

Lab 7 – Walkthrough (EC2 / PuTTY / Jupyter Notebook / GitHub)

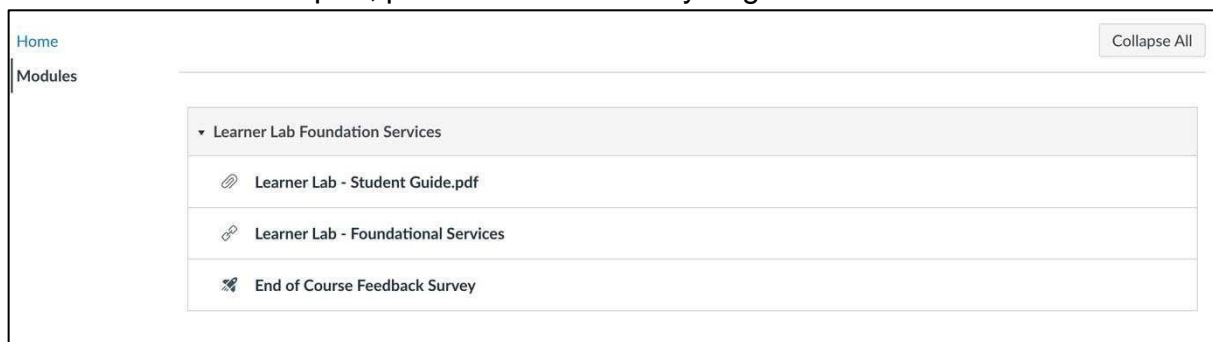
This document is a step-by-step walkthrough of the system configuration described in “Lab 7 - Introduction to GitHub, AWS, Jupyter, SSH, PySpark and Spark”

1. Canvas course acceptance

You will have received an invitation to join a Canvas course, click accept and follow the link to the course.

2. Configure and launch an EC2 instance

Go to Modules and click “Learner Lab – Foundational Services”. Note that the URL should not be <https://canvas.auckland.ac.nz/>, instead it should be <https://awsacademy.instructure.com/>. Also note that further instructions are available in the “Student Guide.pdf”, please refer to that if you get stuck.



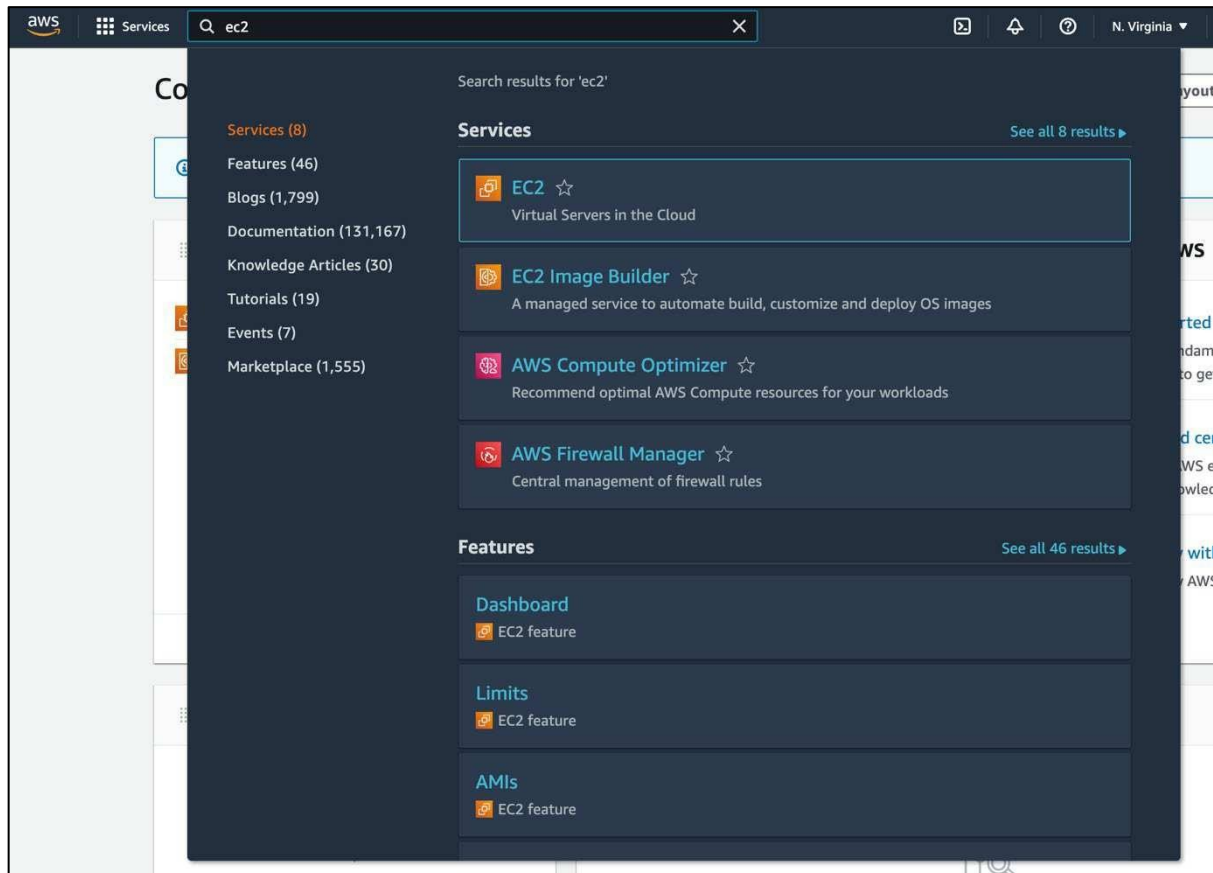
Click “Start Lab” in the top-right.



