Create a Deep learning Ubuntu server with AWS

Version: v01-4
Date: Saturday, 13 March 2021

Introduction

The aim of this project is to set up an Ubuntu server with GPU for Deep Learning projects.

I will start with these guides:

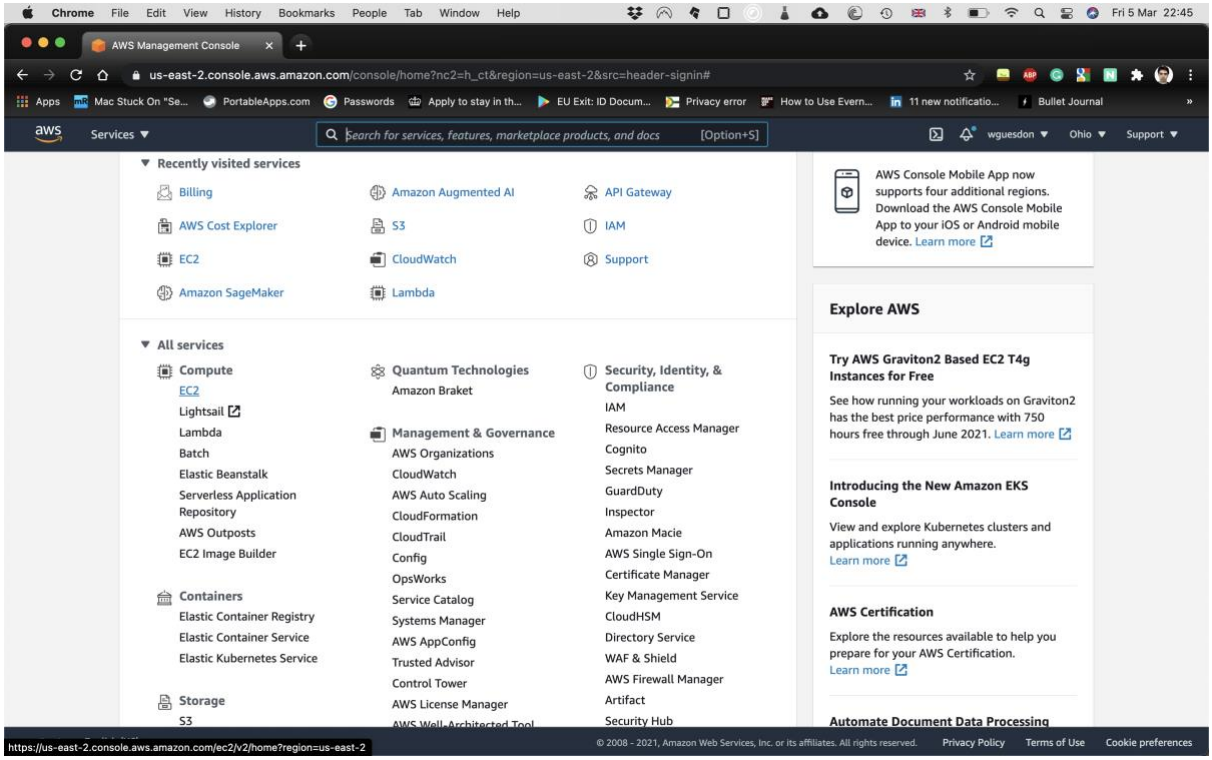
- https://becominghuman.ai/set-up-your-aws-deep-learning-server-for-free-48e2b21ec627
- https://towardsdatascience.com/stop-worrying-and-create-your-deep-learning-server-in-30-minutes-bb5bd956b8de

Once the set-up is validated, I will automate these steps with a script. See this guide:
https://bloggingnectar.com/aws/automate-your-ec2-instance-setup-with-ec2-user-data-scripts/
https://www.digitalocean.com/community/tutorials/automating-initial-server-setup-with-ubuntu-18-04

