Authors: Michael Garod, Ryan Kallicharan

Using Amazon Web Services

Step 1:

Create an Amazon Web Services(AWS) account. To complete this guide you must have administrator privilege.

Step 2:

Login into AWS. In management console, click Services->Analytics->EMR
Now click “Create cluster”.
Cluster name: <anything>
Launch mode: Cluster
Vender: Amazon
Application: spark
Instance type: <depend on needs> (we selected m3.xlarge)
Number of Instance: <depend on needs> (we selected 1)
Everything else leave as is and click “create cluster” to finish.

Step 3:

Click Services->Analytics->EMR
Click on the Cluster that was just created.
Make sure that under Network and Hardware; **Master**:Running 1 m3.xlarge
Copy down the **Master public DNS:** <your cluster dns>
You will need it later.
Example : ec2-54-236-227-115.compute-1.amazonaws.com

Step 4:

In management console, click Services->Compute->EC2
Click “Launch Instance”
Select **Amazon Linux 64-bit**
[Warning] Depending what you choose could cost money.
Amazon at this time is offering Free 1 year for t2.mirco
Meaning once the instance has been shut down the data that you have uploaded or worked on will be deleted
Ok, select one and click “review and launch”

Step 5:

click Services->Compute->EC2
On left panel Locate Network&Security->key pairs
Create a key pair
Name the key pair (we called ours testkey)
File download should be of this for testkey.pem

This Important for accessing the EC2 over ssh
And download the file

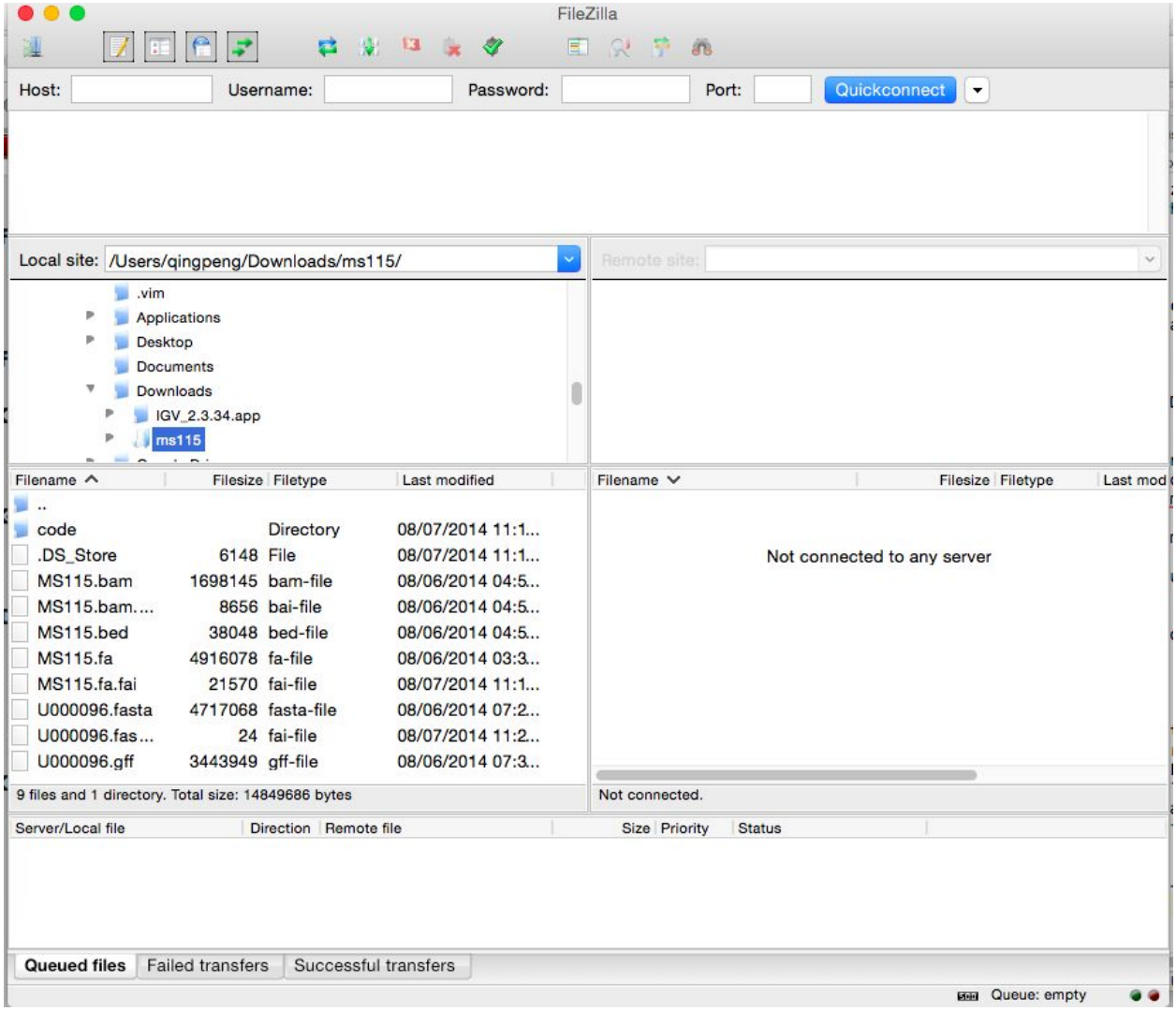
Now Locate Network&Security->Security Groups
Click on group name : default
Click on “inbound” tab
clIck Add Rule: Select HTTP, and source to be anywhere and 0.0.0.0/0
clIck Add Rule: Select HTTPS, and source to be anywhere and 0.0.0.0/0
clIck Add Rule: Select SSH, and source to be anywhere and 0.0.0.0/0
And save
click Services->Compute->EC2->instances->instances (to view)
Waiting until both EMR and EC2 Instance State is running

