AWS EC2 Guide

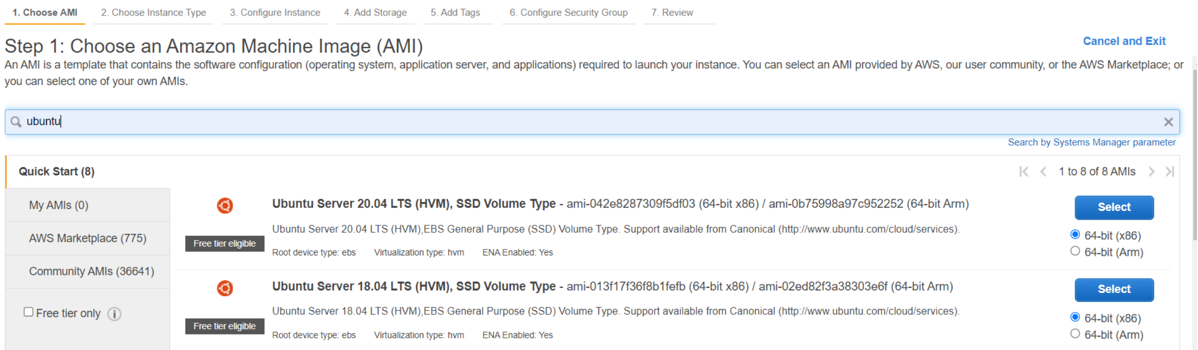
1. Connect to your aws account on url: https://www.awseducate.com/student/s/classrooms

2. Choose “**Big Data Platform**” class

3. Choose “**AWS Console**”

4. Under “**Services**”, go to “**EC2**”

    a. Choose **Instances** resource
    b. Click **Launch Instance**
    c. Choose Ubuntu 18.04 image



Step 1: Choose an Amazon Machine Image (AMI)

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.

ubuntu

Ubuntu Server 20.04 LTS (HVM), SSD Volume Type - ami-042e8287309f5df03 (64-bit x86) / ami-0b75998a97c952252 (64-bit Arm)

Free tier eligible

Ubuntu Server 20.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select ● 64-bit (x86) ○ 64-bit (Arm)

Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-013f17f36f8b1fefb (64-bit x86) / ami-02ed82f3a38303e6f (64-bit Arm)

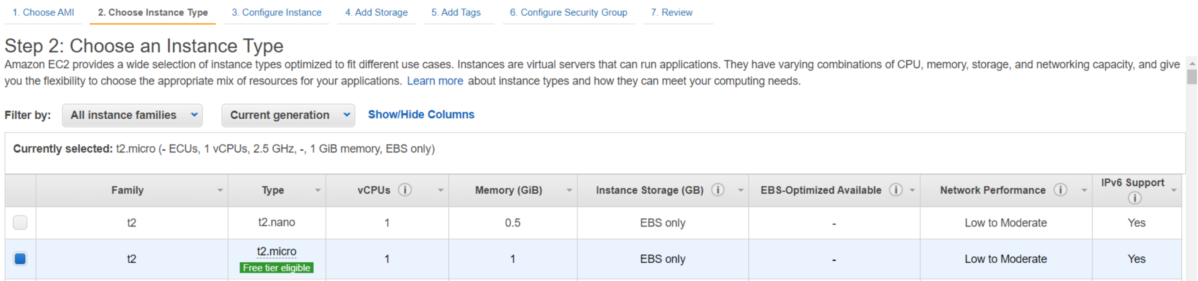
Free tier eligible

Ubuntu Server 18.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select ● 64-bit (x86) ○ 64-bit (Arm)

d. Choose t2.micro instance type and then click **Review and Launch**



Step 2: Choose an Instance Type

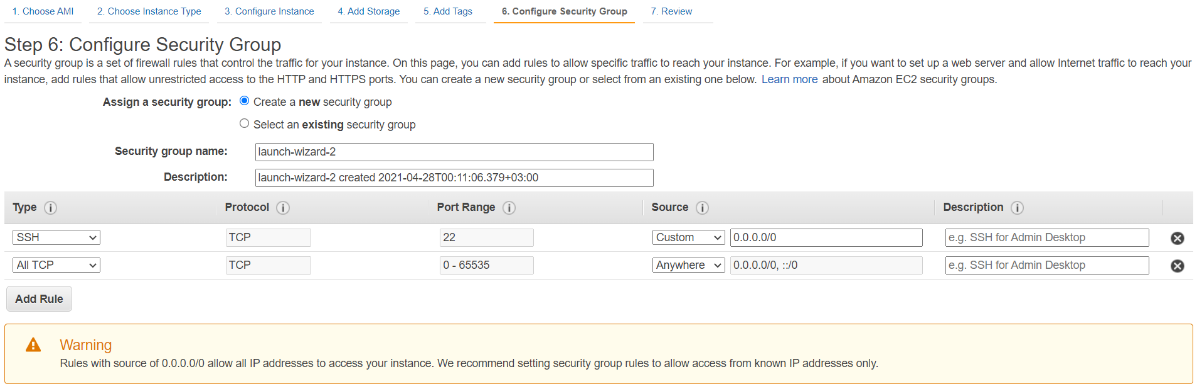
Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources 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.micro (- ECUs, 1 vCPUs, 2.5 GHz, -, 1 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

e. Choose “Configure Security Group” and add rule like we did in assignment 2



Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

Assign a security group: ● Create a **new** security group ○ Select an **existing** security group