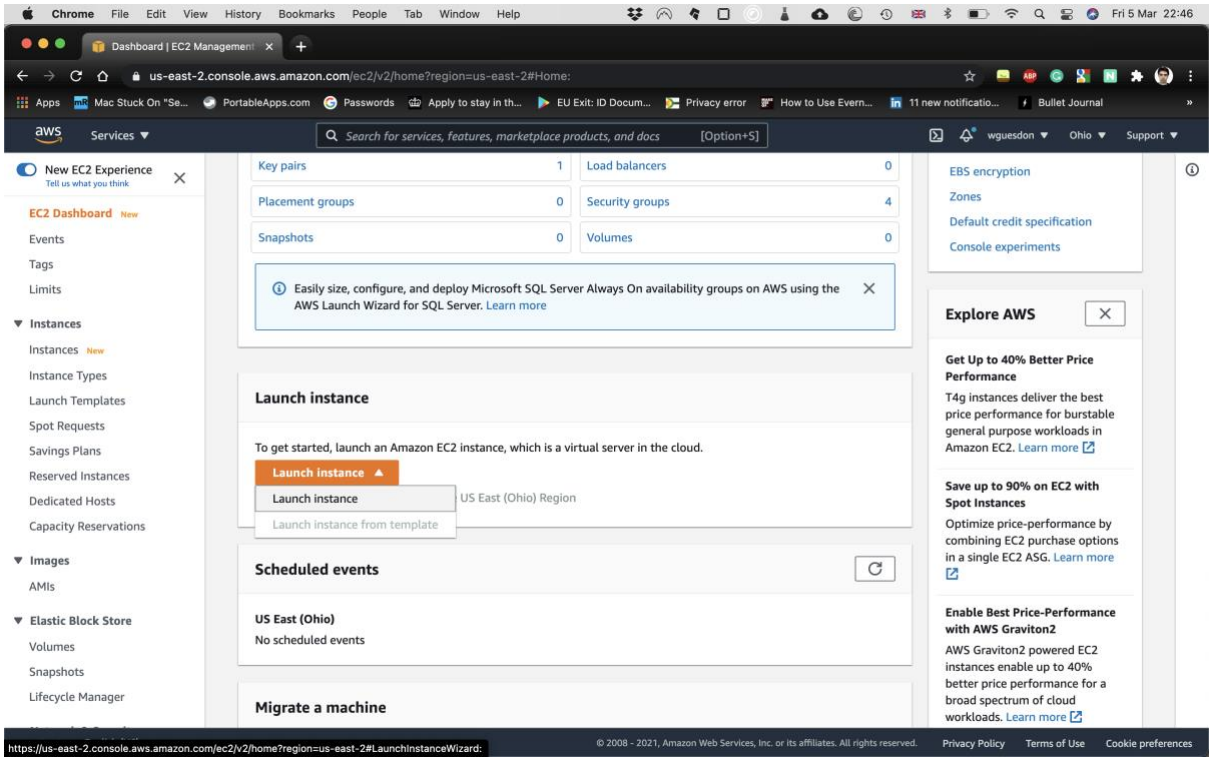
1. Table of Contents

2. Create an instance

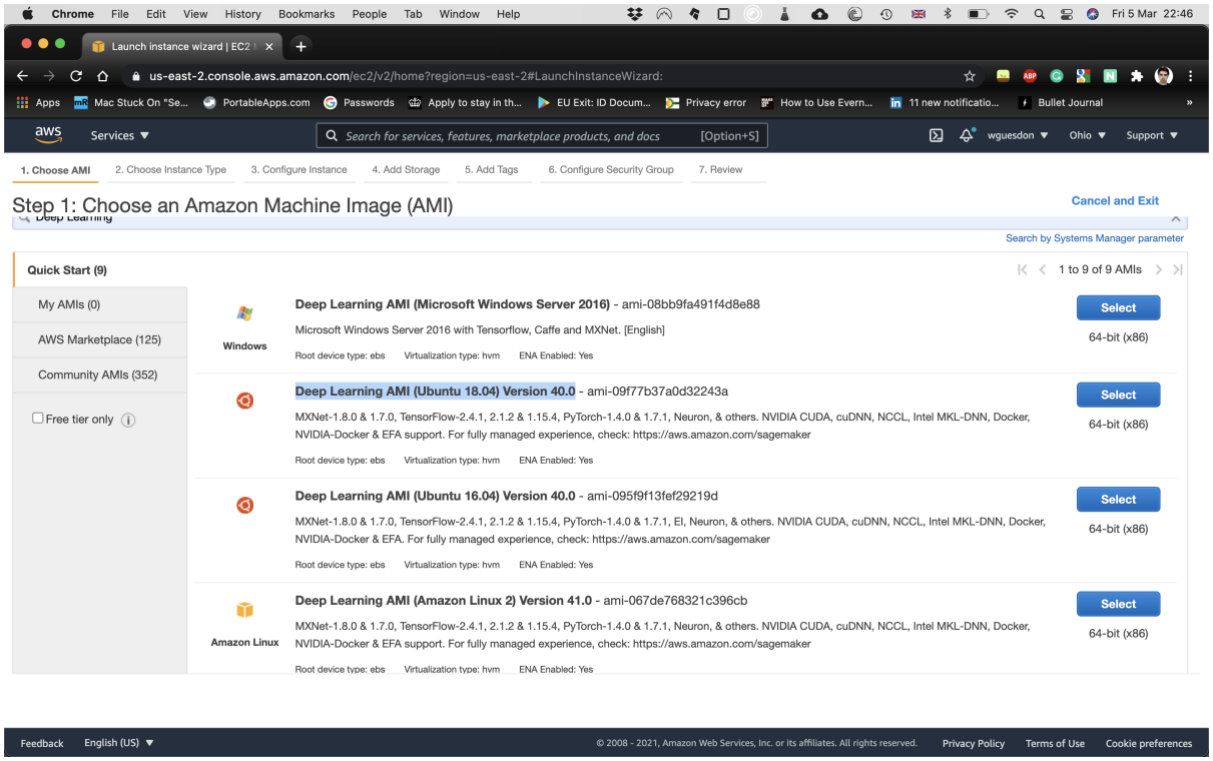
2.1. Select the EC2 service



2.2. Start the Instance



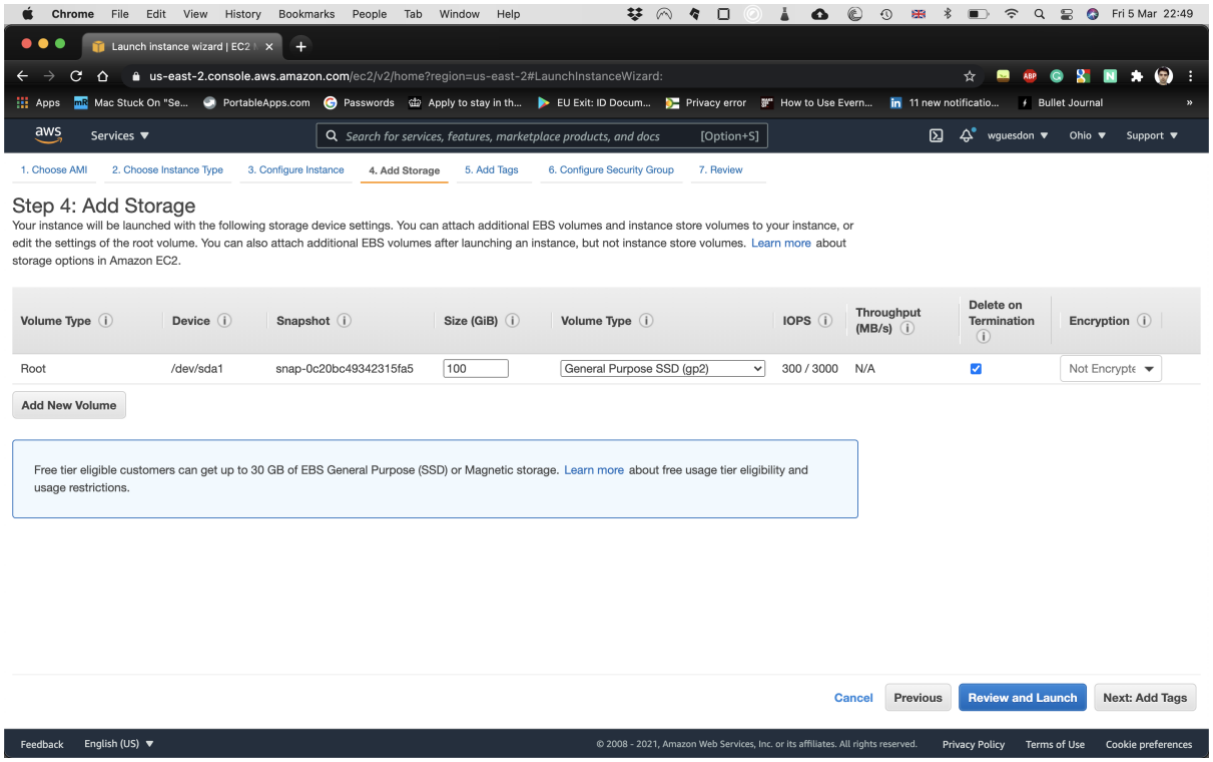
2.3. Select the Ubuntu Deep Learning server



Choose the instance type. For configuration practice I will choose t2.xlarge

For CPU heavy analysis: t3a.2xlarge
For GPU heavy analysis: p2.xlarge

For practicing this set up I will keep a 100 GB disk. Disk space is charged 0.1$ per GB per month

