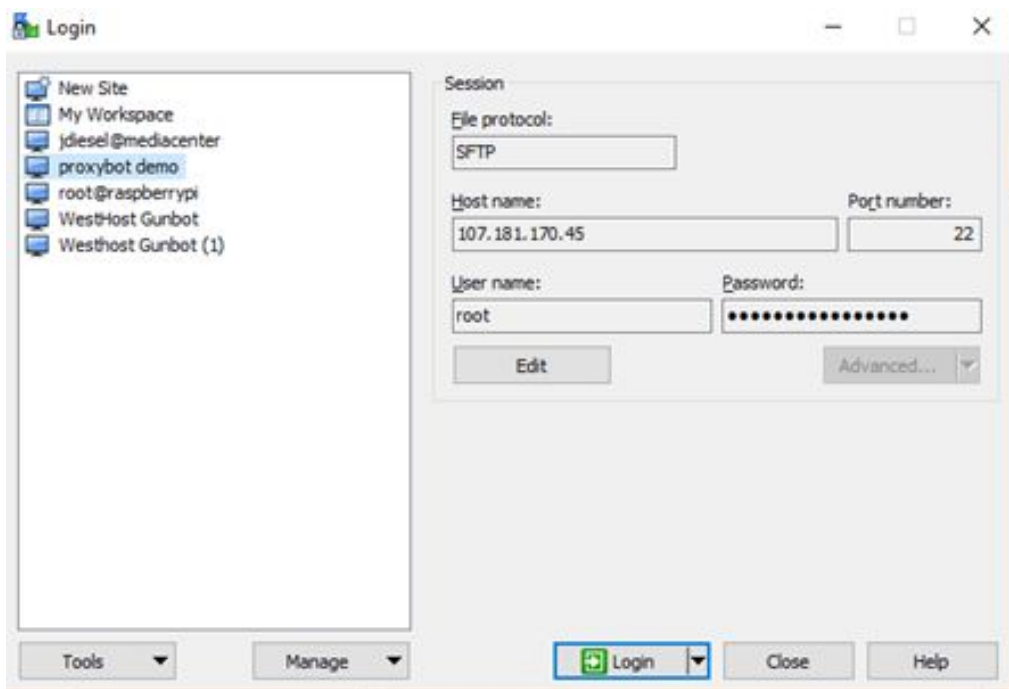
In WinSCP, I click the terminal icon in the menu bar to get to a terminal:
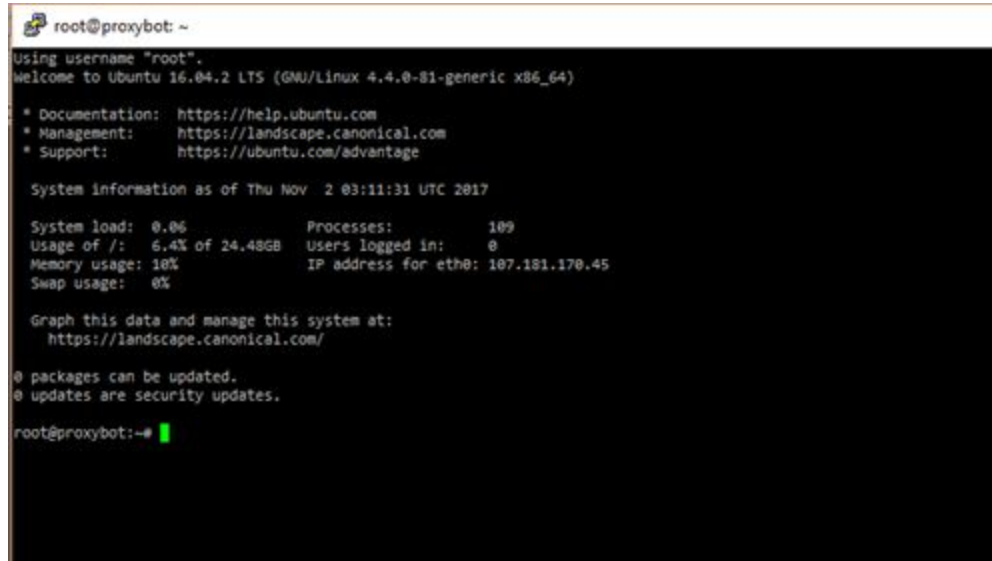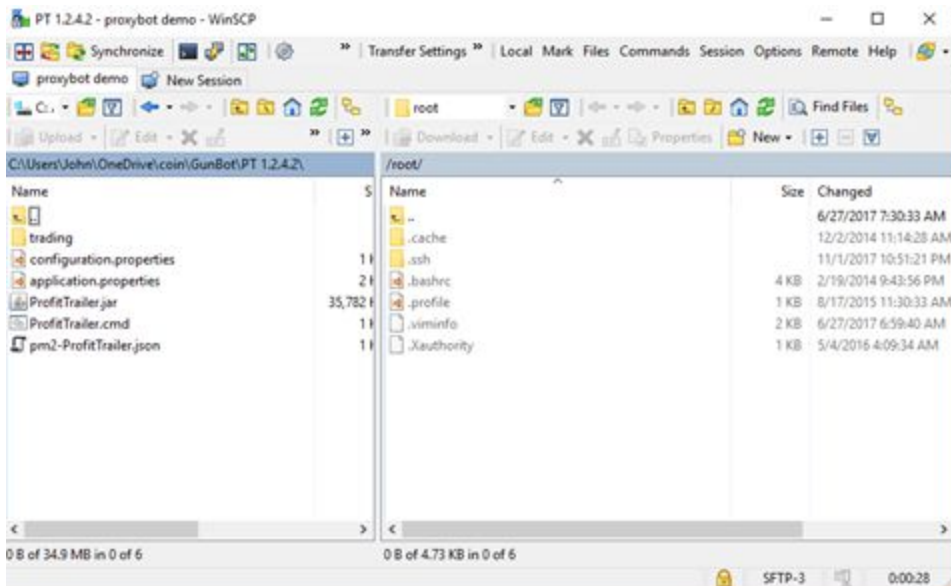
I like to drop my bots into the /var/opt folder:

From the main root login,

```
cd ..
cd /var/opt
mkdir pt
cd pt
```

You should now be inside a folder called pt in your /var/opt path
The prompt will look similar to this:

```
root@proxybot:/var/opt/pt#
```

run an update: `sudo apt-get update`
install java: `sudo apt-get install default-jdk`
install nodejs: `sudo apt-get install nodejs`
install the node fix: `ln -s /usr/bin/nodejs /usr/bin/node`
install npm: `sudo apt-get install npm`
install pm2: `npm install pm2@latest -g`

pm2 is a process manager – it lets you run the bot in the background, and has features to autostart, log-rotation and much more. You do not HAVE to use pm2, but I like it and rely on it, especially from running multiple bots.
http://pm2.keymetrics.io/docs/usage/quick-start/

At this point you have all the apps and tools you need to run the Profit Trailer.
Java to run the jar
Nodejs to run pm2
Pm2 to manage your bot process, log windows and logs

Type `pm2` at your prompt and you should get a list of pm2 commands. This will verify everything is running as expected.

DOWNLOAD PROFIT TRAILER

I like to download the bot to my local machine, edit the configs and then upload everything to the VPS at one time. I'll show you how I do that. If you are familiar with Linux, you can certainly download and unzip the bot directly to your working folder.

Download the Profit Trailer from https://github.com/taniman/profit-trailer/releases
I unzip the file into folder named with the version number (PT 1.2.4.2)

You should have something similar to

Name	Status	Date modified	Type	Si
trading	⊘	10/27/2017 10:04 ...	File folder	
application.properties	⊘	10/27/2017 10:04 ...	Properties Source ...	
configuration.properties	⊘	10/27/2017 10:04 ...	Properties Source ...	
pm2-ProfitTrailer.json	⊘	10/27/2017 10:04 ...	JSON File	
ProfitTrailer.cmd	⊘	10/27/2017 10:04 ...	Windows Comma...	
ProfitTrailer.jar	⊘	10/27/2017 10:04 ...	Executable Jar File	

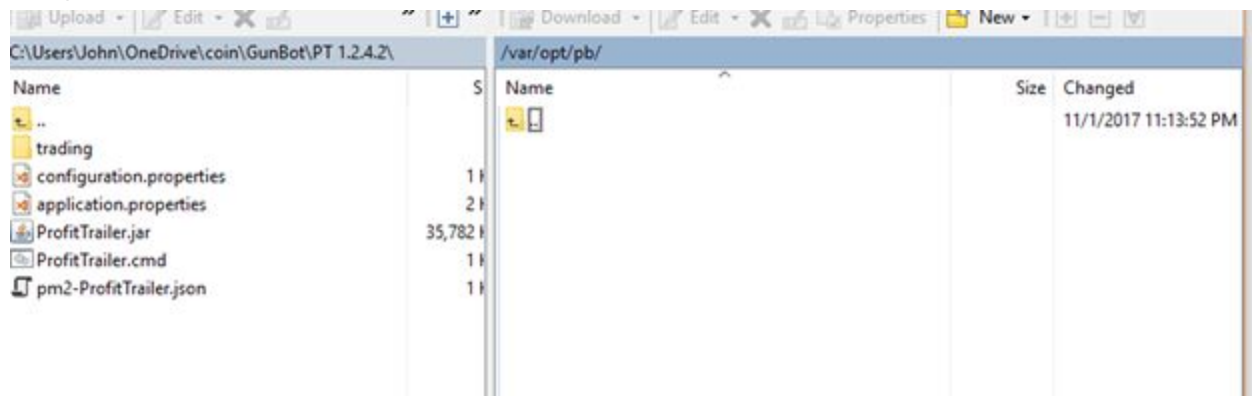Open WinSCP, and connect to your VPS.

