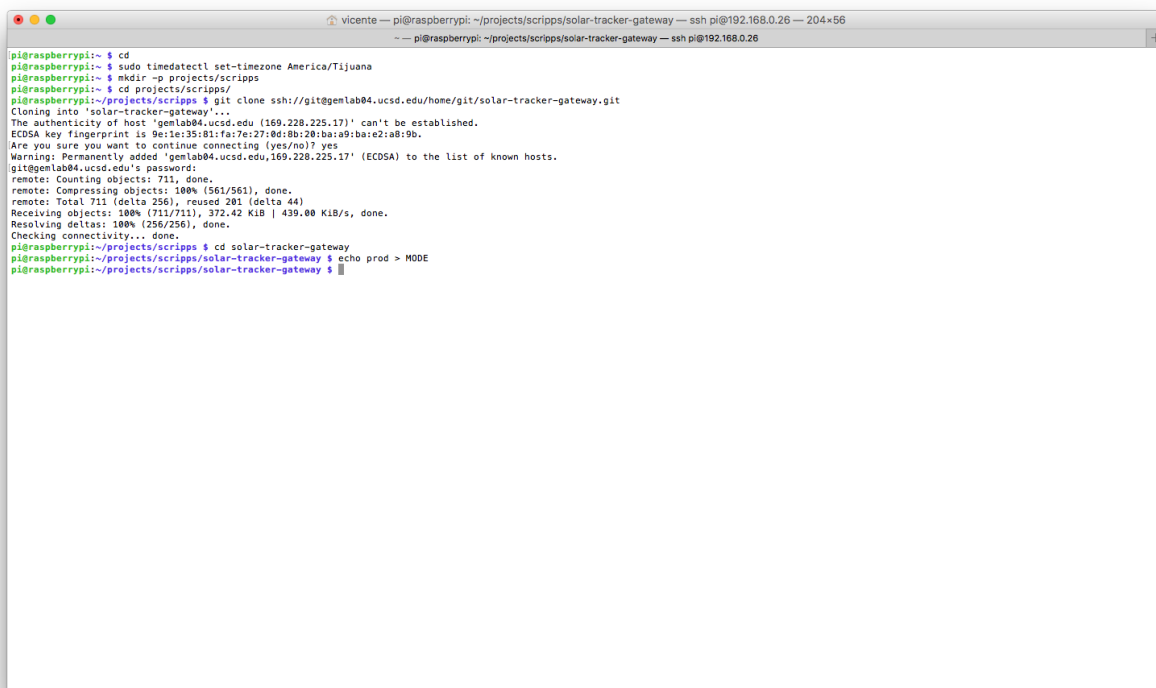


Gateway Setup

Open a terminal to enter next commands:

```
cd
sudo timedatectl set-timezone America/Tijuana
mkdir -p projects/scripps
cd projects/scripps/
git clone ssh://git@gemlab04.ucsd.edu/home/git/solar-tracker-gateway.git
cd solar-tracker-gateway
echo prod > MODE
```

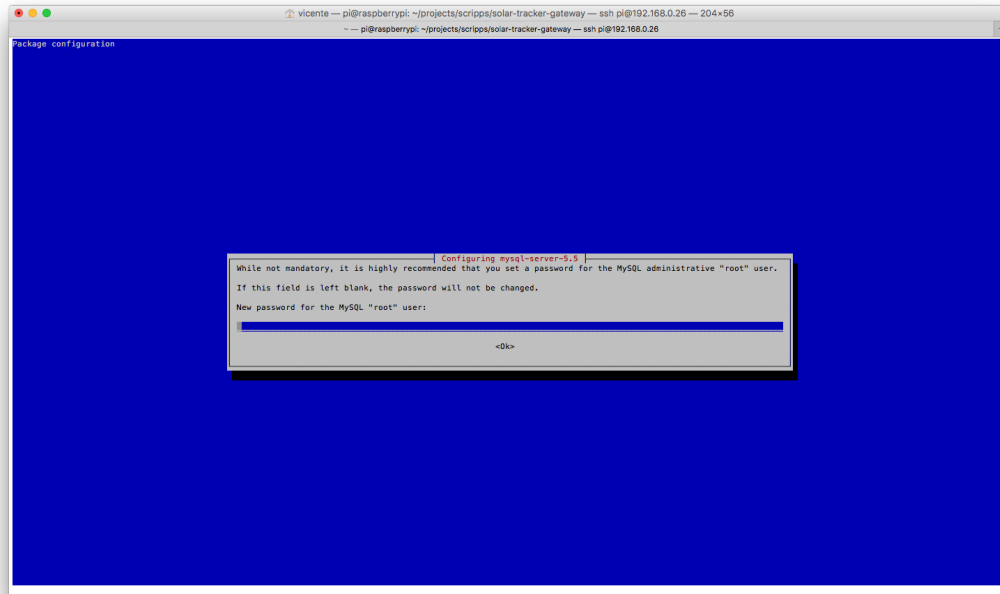


```
vicente — pi@raspberrypi: ~/projects/scripps/solar-tracker-gateway — ssh pi@192.168.0.26 — 204x56
~ — pi@raspberrypi: ~/projects/scripps/solar-tracker-gateway — ssh pi@192.168.0.26

pi@raspberrypi:~$ cd
pi@raspberrypi:~$ sudo timedatectl set-timezone America/Tijuana
pi@raspberrypi:~$ mkdir -p projects/scripps
pi@raspberrypi:~$ cd projects/scripps/
pi@raspberrypi:~/projects/scripps$ git clone ssh://git@gemlab04.ucsd.edu/home/git/solar-tracker-gateway.git
Cloning into 'solar-tracker-gateway'...
The authenticity of host 'gemlab04.ucsd.edu (169.228.225.17)' can't be established.
ECDSA key fingerprint is 9e1e3581:fa7e27:0d:8b:20:ba:a9:ba:e2:a8:9b.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'gemlab04.ucsd.edu,169.228.225.17' (ECDSA) to the list of known hosts.
git@gemlab04.ucsd.edu's password:
remote: Counting objects: 711, done.
remote: Compressing objects: 100% (561/561), done.
remote: Total 711 (delta 256), reused 201 (delta 44)
Receiving objects: 100% (711/711), 372.42 KiB | 439.00 KiB/s, done.
Resolving deltas: 100% (256/256), done.
Checking connectivity... done.
pi@raspberrypi:~/projects/scripps$ cd solar-tracker-gateway
pi@raspberrypi:~/projects/scripps/solar-tracker-gateway$ echo prod > MODE
pi@raspberrypi:~/projects/scripps/solar-tracker-gateway$
```

