Server Project: Initial Setup Documentation

Corinne Koepfel & Simon Sommer
Advanced Numerical Methods and Data Analysis
University of St. Gallen

22.04.2018

1 Server setup

In order to setup our personal server, we each followed the tutorials by DigitalOcean, in the following order:

1. How to use SSH keys with DigitalOcean Droplets
2. Initial server setup
3. How to install LAMP
4. Setup domain and hostname
5. How to set up Apache Virtual Hosts
6. How to secure Apache with Let's Encrypt
7. How to install and secure phpMyAdmin
8. A basic MySQL tutorial

To test, go to `https://simonwsommer.ch` and say hi!

2 Python

Install python via anaconda

- First, check whether we need 32bit or 64bit

  ```
  lscpu
         Architecture: x86_64
  ```

  download installation bash file (64 bit) for miniconda

  ```
  wget https://repo.continuum.io/miniconda/Miniconda3-latest-Linux-x86_64.sh
  ```

  and install via

```
bash Miniconda3-latest-Linux-x86_64.sh

       Do you wish the installer to prepend the Miniconda3 install location
       to PATH in your /home/simon/.bashrc ?

Yes
```

Now python and a few basic libraries should be installed.

- We also want some kind of autocomplete for python, which we can install via:

  ```
  sudo apt install python-argcomplete
  ```

- Next, reboot to finish installation

  ```
  sudo reboot
  ```

- Check the functionality of conda (which is our python package manager)

  ```
  conda list
  ```

- If this works, we can delete our installation file

  ```
  rm Miniconda3-latest-Linux-x86_64.sh
  ```

3 Installing useful tools for python (not required)

This part serves mainly to make working on the server a lot easier (e.g. writing python code using vim).

- First, we need git

  ```
  sudo apt-get update
  sudo apt-get upgrade
  sudo apt-get install git
  ```

- Clone (download) Vundle, which is a package manager for vim

  ```
  git clone https://github.com/VundleVim/Vundle.vim.git ~/.vim/bundle/Vundle.vim
  ```

- Clone personal setup code (containing .bashrc file and all things needed to make vim powerful)

  ```
  git clone https://github.com/Sommer1872/initial_setup.git
  ```

  Note: since then, I changed the structure of the repo and moved the commands below into a single `setup.sh` file, which executes all commands in one go.

- Move a few files to home directory and make them hidden by prefixing a `.`

  ```
  mv setup/bashrc_example ~/.bashrc
  mv setup/vimrc ~/.vimrc
  mv setup/vim ~/.vim
  mv setup/git-completion.bash ~/.git-completion.bash
  mv setup/git-prompt.sh ~/.git-prompt.sh
  cd
  ```

- Install cmake, which is needed for YouCompleteMe

```
sudo apt-get install cmake
```

- Install YouCompleteMe, a strong autocomplete package for vim

```
cd ~/.vim/bundle/YouCompleteMe
./install.py
cd
```

- Install all other plugins

```
vim +PluginInstall +qall
```

- Rebooting

```
sudo reboot
```

4 Make conda environment and install packages

Here, we use conda to create a new workspace (called `environment` in anaconda lingo) which contains the packages we use to connect from our python scripts to the sql databases.

- First, create and activate a new environment (the sql-packages unfortunately still use python 2.7)

```
conda create --name quandl python=2.7
conda activate quandl
```

- Next, install the required packages via conda and pip:

```
conda install configparser
conda install mysql-python
conda install sqlalchemy

pip install quandl
```

- Finally, log into the mysql server and create a new database named `quandl_futures` :

```
mysql> CREATE DATABASE quandl_futures;
```

  We will later fill this database via python.

Now, everything should be ready for our python scripts to be executed without problems. We simply need to make sure that the `quandl` environment is activated (via `conda activate quandl`) before running the code. Otherwise there will be import errors because python doesn't find the packages we installed only in this environment.