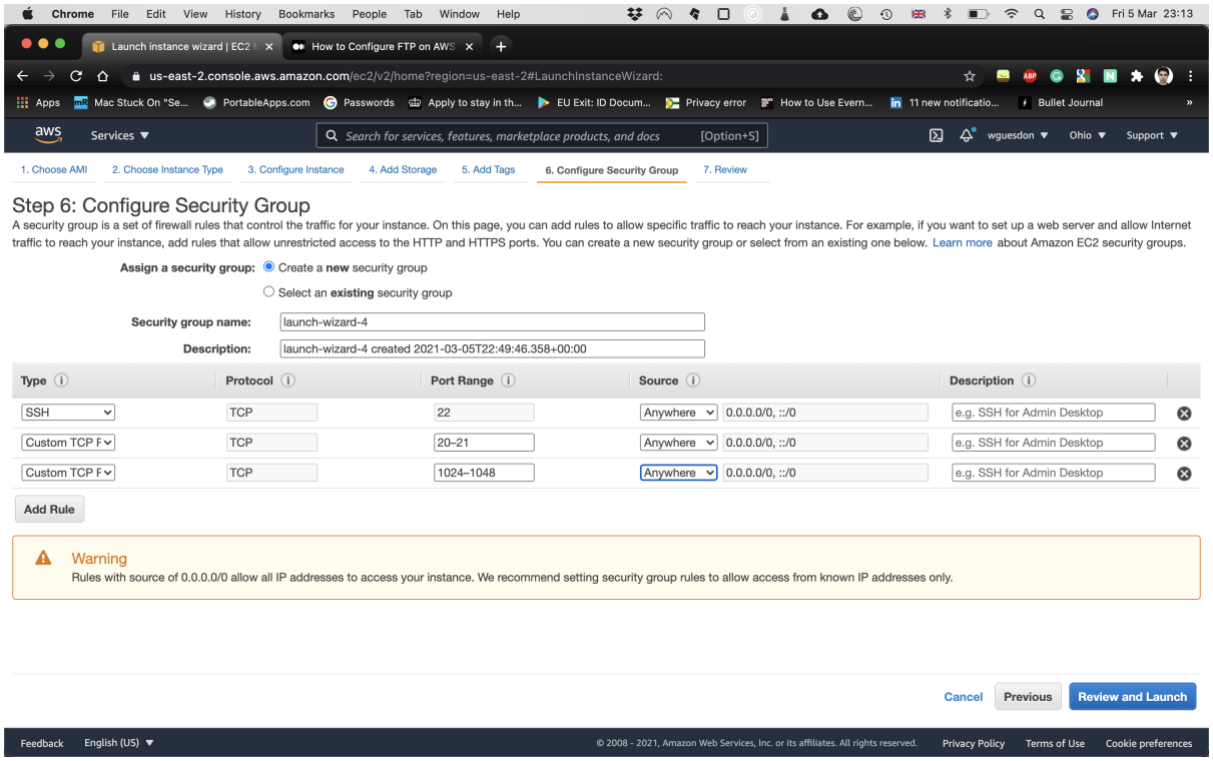


2.4. Configure the permissions

Allow port for SSH

Allow port for FTP

https://medium.com/tensult/configure-ftp-on-aws-ec2-85b5b56b9c94



2.5. A window will pop up for the setup of the SSH key pair

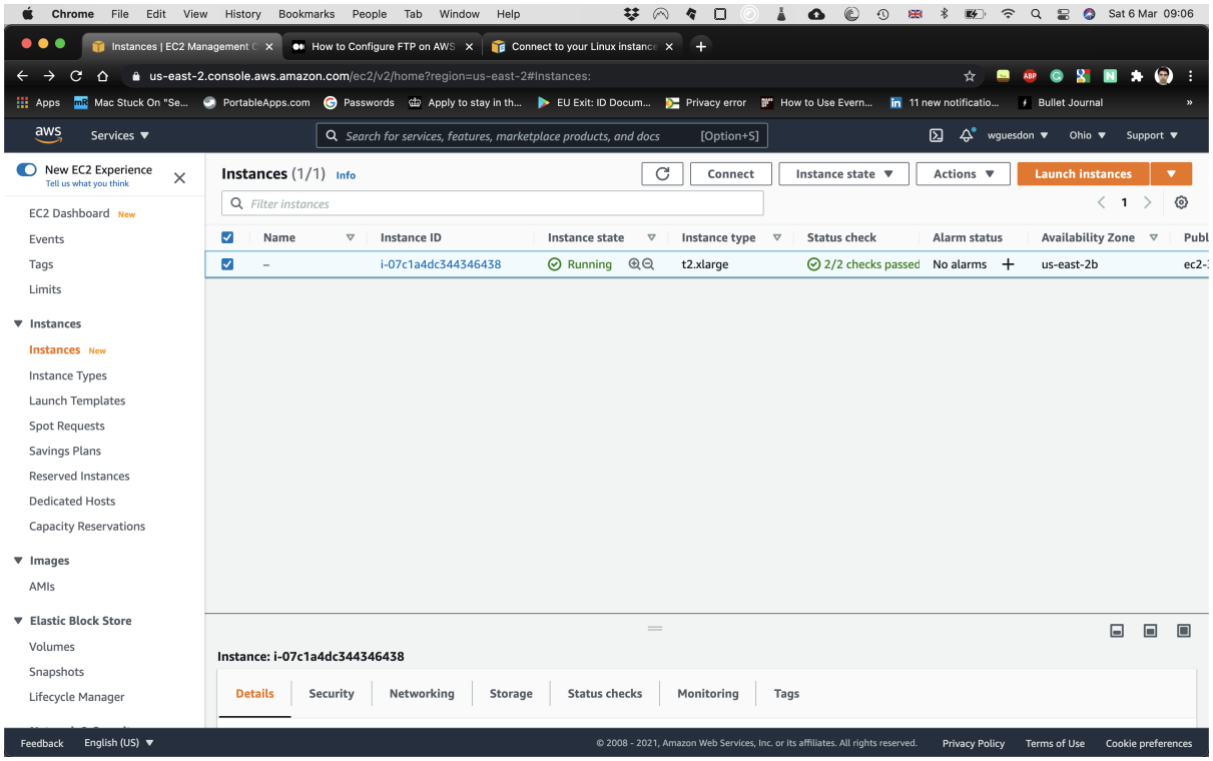
I will use the SSH key pair that I downloaded previously. You can also set up a new one.

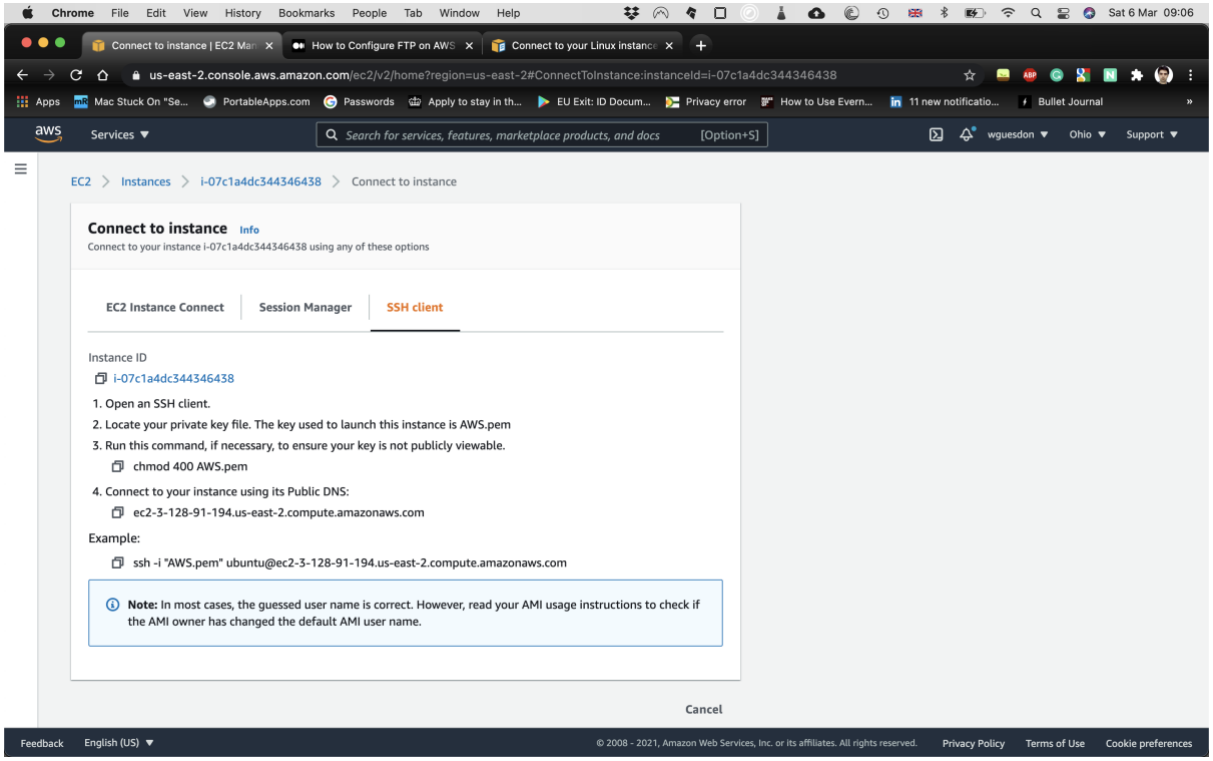
3. Connect to the instance

On Mac/Linux the terminal natively supports SSH

On windows you can use Putty or now WSL which is ubuntu bash on Windows.