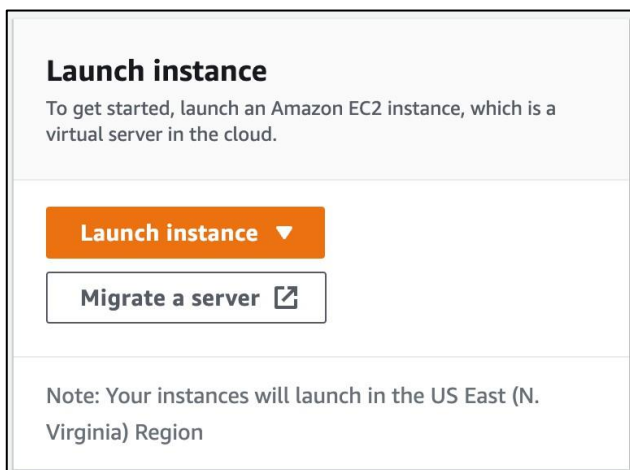
Once the lab has started, the icon next to “AWS” will turn green. Click “AWS” to go to your AWS Console.



In the AWS Console, search for EC2. Under “Services” click the first service highlighted in the blue box below. **Please note that your location in the top-right must be set to “N. Virginia” (i.e. North Virginia).**



Click “Launch instance”.



Name your instance anything you like. Then, under “Application and OS Images” click “Browse more AMIs”

Name and tags [Info](#)

Name

722

Add additional tags

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q Search our full catalog including 1000s of application and OS images

Recents

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

S

Q

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type

Free tier eligible

ami-05fa00d4c63e32376 (64-bit (x86)) / ami-05f3141013eebdc12 (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Click “Community AMIs” and search for “722-v3”. When you see the AMI shown below, click “Select” on the right.

Q 722-v3

X

▼

Quickstart AMIs (0)
Commonly used AMIs

My AMIs (0)
Created by me

AWS Marketplace AMIs (15)
AWS & trusted third-party AMIs

Community AMIs (1)
Published by anyone

Refine results

Clear all filters

▼ Operating system

▼ Linux/Unix

☐ All Linux/Unix

☐ Amazon Linux

☐ CentOS

☐ Debian

☐ Fedora

722-v3 (1 filtered, 1 unfiltered)

< 1 >

Community AMIs

Community AMIs contain all AMIs that are public, therefore anyone can publish an AMI and it will show in this catalog. This catalog can also contain paid products. When using community AMIs it is best practice to ensure you know and trust the publisher before launching an AMI.

aws

722-v3

ami-017ca83bf2e506f5e

INFOSYS 722 AWS image.