Step 6 (meant for linux users and OSX, Windows users should use putty to ssh):
Once the instances are running, we will connect to the EC2 instance
Click the EC2 instance and click connect:
Follow the instructions here from amazon pop up.
Open up your terminal on your local machine and type the ssh command:
Example:
ssh -i "testkey.pem" ec2-user@ec2-54-172-130-202.compute-1.amazonaws.com

Now type sudo -i
You are now in root
Navigate to your user folder
Example: cd home/ec2-user

Step 7 (step filezilla for file transfer):

Firstly, go to ‘https://filezilla-project.org/‘ and click “Download FileZilla Client” button to download it.
The interface of FileZilla is like this:



If you want to use FileZila to upload to or download data from a normal FTP server if you have the user and password, just put the information in the “Host”, “Username”, “Password” box and connect. However for Amazon instance, we use key-pair to log in instead of password for better safety. So it is a little bit more complicated to configure.
Open “Settings” and click “SFTP”:

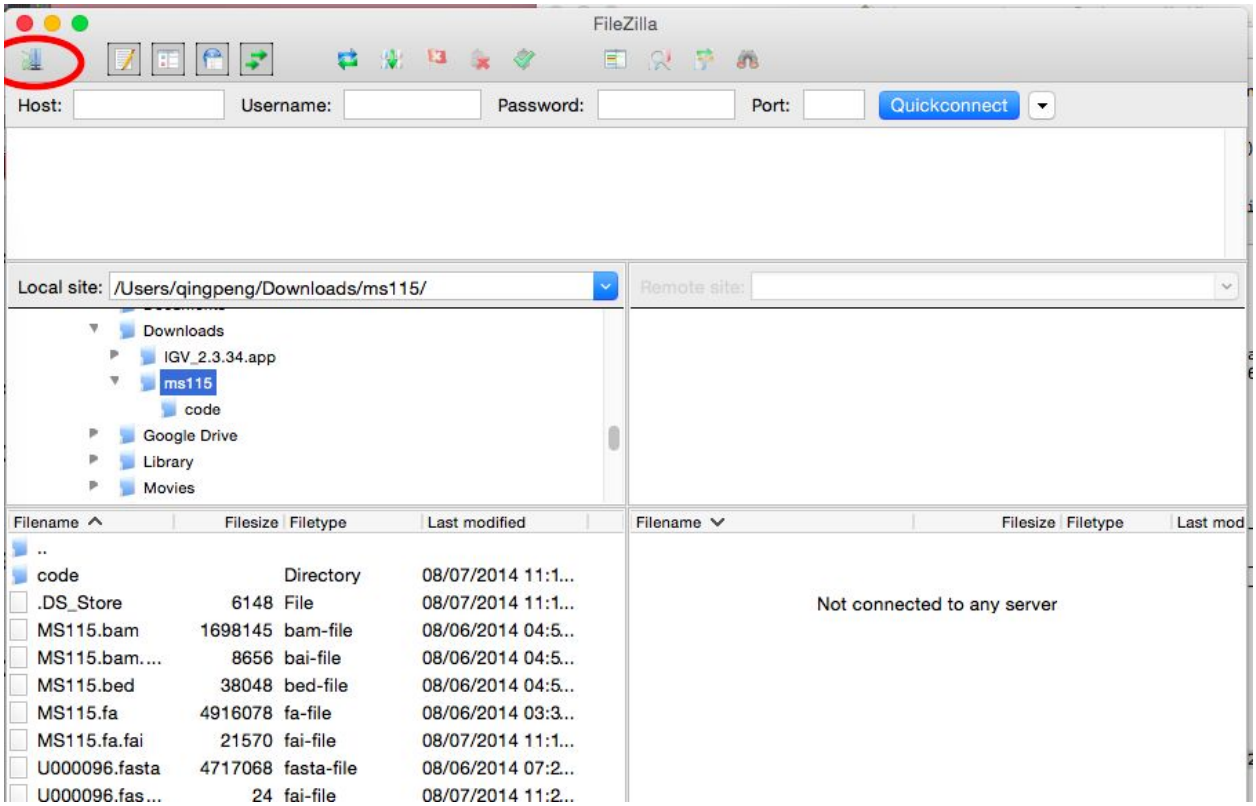
Click “Add keyfile...”:

Then select the ”.pem” file you used to connect to Amazon instance with ssh.

There is a dialog box to ask you if you want to convert the ”.pem” file into a supported format. Click “Yes”**. If the dialog box does not appear change the file extension to ppk manual. But also keeping a copy .pem file as well**

Name it with extension as ”.ppk” and save it.

You will see the a private key has been added.
Close “Settings” and go back to the main interface.
Click button to open the site manager.