See the instructions here: https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/WSL.html






```
ssh -i ~/Desktop/SHH_keys/AWS.pem ubuntu@ec2-3-128-91-194.us-east-2.compute.amazonaws.com
```

4. Install Tmux to run operation on server after SSH disconnection

This come preinstalled on the Deep Learning Ubuntu 18.04 instance. For other instance

```
sudo apt-get update
sudo apt-get install tmux
```

5. Install Docker

This come pre-installed in Ubuntu 18.04 Deep Learning. If needed this is the steps for installation.

```
sudo apt update
sudo apt install apt-transport-https ca-certificates curl software-properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu bionic stable"
sudo apt update
apt-cache policy docker-ce
sudo apt install docker-ce
sudo systemctl status docker
# exit with control + c
```

6. Install and configure Anaconda

This come pre-installed in Ubuntu 18.04 Deep Learning. If needed this is the steps for installation.

see https://medium.com/google-cloud/set-up-anaconda-under-google-cloud-vm-on-windows-f71fc1064bd7

```
sudo apt-get update
sudo apt-get install bzip2 libxml2-dev

wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
bash Miniconda3-latest-Linux-x86_64.sh
rm Miniconda3-latest-Linux-x86_64.sh
source .bashrc
conda
conda create --name ml
```

```
conda activate ml
conda install scikit-learn pandas jupyter ipython
```

7. Connecting to Jupyter notebook

See https://towardsdatascience.com/stop-worrying-and-create-your-deep-learning-server-in-30-minutes-bb5bd956b8de

```
# Start the jupyter notebook in a tmux session which will run on the server even after disconnection from the SHH terminal
tmux new -s StreamSession
jupyter notebook
```

Note the token for connection for example here:
token=3d4581f9fb8bbe2649a69cfdae8d135376596d642695cb0a

Connect to the Jupyter notebook session by SSH in a new terminal

```
ssh -i ~/Desktop/SHH_keys/AWS.pem -L 8001:localhost:8888 ubuntu@ec2-3-128-91-194.us-east-2.compute.amazonaws.com
```

Connect to session on: http://localhost:8001/
All set, you are now running a Jupyter notebook on the Ubuntu instance in under 5 min.