Platform: Other Linux Architecture: x86_64 Owner: 292308689702 Publish date: 2022-05-31 Root device type: ebs

Virtualization: hvm

ENA enabled: Yes

Select

The following results for "722-v3" were found in other categories

Property of The University of Auckland

Under “Instance type” change the selection to “t3.medium”. **Note that you only have \$100 credit, so do not change this to an instance that is more expensive than \$0.05 USD per hour otherwise you may run out of credit before your assignment due date.**

Under “Key pair” click “Create new key pair” if required and keep the downloaded file somewhere secure as you will need this file to access the server. Use the default settings.

▼ Instance type [Info](#)

Instance type

t3.medium

Family: t3 2 vCPU 4 GiB Memory

On-Demand Linux pricing: 0.0416 USD per Hour

On-Demand Windows pricing: 0.06 USD per Hour

▼

[Compare instance types](#)

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

722

▼

↻

[Create new key pair](#)

Under “Network settings” and “Inbound security group rules”, ensure “Type” is set to “All traffic” and “Source type” is set to “Anywhere”.

Inbound security groups rules

▼ Security group rule 1 (All, All, 0.0.0.0/0)

Remove

Type [Info](#)

Protocol [Info](#)

Port range [Info](#)

All traffic ▼

All

All

Source type [Info](#)

Source [Info](#)

Description - *optional* [Info](#)

Anywhere ▼

🔍 Add CIDR, prefix list or sec

0.0.0.0/0 ✕

e.g. SSH for admin desktop

Lastly, click “Launch instance” and “View instances”. It will take a few minutes, but eventually you will see that your instance is running as shown below. Note the **IPv4 Public IP** address for later (in this case, it is 18.205.7.167).

| <input checked="" type="checkbox"/> | Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone |
|-------------------------------------|------|---------------------|----------------------|---------------|--------------|--------------|-------------------|
| <input checked="" type="checkbox"/> | 722 | i-0f7a587028162c67e | Running | t3.medium | Initializing | No alarms + | us-east-1b |

Instance: i-0f7a587028162c67e (722)

Details

Security

Networking

Storage

Status checks

Monitoring

Tags

▼ Instance summary Info

Instance ID

i-0f7a587028162c67e (722)

IPv6 address

—

Hostname type

IP name: ip-172-31-7-31.ec2.internal

Answer private resource DNS name

IPv4 (A)

Public IPv4 address

18.205.7.167 | open address

Instance state

Running

Private IP DNS name (IPv4 only)

ip-172-31-7-31.ec2.internal

Instance type

t3.medium

Private IPv4 addresses

172.31.7.31

Public IPv4 DNS

ec2-18-205-7-167.compute-1.amazonaws.com | open address

Elastic IP addresses

—

3. Connecting to your instance

Right-click your instance and click “Connect”.

| <input checked="" type="checkbox"/> | Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone |
|-------------------------------------|------|---|----------------|---------------|-------------------|--------------|-------------------|
| <input checked="" type="checkbox"/> | 722 | i-0f7a587028162c67e | Running | t3.medium | 2/2 checks passed | No alarms | us-east-1b |
| | | <div>Launch instances</div> <div>Launch instance from template</div> <div>Migrate a server</div> <div>Connect</div> | | | | | |

Change the username to “ubuntu” and click “Connect” at the bottom-right as depicted below.

Connect to instance [Info](#)

Connect to your instance i-0f7a587028162c67e (722) using any of these options

EC2 Instance Connect

Session Manager

SSH client

EC2 serial console

Instance ID
i-0f7a587028162c67e (722)

Public IP address
18.205.7.167

User name

Connect using a custom user name, or use the default user name root for the AMI used to launch the instance.

Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

Cancel

Connect

Once the connection is established, you will see the screen below:

```
Welcome to Ubuntu 22.04 LTS (GNU/Linux 5.15.0-1008-aws x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/advantage

System information as of Thu Sep  8 03:10:37 UTC 2022

System load:  0.080078125      Processes:            118
Usage of /:   71.8% of 7.58GB   Users logged in:      0
Memory usage: 7%               IPv4 address for ens5: 172.31.7.31
Swap usage:   0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.

  https://ubuntu.com/aws/pro

0 updates can be applied immediately.

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
```

Click into this terminal and type “jupyter notebook”. You will see a unique URL at the bottom of the screen. Note that your URL will not be exactly the same as the URL in this screenshot. **Do not close this window/tab otherwise you will no longer be able to access Jupyter Notebook.**