Select the "Open directory/bookmark" icon and type in `/var/opt/pt`

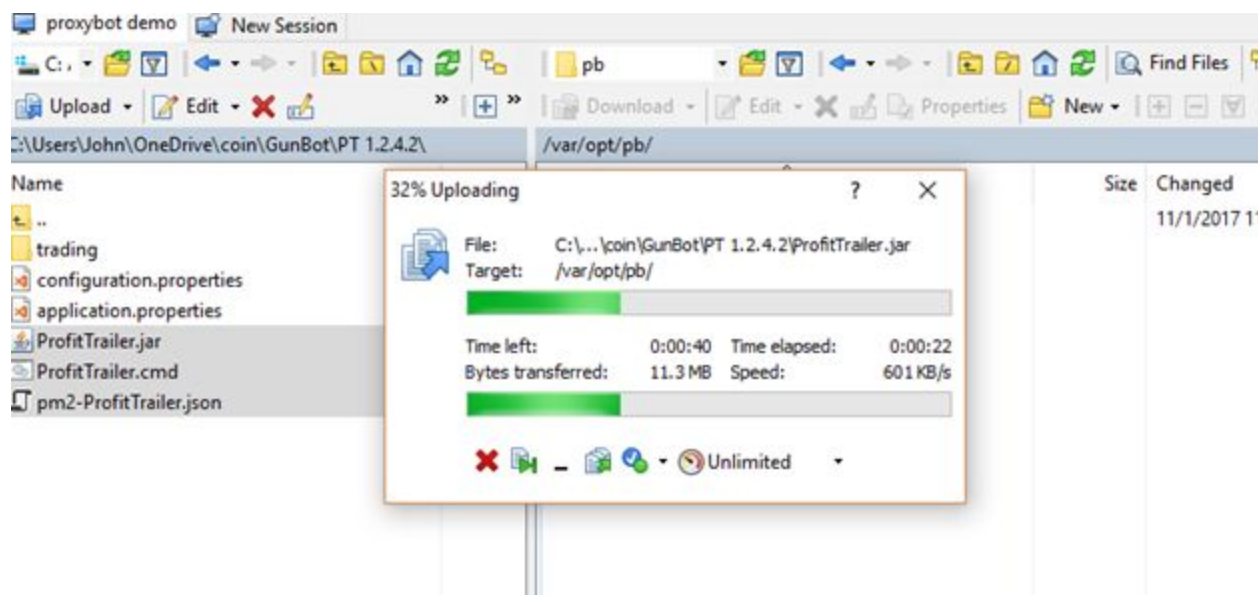This should bring you to your pt folder
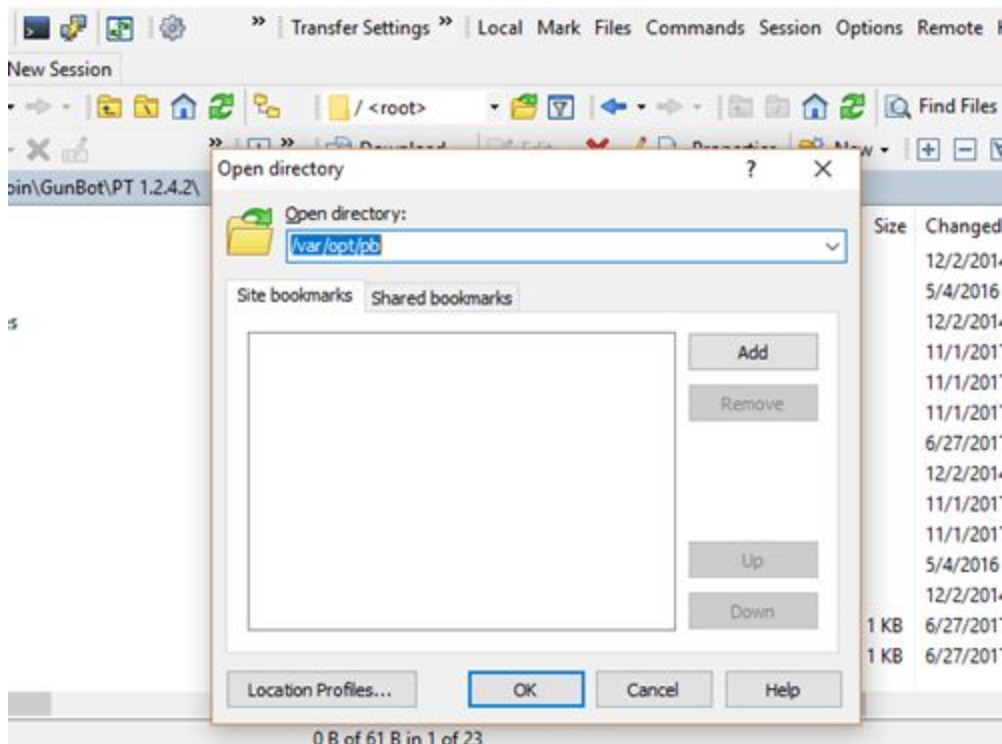
In the left hand window of WinSCP, find your new PT folder, the one you just unzipped from the download.