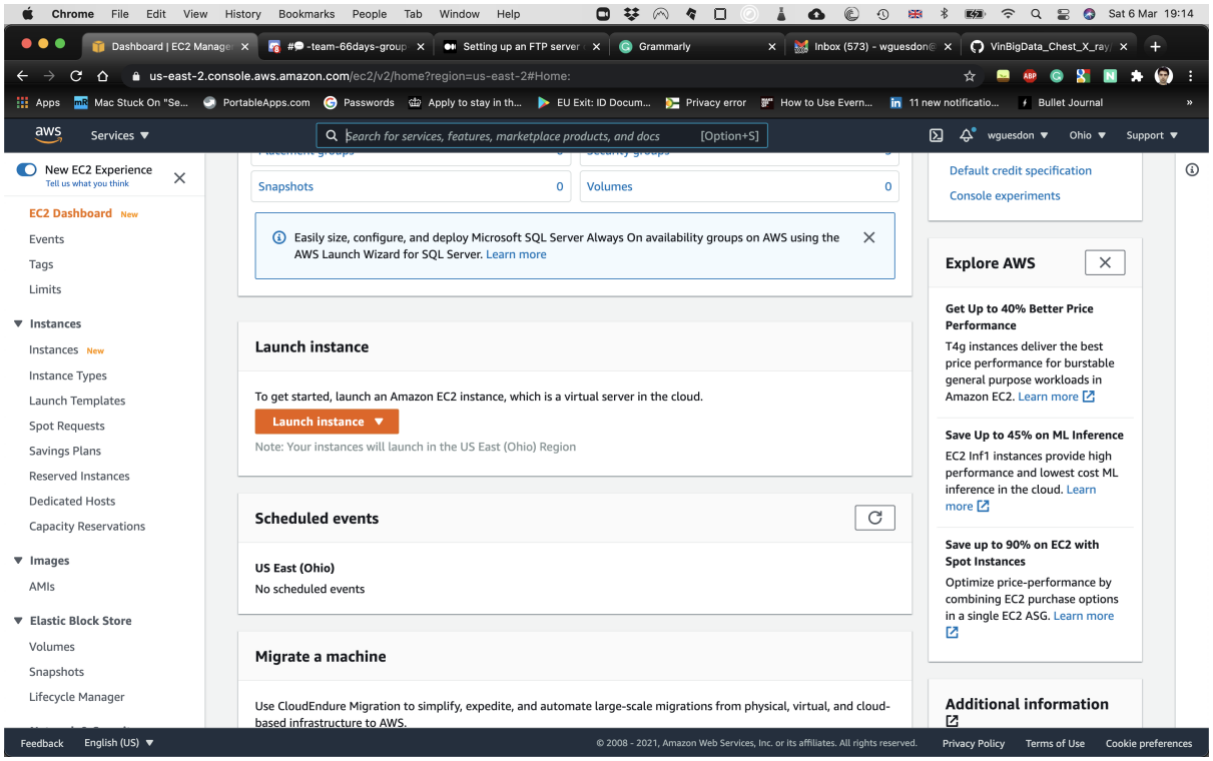
8. Files sharing with your instance

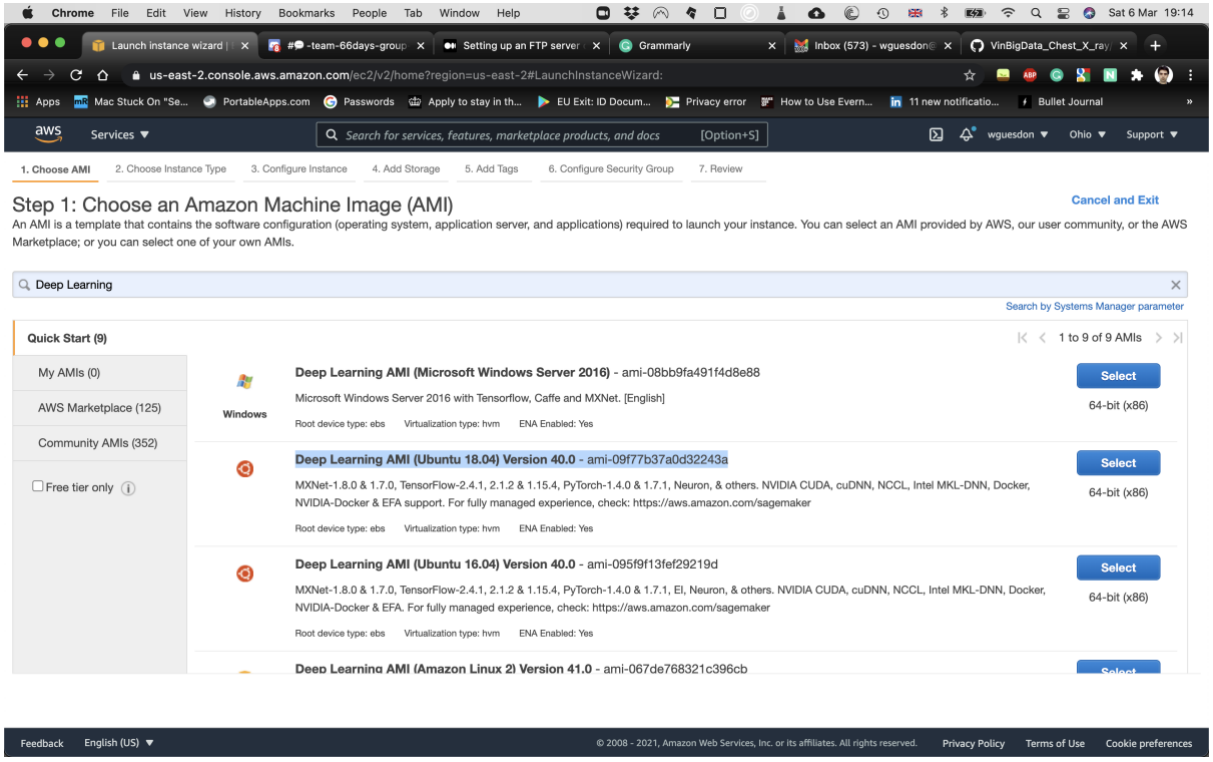
8.1. File sharing with Rsync

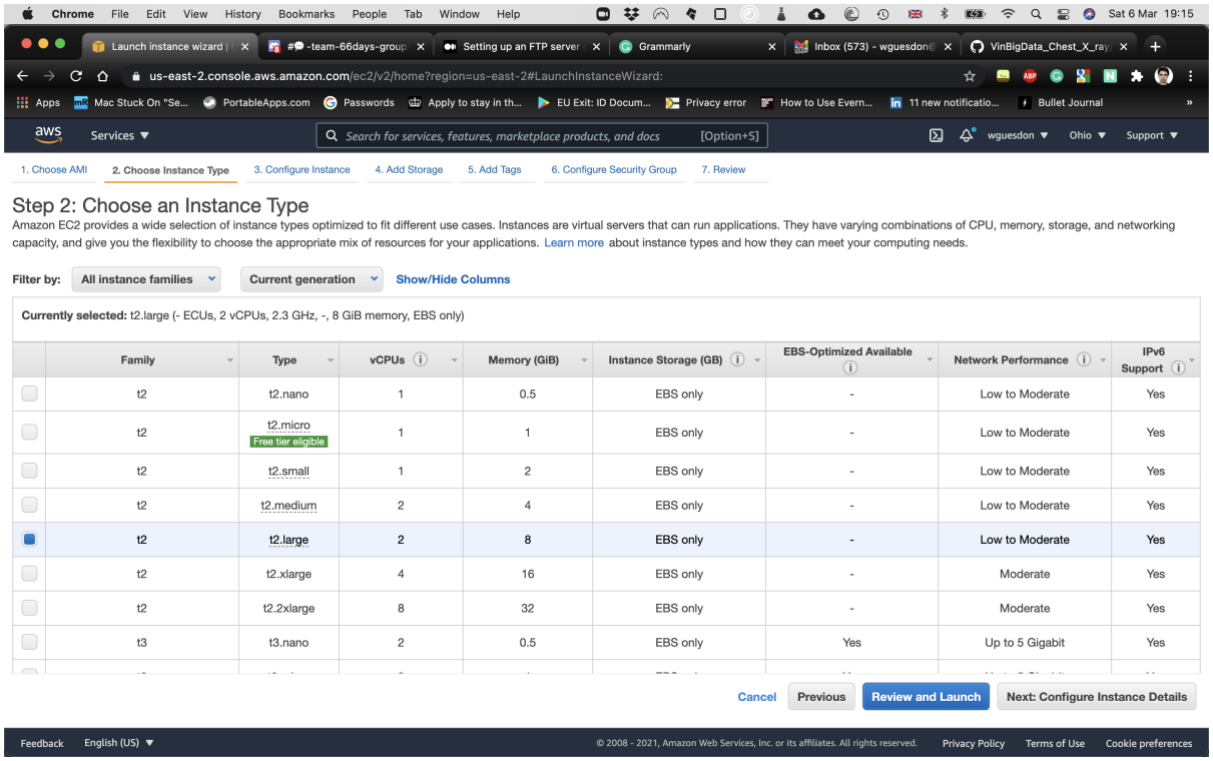
See this guide: https://linuxize.com/post/how-to-transfer-files-with-rsync-over-ssh/