However, please note the service we are using will pause AWS instances after four hours. You can reset the timer by clicking “Start Lab” on Canvas > Modules > Learner Lab – Foundational Services if you would like to use AWS for an extended period of time. To regain access after it has automatically shutdown, follow the same instructions, by clicking on “Start Lab” and waiting for the AWS icon to turn green. All of your materials will be saved on persistent storage but it is important that you backup your files using Git and GitHub, as details in Section 4.

```
ubuntu@ip-172-31-7-31:~$ jupyter notebook
[I 03:11:16.206 NotebookApp] Serving notebooks from local directory: /home/ubuntu
[I 03:11:16.206 NotebookApp] Jupyter Notebook 6.4.11 is running at:
[I 03:11:16.206 NotebookApp] https://ip-172-31-7-31:8888/?token=319d0281cf8094ecbde0fff99fffc28a9eed95c86ef640d3
[I 03:11:16.206 NotebookApp] or https://127.0.0.1:8888/?token=319d0281cf8094ecbde0fff99fffc28a9eed95c86ef640d3
[I 03:11:16.206 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 03:11:16.209 NotebookApp]

To access the notebook, open this file in a browser:
  file:///home/ubuntu/.local/share/jupyter/runtime/nbserver-2879-open.html
Or copy and paste one of these URLs:
  https://ip-172-31-7-31:8888/?token=319d0281cf8094ecbde0fff99fffc28a9eed95c86ef640d3
  or https://127.0.0.1:8888/?token=319d0281cf8094ecbde0fff99fffc28a9eed95c86ef640d3
```

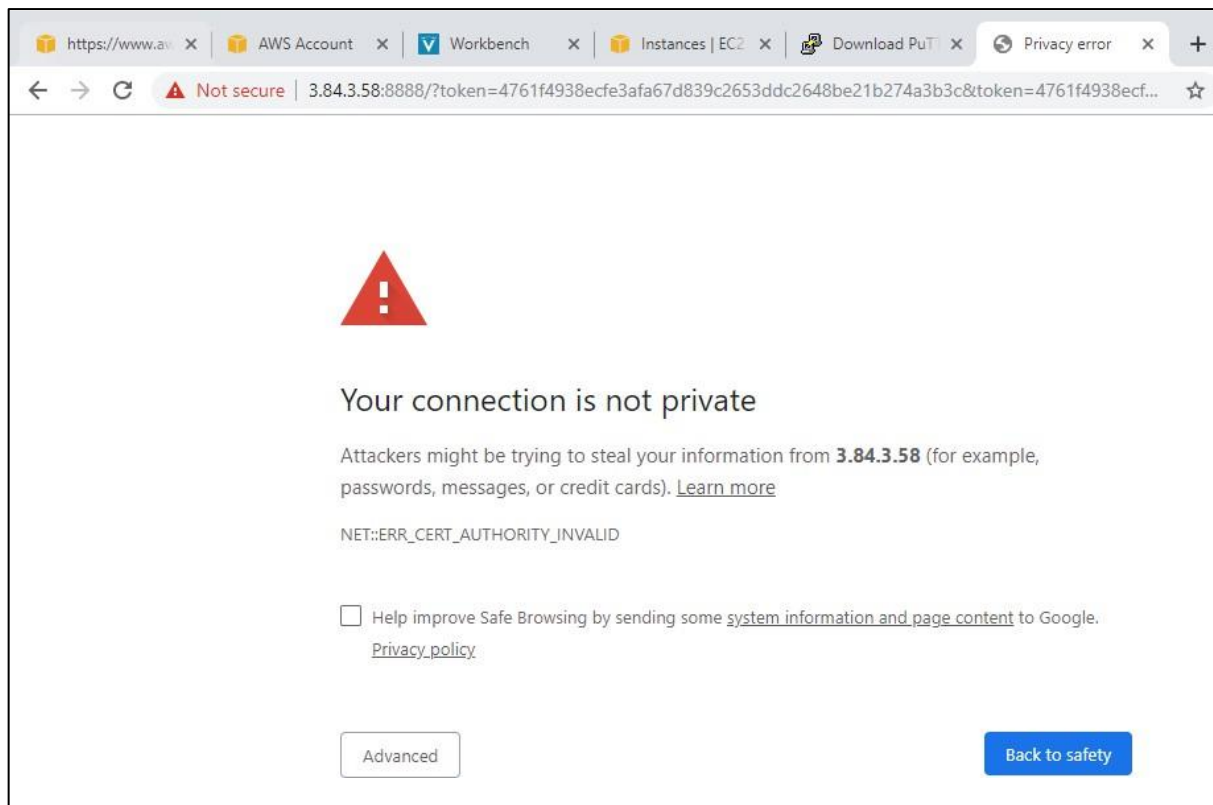
Jupyter Notebook is now running on an AWS instance in the cloud. To access, first copy and paste the Jupyter Notebook URL. Be careful not to copy both URLs by accident. Replace the highlighted part of the copied URL below with your public IP address (noted down previously).