Security group name: launch-wizard-2

Description: launch-wizard-2 created 2021-04-28T00:11:06.379+03:00

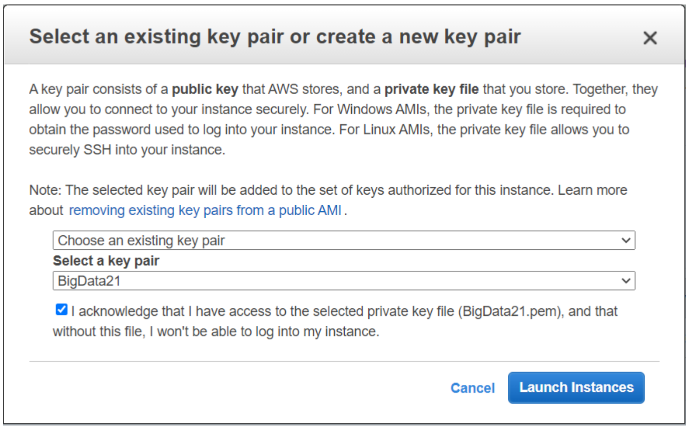
Type	Protocol	Port Range	Source		Description
SSH	TCP	22	Custom	0.0.0.0/0	e.g. SSH for Admin Desktop
All TCP	TCP	0 - 65535	Anywhere	0.0.0.0/0, ::/0	e.g. SSH for Admin Desktop

Add Rule

Warning

Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

e. Click **review and launch.** pay attention to select a key pair for connecting your virtual machine



Select an existing key pair or create a new key pair ×

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI .

Choose an existing key pair

Select a key pair

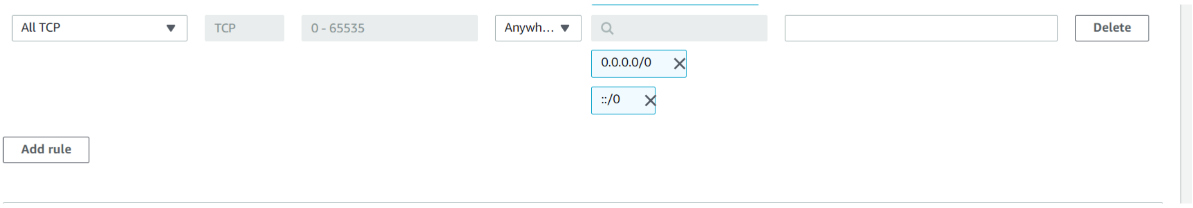
BigData21

☑ I acknowledge that I have access to the selected private key file (BigData21.pem), and that without this file, I won't be able to log into my instance.

5. Under “**Services**”, go to “**EC2**” and Choose **Instances** to see the details of

   a. Choose **Instances**

      i. Go to Advanced options if you want use spot instances
      ii. Wait until your cluster is in “Waiting” state (it takes about 10 minutes)
      iii. Scroll down the page and choose **“Security groups for Master”**
      iv. Choose “ElasticMapReduce-master” and click “edit Inbound Rules”
      v. Create new rule like that for allow jupyter-lab on your browser:



All TCP | TCP | 0 - 65535 | Anywh... | 0.0.0.0/0 ::/0

6. connect to your ec2 instance:

   a. Mac: ssh -i *my-key.pem ubuntu*@**Public-IPv4-address**
   b. Windows: connect with mobaXterm by create new session:
      i. host: **Public IPv4 address**
      ii. user: **ubuntu**
      iii. choose advanced options and choose your key pair, which you create in the section above

IMPORTANT: replace my-key.pem with your key, and Public IPv4 address to yours address

7. After connect the ec2 instance run the following commands:

   a. sudo apt update

b. *sudo apt install -y python3-pip python3-dev ipython3*

c. *pip3 install --upgrade pip*

d. *sudo python3 -m pip install ipykernel jupyterlab boto3 aws*

e. *jupyter lab --allow-root --ip=0.0.0.0 --no-browser*

i. You should get output like this:

http://127.0.0.1:8888
?token=ec0ad5afba127fadbdb2ed10ed763945d10283b0f4a68db5)

ii. open your computer browser and go to that address **by replace 127.0.0.1 with your *ec2 ip***