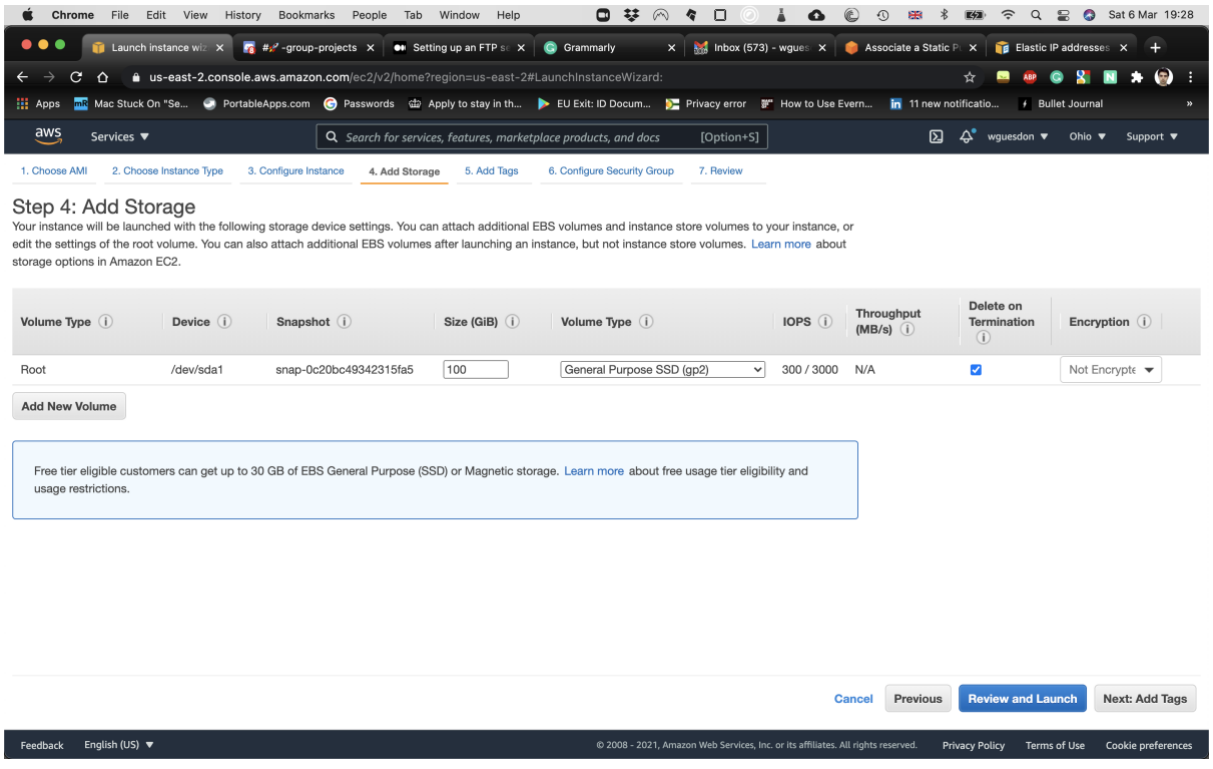
8.2. File sharing with FTP

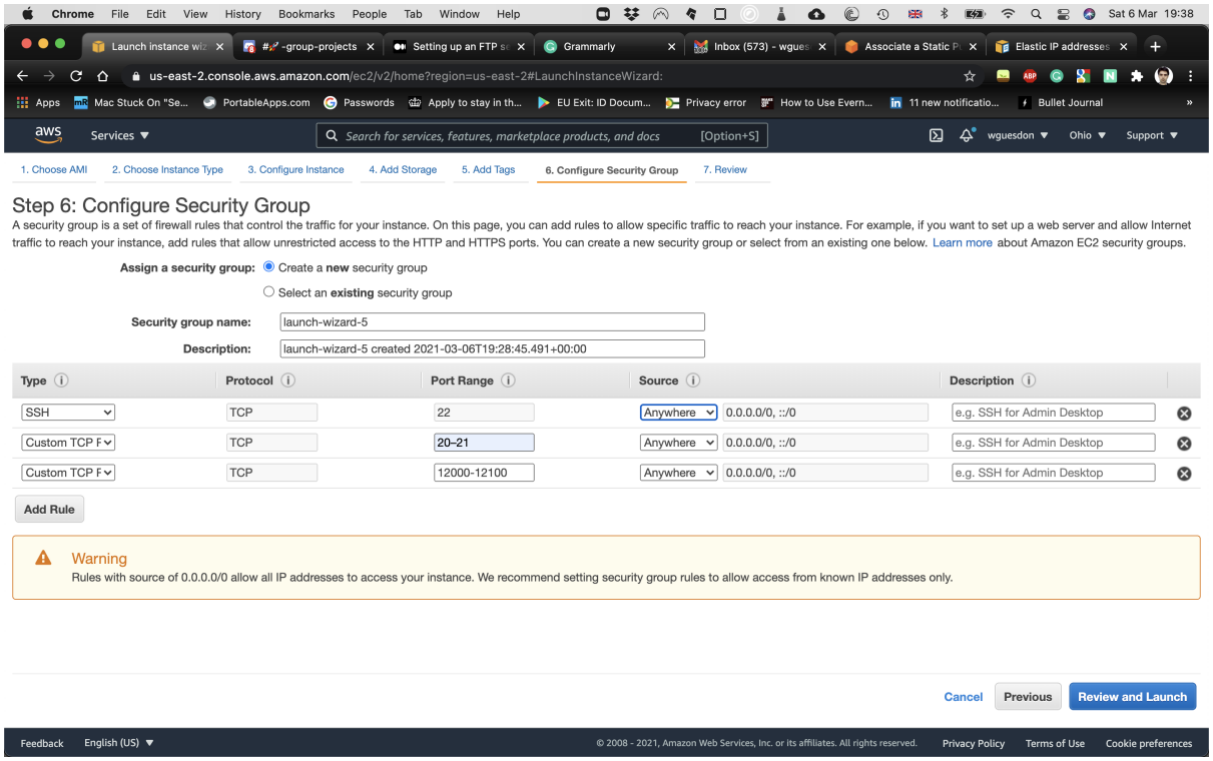
See https://jayden-chua.medium.com/setting-up-an-ftp-server-on-ubuntu-18-04-on-aws-79bd55ab32bb


Chrome File Edit View History Bookmarks People Tab Window Help
Sat 6 Mar 19:14
Dashboard | EC2 Manage
#-team-66days-group
Setting up an FTP server
Grammarly
Inbox (573) - wguesdon
VinBigData_Chest_X_ray
us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#Home:
Apps
Mac Stuck On "Se...
PortableApps.com
Passwords
Apply to stay in th...
EU Exit: ID Docum...
Privacy error
How to Use Evern...
11 new notificatio...
Bullet Journal
aws
Services
Search for services, features, marketplace products, and docs [Option+S]
wguesdon
Ohio
Support
New EC2 Experience
Tell us what you think
EC2 Dashboard New
Events
Tags
Limits
Instances
Instances New
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Snapshots 0
Volumes 0
Easily size, configure, and deploy Microsoft SQL Server Always On availability groups on AWS using the AWS Launch Wizard for SQL Server. Learn more
Launch instance
To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.
Launch instance
Note: Your instances will launch in the US East (Ohio) Region
Scheduled events
US East (Ohio)
No scheduled events
Migrate a machine
Use CloudEndure Migration to simplify, expedite, and automate large-scale migrations from physical, virtual, and cloud-based infrastructure to AWS.
Default credit specification
Console experiments
Explore AWS
Get Up to 40% Better Price Performance
T4g instances deliver the best price performance for burstable general purpose workloads in Amazon EC2. Learn more
Save Up to 45% on ML Inference
EC2 Inf1 instances provide high performance and lowest cost ML inference in the cloud. Learn more
Save up to 90% on EC2 with Spot Instances
Optimize price-performance by combining EC2 purchase options in a single EC2 ASG. Learn more
Additional information
Feedback
English (US)
© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Cookie preferences


Chrome File Edit View History Bookmarks People Tab Window Help
Sat 6 Mar 19:14
Launch instance wizard |
#-team-66days-group
Setting up an FTP server
Grammarly
Inbox (573) - wguesdon
VinBigData_Chest_X_ray
us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:
Apps
Mac Stuck On "Se...
PortableApps.com
Passwords
Apply to stay in th...
EU Exit: ID Docum...
Privacy error
How to Use Evern...
11 new notificatio...
Bullet Journal
aws
Services
Search for services, features, marketplace products, and docs [Option+S]
wguesdon
Ohio
Support
1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
Step 1: Choose an Amazon Machine Image (AMI)
Cancel and Exit
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.
Deep Learning
Search by Systems Manager parameter
Quick Start (9)
1 to 9 of 9 AMIs
My AMIs (0)
AWS Marketplace (125)
Community AMIs (352)
Free tier only
Deep Learning AMI (Microsoft Windows Server 2016) - ami-08bb9fa491f4d8e88
Microsoft Windows Server 2016 with Tensorflow, Caffe and MXNet. [English]
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
Windows
Select
64-bit (x86)
Deep Learning AMI (Ubuntu 18.04) Version 40.0 - ami-09f77b37a0d32243a
MXNet-1.8.0 & 1.7.0, TensorFlow-2.4.1, 2.1.2 & 1.15.4, PyTorch-1.4.0 & 1.7.1, Neuron, & others. NVIDIA CUDA, cuDNN, NCCL, Intel MKL-DNN, Docker, NVIDIA-Docker & EFA support. For fully managed experience, check: https://aws.amazon.com/sagemaker
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
Select
64-bit (x86)
Deep Learning AMI (Ubuntu 16.04) Version 40.0 - ami-095f9f13fef29219d
MXNet-1.8.0 & 1.7.0, TensorFlow-2.4.1, 2.1.2 & 1.15.4, PyTorch-1.4.0 & 1.7.1, EI, Neuron, & others. NVIDIA CUDA, cuDNN, NCCL, Intel MKL-DNN, Docker, NVIDIA-Docker & EFA. For fully managed experience, check: https://aws.amazon.com/sagemaker
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
Select
64-bit (x86)
Deep Learning AMI (Amazon Linux 2) Version 41.0 - ami-067de768321c396cb
Select
Feedback
English (US)
© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Cookie preferences