`https://ip-172-31-7-31:8888/?token=319d0281cf8094ecbde0fff99ffc28a9eed95c86ef640d3`

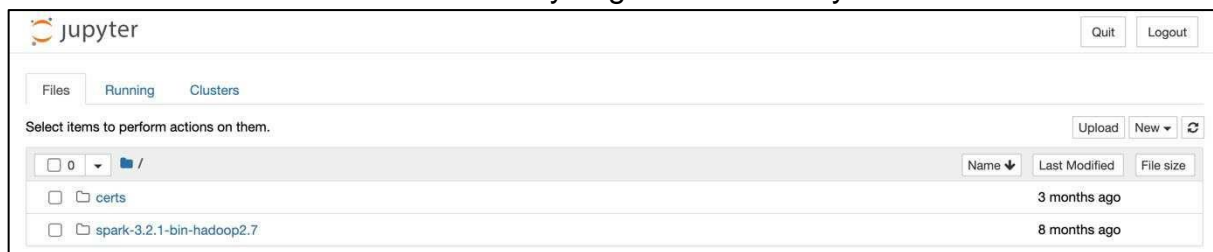
It should look similar to this (note that your public IP address will be different).

`https://18.205.7.167:8888/?token=319d0281cf8094ecbde0fff99ffc28a9eed95c86ef640d3`

Also note that if you are using Chrome, the following screen will pop up. To get past this screen, you can override this security measure by typing 'thisisunsafe'. Do not type this into the URL bar, click on the screen and begin typing. Other browsers should work fine.



If you do this correctly, the page should refresh and you will be able to see the Jupyter Notebook window below. You are ready to get started with your labs!



4. Working with GitHub

Git is used for version control while GitHub is a hosting service for Git repositories. It's useful when you're working with a team, but it's still a great tool when you're working by yourself. The lab machines as well as the EC2 instance you're working on have Git installed.

1. As an initial task, let's introduce you to some basic Git commands including git clone, git add, git commit, and git push. After creating an account on GitHub, follow along with [this video](#) created by the 722 team.
 - a. **Git Clone:** Git clone allows you to clone an already existing repository. Because Git encourages open-source, you have the option to clone code that already exists.
 - b. **Git Add, Commit and Push:** You need to type in all three of these commands to push your data from your local repository to GitHub. If you'd like some information on the differences between the three, click [here](#).
2. If you need more information, [try going through this tutorial](#).
3. Note that Git is the simplest way to have a backup of your code/outputs that you'll be developing on your AWS EC2 instance. **It is your duty to keep backups of your code and outputs.**

Now that you are comfortable with the basics of Git and GitHub, you can use the clone command to get access to the lab materials.

1. First, we need to request access to [Shohil's GitHub repository](#) as it is a private repository. To request access, fill out the spreadsheet [here](#). Alternatively, email your Course Coordinator, Ajay.
2. Once you've been given access (usually within 48 hours), login to your EC2 instance and type in the commands below.
 - a. Type "mkdir 722" to create a new folder named 722.
 - b. Then, "cd 722" to move into that folder.
 - c. Then, type "git clone <https://github.com/shohil-kishore/aws-instance>" to clone Shohil's instance.
 - d. Note that GitHub will ask you to input your username and password at this stage. Input your username as is, but for the password, you need to generate a personal access token. [Detailed instructions can be found here](#).
 - e. If there's an error, please try again. If it still doesn't work, please screenshot the error and send it to your Course Coordinator, Mina.

Note that it is best practice to back up your files using GitHub regularly, especially as all AWS instances shutdown after four hours automatically.