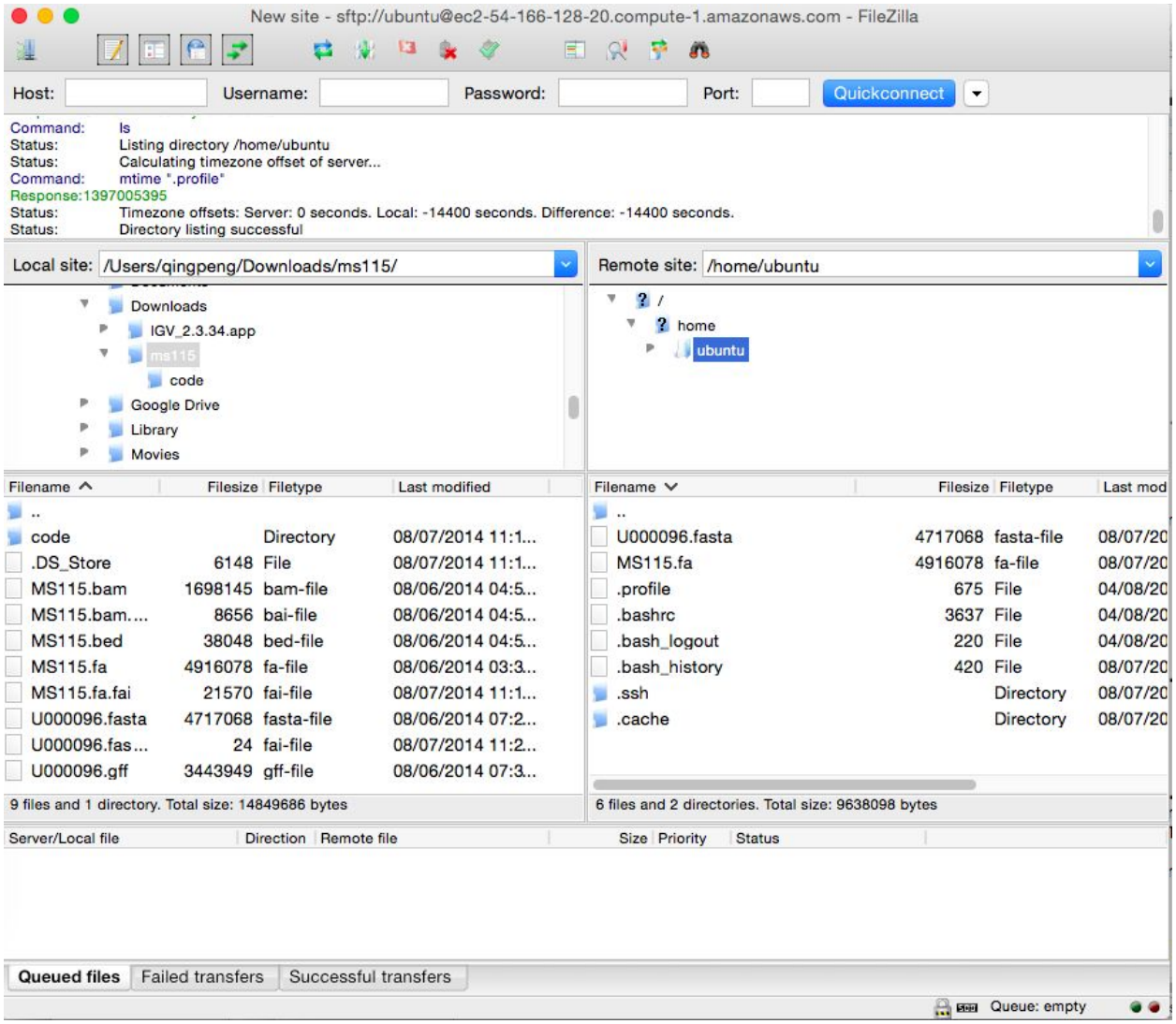


Click “New Site”.

Put the Amazon instance URL like example: ec2-54-166-128-20.compute-1.amazonaws.com in the “Host” box. Set “Protocol” as “SFTP”, “Logon Type” as “key file”, brower for .ppk example testkey.ppk Then click “Connect”.

There will be a dialogue box to ask you about “Unknown host key”, just click “Ok”.

All right. Now you have logged in the Amazon instance. You can drag and drop to transfer the files between the remote machine and your local laptop.



Step 8 (setting up start on instance) :
Download spark on local machine and extract.
You should now have a folder called spark-1.6.1
1.6.1 is the current release if you have a later release it will be different change accordingly.
Now in FIleZilla move spark-1.6.1 to home/ec2-user (this should be simple to figure out on your own)

Once copied over, Go back to terminal window where we are currently connected to our EC2 instance and type the following command

sudo mv spark-1.6.1 /srv/spark-1.6.1
ln -s /srv/spark-1.4.8 /srv/spark

Now Spark is ready to be used

Step 9:
Use filezilla to copy your python script and data needed to home/ec2-user

Step 10:
Running your code:
Every Time you wish to use spark run this in the EC2 terminal to set the environment

Must run this in terminal every time you logon to your EC2 instance for spark to run:
export SPARK_HOME=/srv/spark
export PATH=$SPARK_HOME/bin:$PATH

After this you can run your spark code:

To run the code using EMR:
spark-submit --master spark://<EMR dns> <your script>

example:
spark-submit --master spark://ec2-54-236-227-115.compute-1.amazonaws.com pysparktest.py

To run the code just on just EC2 instance::
spark-submit <your script>

Important side note for running script on large data:
It will be expensive to have large data on your EC2 instance and impractical.
You will have to set up an AWS s3 storage to hold your data.
Create a bucket and upload your large files there.
In your script you can use **boto** libraries to connect to the s3 data.

Example code (all argument in following code are passed as strings):
import boto from boto.s3.key import Key
conn = boto.connect_s3(< access id>,<secret access key>)
bucket = conn.get_bucket(<name of bucket>)
k = Key(bucket)
k.key = <file name>
dataFile = k.key