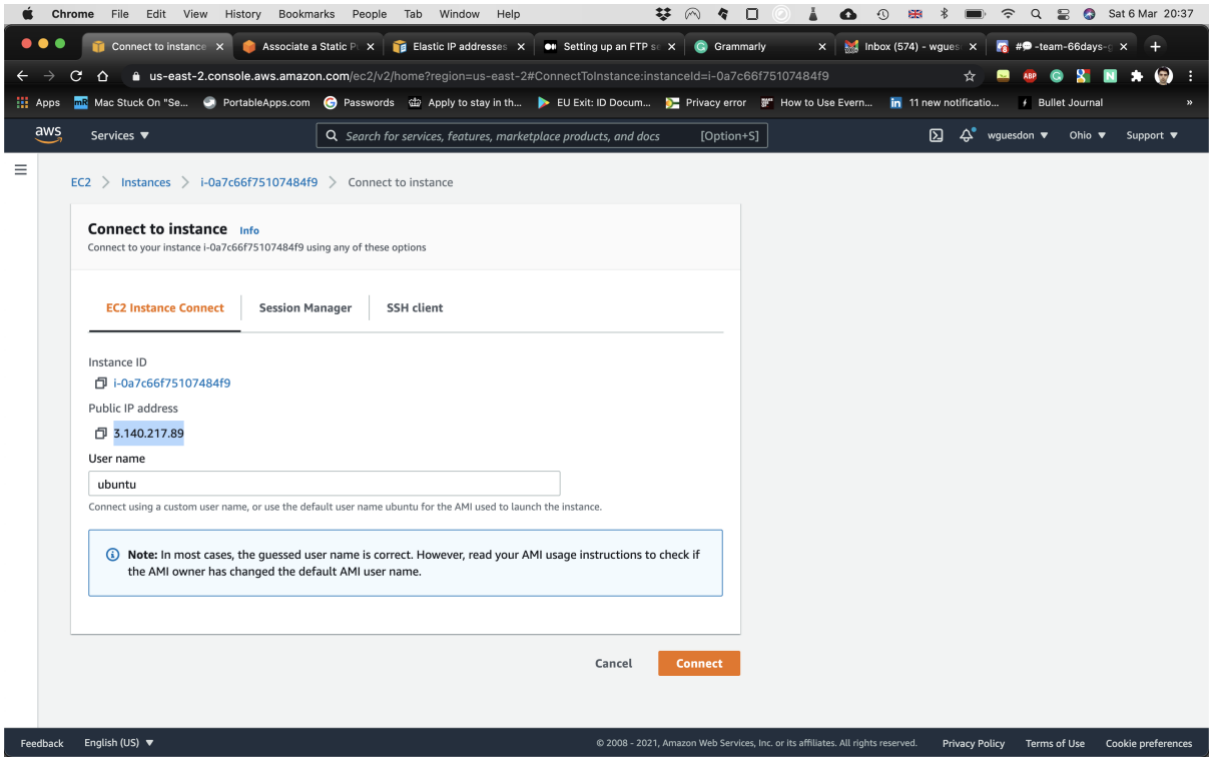

Chrome File Edit View History Bookmarks People Tab Window Help
Sat 6 Mar 19:15
Launch instance wizard
Setting up an FTP server
Grammarly
Inbox (573) - wguesdon
us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:
Apps
Mac Stuck On "Se...
PortableApps.com
Passwords
Apply to stay in th...
EU Exit: ID Docum...
Privacy error
How to Use Evern...
11 new notificatio...
Bullet Journal
aws
Services
Search for services, features, marketplace products, and docs [Option+S]
wguesdon
Ohio
Support
1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
Step 2: Choose an Instance Type
Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.
Filter by:
All instance families
Current generation
Show/Hide Columns
Currently selected: t2.large (- ECUs, 2 vCPUs, 2.3 GHz, -, 8 GiB memory, EBS only)
Family
Type
vCPUs
Memory (GiB)
Instance Storage (GB)
EBS-Optimized Available
Network Performance
IPv6 Support
t2 t2.nano 1 0.5 EBS only - Low to Moderate Yes
t2 t2.micro Free tier eligible 1 1 EBS only - Low to Moderate Yes
t2 t2.small 1 2 EBS only - Low to Moderate Yes
t2 t2.medium 2 4 EBS only - Low to Moderate Yes
t2 t2.large 2 8 EBS only - Low to Moderate Yes
t2 t2.xlarge 4 16 EBS only - Moderate Yes
t2 t2.2xlarge 8 32 EBS only - Moderate Yes
t3 t3.nano 2 0.5 EBS only Yes Up to 5 Gigabit Yes
Cancel
Previous
Review and Launch
Next: Configure Instance Details
Feedback
English (US)
© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Cookie preferences


Chrome File Edit View History Bookmarks People Tab Window Help
Sat 6 Mar 19:28
Launch instance wiz
-group-projects
Setting up an FTP s
Grammarly
Inbox (573) - wgues
Associate a Static P
Elastic IP addresses
us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:
Apps
Mac Stuck On "Se...
PortableApps.com
Passwords
Apply to stay in th...
EU Exit: ID Docum...
Privacy error
How to Use Evern...
11 new notificatio...
Bullet Journal
aws
Services
Search for services, features, marketplace products, and docs [Option+S]
wguesdon
Ohio
Support
1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
Step 4: Add Storage
Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. Learn more about storage options in Amazon EC2.
Volume Type
Device
Snapshot
Size (GiB)
Volume Type
IOPS
Throughput (MB/s)
Delete on Termination
Encryption
Root /dev/sda1 snap-0c20bc49342315fa5 100 General Purpose SSD (gp2) 300 / 3000 N/A Not Encrypte
Add New Volume
Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. Learn more about free usage tier eligibility and usage restrictions.
Cancel
Previous
Review and Launch
Next: Add Tags
Feedback
English (US)
© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Cookie preferences




```
# see https://jayden-chua.medium.com/setting-up-an-ftp-server-on-ubuntu-18-04-on-aws-79bd55ab32bb
# Connect to the instance
ssh -i ~/Desktop/SHH_keys/AWS.pem ubuntu@ec2-3-140-217-89.us-east-2.compute.amazonaws.com

# 1. Install vsftpd
sudo apt-get update && sudo apt-get install vsftpd

# Check that the server is running. Exit with ctrl+c
sudo service vsftpd status

# 2. Configure firewall
sudo ufw allow OpenSSH
sudo ufw allow 20:21/tcp
sudo ufw allow 12000:12100/tcp
sudo ufw enable

# Check status
sudo ufw status
```

```
To                         Action      From
--                         ------      ----
OpenSSH                    ALLOW       Anywhere
20:21/tcp                  ALLOW       Anywhere
12000:12100/tcp            ALLOW       Anywhere
OpenSSH (v6)               ALLOW       Anywhere (v6)
20:21/tcp (v6)             ALLOW       Anywhere (v6)
12000:12100/tcp (v6)       ALLOW       Anywhere (v6)

# 3. Create User
sudo adduser ftpuser

# Deny the user SSH access
sudo nano /etc/ssh/sshd_config

# Add this line to the file
DenyUsers ftpuser

# Restart the SSH service
sudo service sshd restart

# Limit access rights
sudo mkdir /home/ftpuser/ftp

sudo chmod a-w /home/ftpuser/ftp
sudo mkdir /home/ftpuser/ftp/files
sudo chown ftpuser:ftpuser /home/ftpuser/ftp/files

# 5. Configure FTP server
sudo cp /etc/vsftpd.conf /etc/vsftpd.conf.bak
sudo nano /etc/vsftpd.conf
```

