**SETTING UP AND INSTALLING NODE**

1. Create a node directory under the /opt directory.  This is where we’re going to store our version of node:

sudo mkdir /opt/node

1. Go to <http://nodejs.org/dist/> and find the latest distro with a linux arm version.  At the time of this writing, 11.3 was available.  The file you are looking for will be named something like node-v0.11.3-linux-arm-pi.tar.gz but it will have the words “linux” and “arm” in the name.
2. Right-click on the .gz file and copy the full path.
3. SSH into your Raspberry Pi
4. Use wget to pull the path you just copied:

sudo wget http://nodejs.org/dist/v0.11.3/node-v0.11.3-linux-arm-pi.tar.gz

1. Use tar to unstuff the .gz file you just downloaded:

tar xvzf node-v0.11.3-linux-arm-pi.tar.gz

1. Copy the contents of the unstuffed directory to the /opt/node directory we created earlier:

sudo cp -r node-v0.11.3-linux-arm-pi/\* /opt/node

C:\Users\qiany\AppData\Roaming\Tencent\Users\1310211742\QQ\WinTemp\RichOle\9IHBQ_N_$@T6@)_RP}`A{JQ.png

1. This next part is where Matt and I deviate.  I’m a fan of making things easy.  Had we been installing NodeJS on a desktop or laptop, we would be able to use node from the command line simply by typing “node [COMMAND]”.  This was how I set that up.  We’re going to create a symbolic link to both node and npm and put those links in /usr/local/bin:

sudo ln -s /opt/node/bin/node /usr/local/bin/node

sudo ln -s /opt/node/bin/npm /usr/local/bin/npm

C:\Users\qiany\AppData\Roaming\Tencent\Users\1310211742\QQ\WinTemp\RichOle\J[D801DF@FM_}L)5C@QZ}ZV.png

1. Now restart the .bashrc as follows:

source ~/.bashrc

C:\Users\qiany\AppData\Roaming\Tencent\Users\1310211742\QQ\WinTemp\RichOle\8DW$U3$6GBT3`15WG9%GM%2.png

1. Now that .bashrc has been restarted, let’s verify that not only did our symbolic link work, but we have node and npm properly installed.  First type node -v to see the installed version of node and then type npm -v to see the installed version of npm.  You should see something like the following:

pi@raspberrypi ~ $ node –v

v0.11.3

pi@raspberrypi ~ $ npm –v

1.2.25

C:\Users\qiany\AppData\Roaming\Tencent\Users\1310211742\QQ\WinTemp\RichOle\JCPY8E{YS8LD)SKUMH@8_59.png

**INSTALLING EXPRESS**

*Please note that this next section is completely optional and as of the latest RPi OS, seem to not work as listed here.  Please use with caution.*

1. To install Express, simply enter the following on the command line:

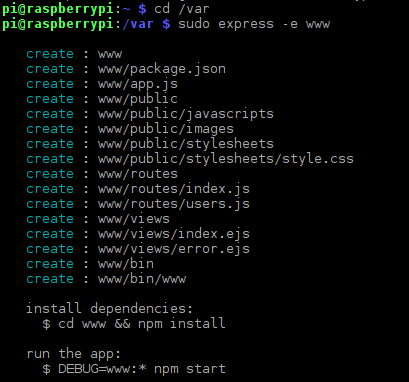
sudo npm install -g express

You may see some warnings, that’s OK for now. If you want to dig in further, you can review them and fix them at your leisure.

1. With Express properly installed, it’s now time to create our working directory.  However, we’re going to let Express do that for us.  The working directory that we’re going to create will be /var/www though depending upon your setup, the directory www may or may not exist yet.  For the purposes of this tutorial, we are assuming it does not yet exist.  Simply type the following into the command line:

cd /var

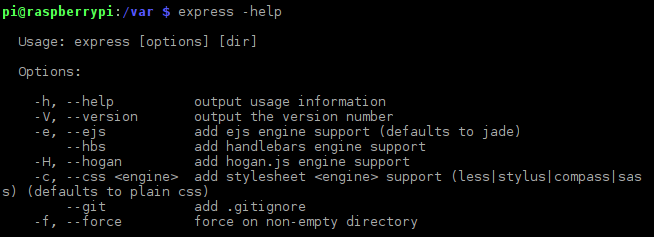
sudo express -e www



1. There are several flags you can set and depending upon what you prefer, many combinations.  To see all of them, type express –help in the command line.  You should see something like the following:

express –help

Usage: express [options] Options: -h, --help output usage information -V, --version output the version number -s, --sessions add session support -e, --ejs add ejs engine support (defaults to jade) -J, --jshtml add jshtml engine support (defaults to jade) -H, --hogan add hogan.js engine support -c, --css add stylesheet support (less|stylus) (defaults to plain css) -f, --force force on non-empty directory