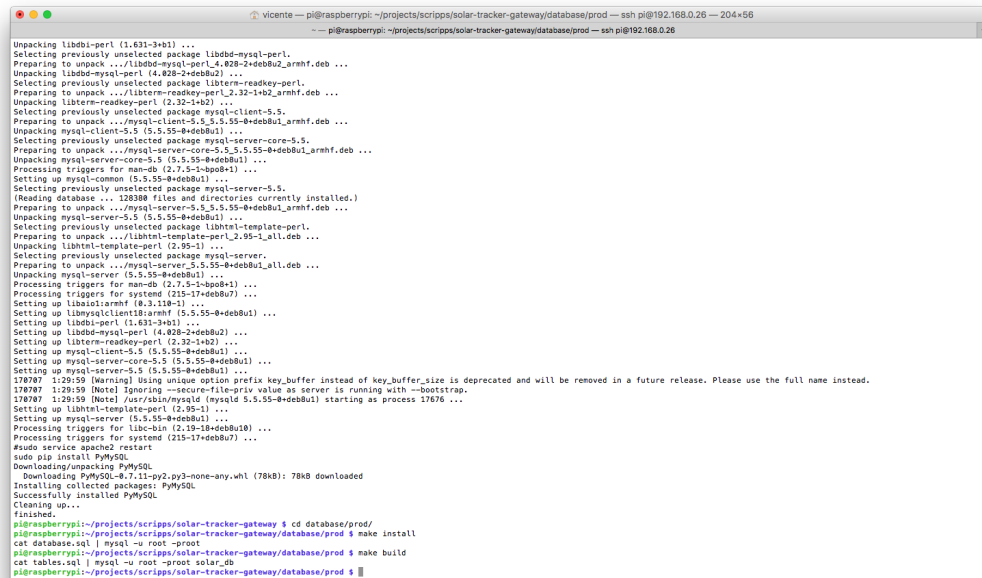
Enter next command but make sure that when mysql server password is asked, you enter “root”.

```
make install
```



Next commands are used to create a database:

```
cd database/prod/
make install
make build
```



Next, we need to add a service to listen to a Gateway device for incoming data, and another service to upload received data to a web server:

sudo crontab -e

```
cat tables.sql | mysql -u root -proot solar_db
[pi@raspberrypi:~/projects/scripps/solar-tracker-gateway/database/prod $ sudo crontab -e
no crontab for root - using an empty one
```

Select an editor. To change later, run 'select-editor'.

1. /bin/ed
2. /bin/nano <---- easiest
3. /usr/bin/vim.tiny

Choose 1-3 [2]: █

Press Enter to choose default (nano editor) or choose the one you prefer. In the editor that opens, add next two commands, after that, press Ctrl + O and Enter, then Ctrl + X:

@reboot /opt/solar-tracker/solar-tracker-gateway.py start

@reboot /opt/solar-tracker/solar-tracker-uploader.py



```
GNU nano 2.2.6 File: /tmp/crontab.1ENNdH/crontab Modified
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
@reboot /opt/solar-tracker/solar-tracker-gateway.py start
@reboot /opt/solar-tracker/solar-tracker-uploader.py
```

Now, after reboot (see next command) you can connect Gateway device.

sudo reboot