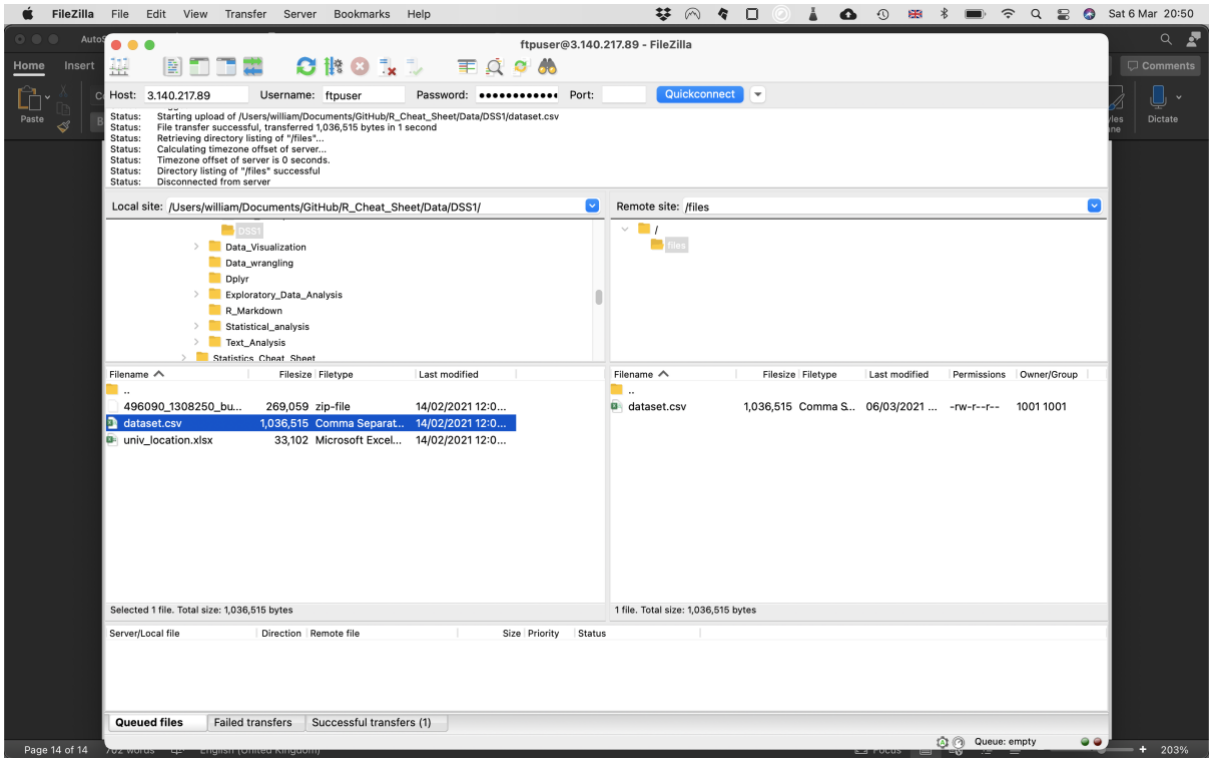



```
# Avoid error Server sent passive reply with un routable address. Using server address instead
listen=YES

listen_ipv6=NO
write_enable=YES
chroot_local_user=YES
local_umask=022
force_dot_files=YES
pasv_enable=YES
pasv_min_port=12000
pasv_max_port=12100
port_enable=YES

# Avoid error Server sent passive reply with unroutable address. Using server address instead
pasv_address=3.140.217.89
user_sub_token=$USER
local_root=/home/$USER/ftp

# Restart the server
sudo systemctl restart vsftpd
sudo service vsftpd status
```

```
● vsftpd.service - vsftpd FTP server
   Loaded: loaded (/lib/systemd/system/vsftpd.service; enabled; vendor preset:
e
   Active: active (running) since Sat 2021-03-06 20:41:12 UTC; 1min 21s ago
  Process: 24490 ExecStartPre=/bin/mkdir -p /var/run/vsftpd/empty
(code=exited,
 Main PID: 24498 (vsftpd)
    Tasks: 1 (limit: 4915)
   CGroup: /system.slice/vsftpd.service
           └─24498 /usr/sbin/vsftpd /etc/vsftpd.conf

Mar 06 20:41:12 ip-172-31-19-242 systemd[1]: Starting vsftpd FTP server...
Mar 06 20:41:12 ip-172-31-19-242 systemd[1]: Started vsftpd FTP server.
```

Connect to the server

3.140.217.89
ftpuser
pswd
21



9. Install git

See:

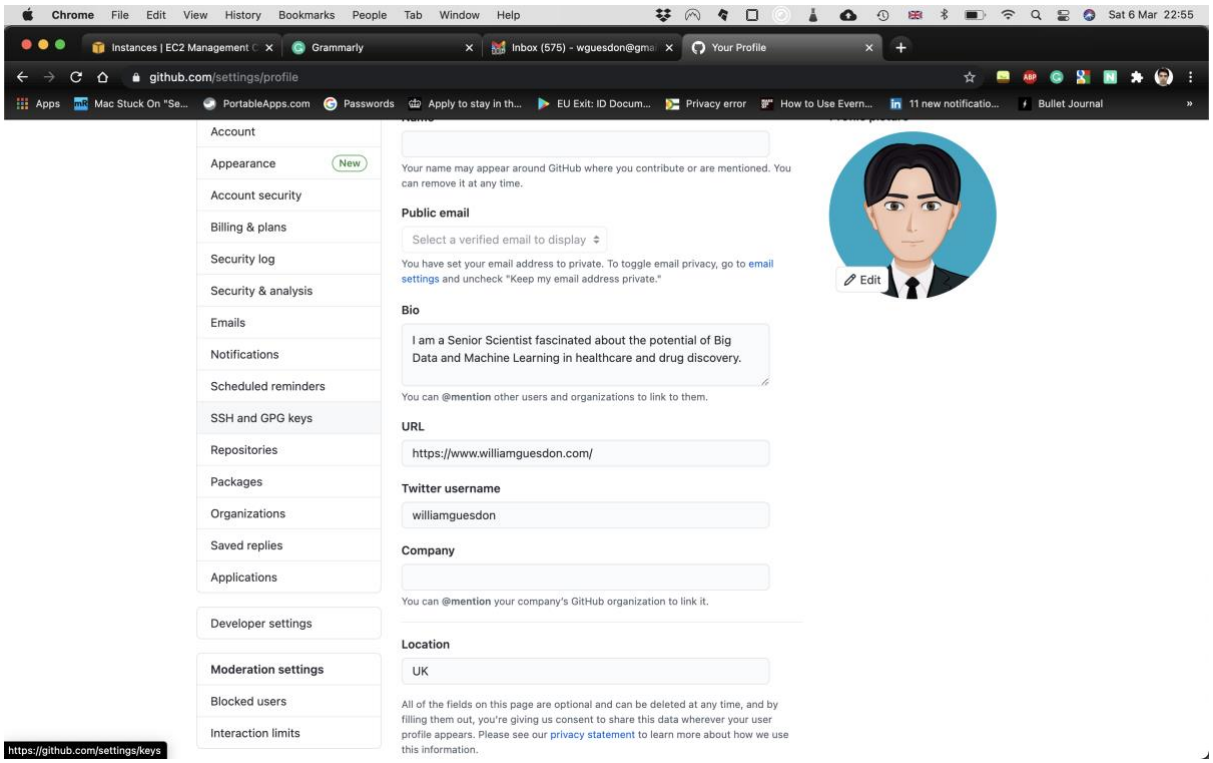
https://www.geeksforgeeks.org/how-to-install-configure-and-use-git-on-ubuntu/

https://www.digitalocean.com/community/tutorials/how-to-install-git-on-ubuntu-20-04

https://medium.com/@sangeethkumar.tvm.kpm/how-to-connect-your-github-repository-using-ssh-key-fcfbf0d62eb6

```
sudo apt install git

git config --global user.name "wguesdon"
git config --global user.email "wguesdon@gmail.com"
git config --list

ssh-keygen -t rsa -b 4096 -C "wguesdon@gmail.com"
cat ~/.ssh/id_rsa.pub
```

Add they Key to the account.



```
mkdir GitHub
cd GitHub
git clone git@github.com:wguesdon/Data_Science_portfolio.git

git clone git@github.com:66Days-group-learners/VinBigData_Chest_X_ray.git
```

10. References

1. https://medium.com/tensult/configure-ftp-on-aws-ec2-85b5b56b9c94
2. https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/WSL.html
3. https://towardsdatascience.com/stop-worrying-and-create-your-deep-learning-server-in-30-minutes-bb5bd956b8de
4. https://bloggingnectar.com/aws/automate-your-ec2-instance-setup-with-ec2-user-data-scripts/
5. https://stackoverflow.com/questions/28356796/aws-ec2-passive-ftp-server-sent-passive-reply-with-unroutable-address-using-s
6. see https://medium.com/google-cloud/set-up-anaconda-under-google-cloud-vm-on-windows-f71fc1064bd7
7. Automate Your EC2 Instance Setup with EC2 User Data Scripts
8. A Simple BASH Script For Ubuntu Server Post Installation
9. Transferring Files between your laptop and Amazon instance