Things should look like this

Simply select and drag all of the files from the left window to the right window.

You now have the PT files on your VPS, and are ready to start editing configs.
CONFIGURING YOUR BOT IS NOT IN THE SCOPE OF THIS TUTORIAL.
You can double-click on a file in WinSCP to edit that file (left or right windows). The bot will be using the files on the right, in the /var/opt/pt folder.

At this point, you can minimize WinSCP and go back to your terminal window (or click on the terminal icon to connect again)

RUNNING YOUR BOT

Now for the fun part. To run your bot, you first navigate to your pt folder
        `cd /var/opt/pt`
Start the bot using pm2:
        `pm2 start pm2-ProfitTrailer.json`
At this point, the bot is running. To see the live log scrolling by, type `pm2 log 0`. You use CTRL-C to get out of the window.

`pm2 log 0`	See log live
`pm2 stop 0`	Stop the bot
`pm2 reload 0`	reload and restart the bot

Text logs are maintained by pm2 and are located at /root/.pm2/ (for my root account access)

Finally, run a save and startup command. Ater this is complete, PM2 will restart your bot if you restart the server. Run this command again if you change your app, or add a second bot.
        `pm2 save`
        `pm2 startup`

ADDING A SECOND BOT

Time to go bigger? RUnning a second bot is fairly straightforward.

Make a second folder in your /var/opt/ folder, maybe pt2
/var/opt/pt2
Upload a copy of your Profit Bot to this folder.
Edit the application.properties file and change the two port numbers - I increment them by 1
(If these variable do not exist, add them)

/var/opt/pt/application.properties
        server.port = 8443
        server.additionalPort =8083

/var/op/pt2/application.properties
        server.port = 8444
        server.additionalPort =8084

RUNNING THAT SECOND BOT IN PM2

You will need to edit /var/opt/pt/pm2-start.json in order to change the Application name. Pm2 requires unique application names. I like to edit all of my start jsons to be descriptive of the bot:

```
{
  "apps": [
    {
      "name": "PT_Polo",
      "cwd": ".",
      "script": "java",
      "args": [
        "-XX:+UseConcMarkSweepGC",
        "-Xmx512m",
        "-Xms512m",
        "-jar",
        "ProfitTrailer.jar"
      ],
      "env": {
        "NODE_TLS_REJECT_UNAUTHORIZED": 0,
        "SPRING_PROFILES_ACTIVE": "prod"
      },
      "node_args": [],
      "log_date_format": "YYYY-MM-DD HH:mm Z",
      "exec_interpreter": "",
      "exec_mode": "fork"
    }
  ]
}
```

Just start the second bot the same way you started the first:
        pm2 start pm2-start.json
This will display PT Polo as the app name in the pm2 list.

App name	id	mode	pid	status	restart	uptime	cpu	mem	user	watching
PT Polo	13	fork	8824	online	0	7D	0%	637.6 MB	root	disabled
PT Trex	12	fork	10963	online	2	5D	0%	349.6 MB	root	disabled
PT USD	14	fork	8908	online	0	7D	0%	568.7 MB	root	disabled

Module activated

Module	version	target PID	status	restart	cpu	memory	user
pm2-logrotate	2.2.0	N/A	online	0	0%	96.953 MB	root

USING NON BTC MARKETS

Running 2 bots probably means you are running a bot against 2 markets.
Please use a second exchange account for this, if you use the same account, it becomes extremely easy for the bot to make trades you may not agree with.

In your application.properties, you will need to help the bot define your market keys.
I have all of my bots defined with the market name in the key, even though BTC works without this.. Remember, the default key is your licensed key.

default_BTC_apiKey =
default_BTC_apiSecret =
BTC_apiKey1 =
BTC_apiSecret1 =
What's this? Yes, you can assign even more (unlicensed) keys to help the bot along. I have no idea if this is even necessary anymore, but it IS available.
BTC_apiKey2 =
BTC_apiSecret2 =

default_ETH_apiKey =
default_ETH_apiSecret =
ETH_apiKey1 =
ETH_apiSecret1 =

default_USDT_apiKey =
default_USDT_apiSecret =
USDT_apiKey1 =
USDT_apiSecret1 =