1. Once Express successfully creates the necessary directory structure, the last step before we turn the sucker on is to jump into the www directory and let npm add the finishing touches.  We do this as so (please note that I’ve chosen to add sudo to the command which deviates from what Express shows.  Without sudo permissions, npm will not be able to install all the dependencies properly:

cd www && sudo npm install

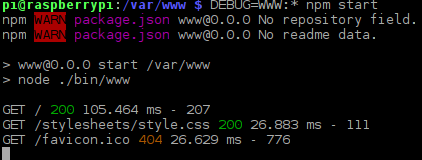
Again, you should see some warnings which you are welcome to go back and correct but they will not impact this process.

1. Once npm is done, it’s finally time to flip the switch and see if we made Frankenstein or a flub:

node app.js

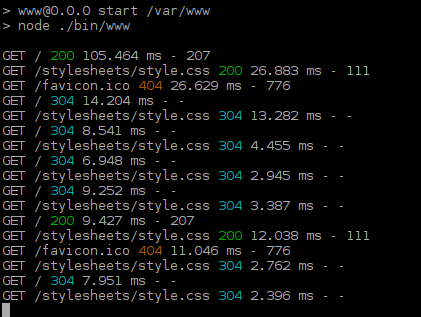
此处修改为：

DEBUG=WWW:\* npm start

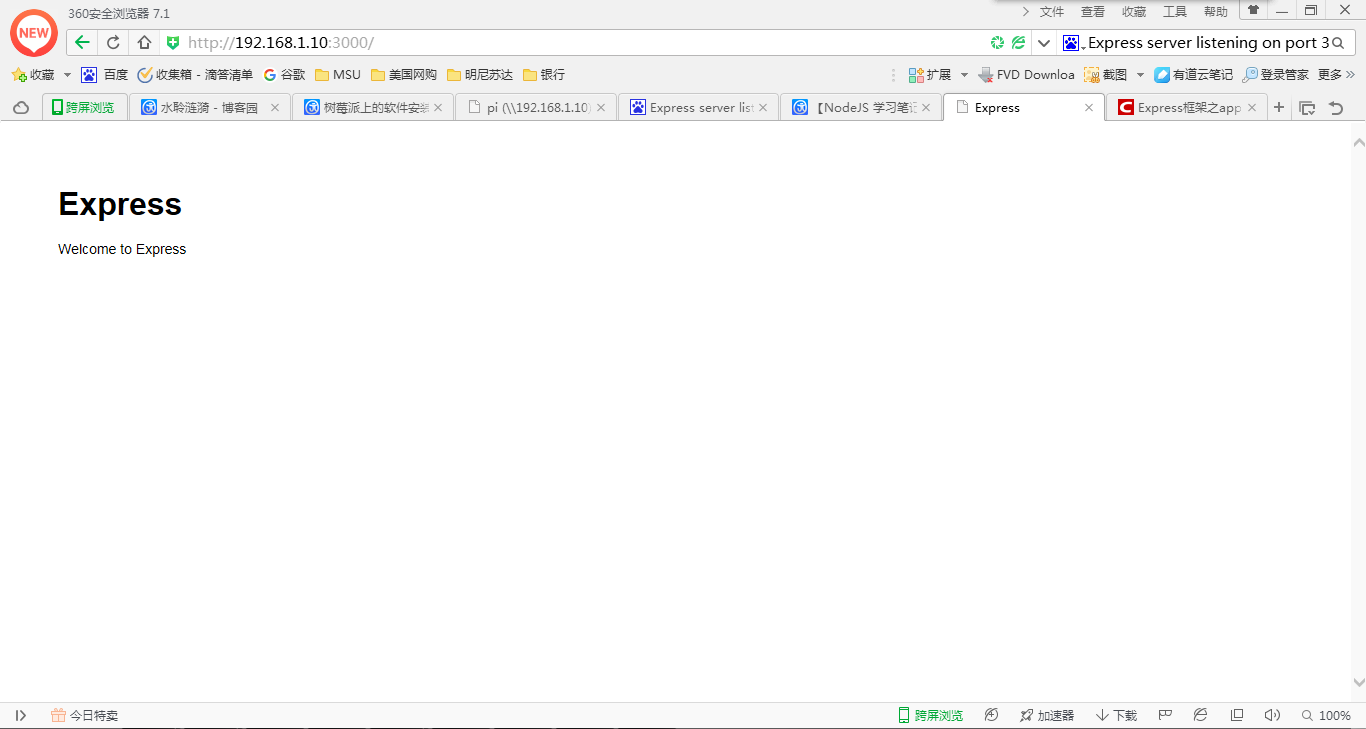


If all goes well, you should see the following message:

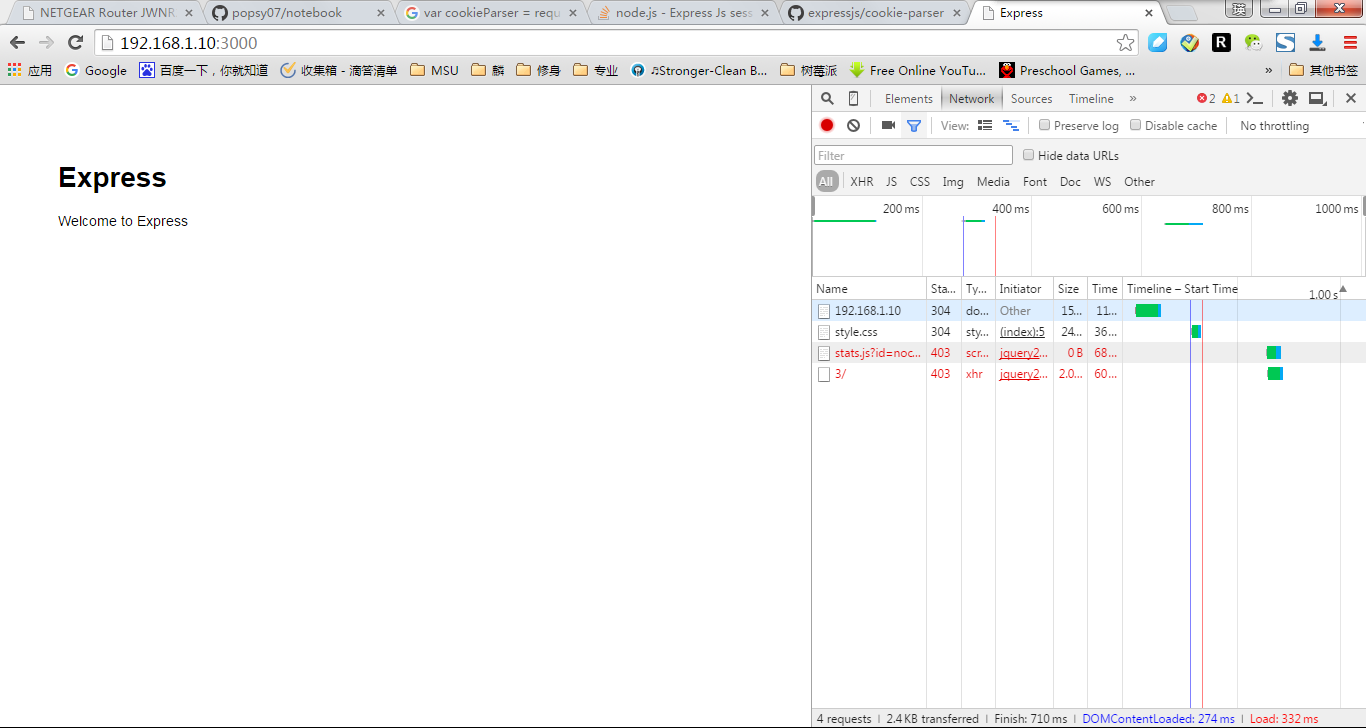
无此显示Express server listening on port 3000



1. Go out to your Raspberry Pi’s ip address at port 3000 and if you created Frankenstein, you should see something like this:



按F12，可查看



Congratulations!  You have successfully installed NodeJS, NPM and Express on the Raspberry Pi!