27th April 2017

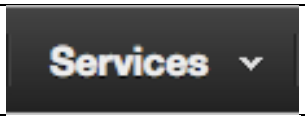
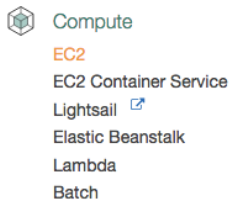
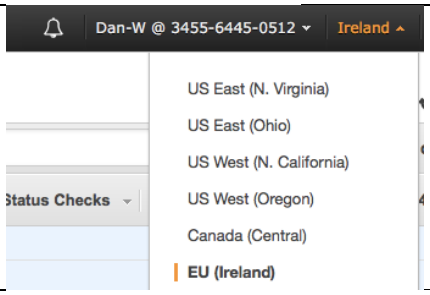
Training Materials

Log in here using the username and password provided in the workshop

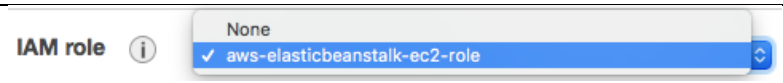
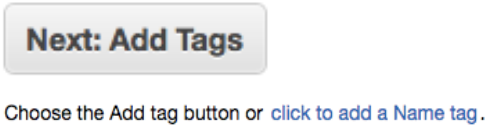
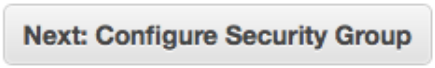
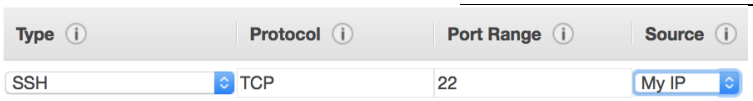
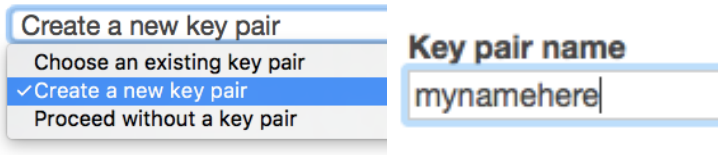
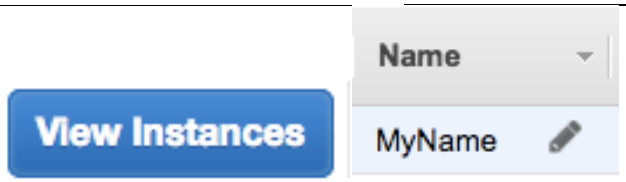
https://345564450512.signin.aws.amazon.com/console

To change your Identity and Access Management (IAM) password, click the drop down by your username, then My Security Credentials.

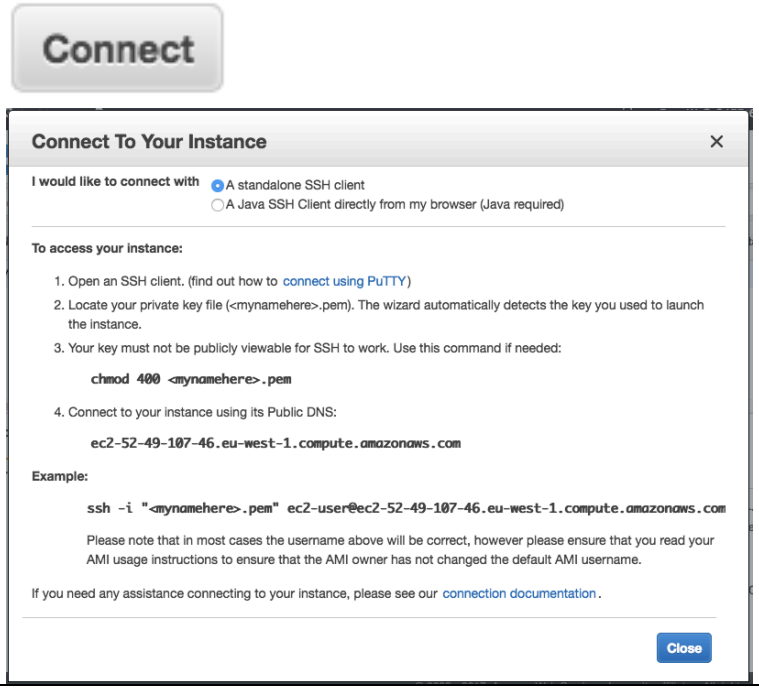
Connecting to EC2

Go to the Services Drop-Down	
Select EC2	
Make sure you are in the EU-West region (Ireland) using the drop-down in the top right corner of the screen.	
Under Create Instance select Launch Instance	
Create Amazon Linux AMI (Free Tier Eligible)	
Select General Purpose t2.micro	
Configure instance details	

27th April 2017

Select the aws-elasticbeanstalk-ec2-role. Roles encapsulate permissions (the ability to access, view or modify parameters) within AWS. They can be applied to EC2 instances, AWS users or groups of users.	
Add storage – defaults are fine.	
Add Tags Add a name tag with your name, for identification of your instance.	
Move on to security groups	
Edit Security groups - Allow SSH, Source: My IP	
Review and Launch	
Launch	
Create a new Key Pair, give it your name, press Download and save to your machine (mynamehere.pem)	
Save this keypair somewhere easy to get to. We suggest a new folder in your home directory	$ mkdir Key $ mv Downloads/mynamehere.pem Key/mynamehere.pem
Launch your instance	
Click View Instances. Lots of instances will be launching at the same time – use the name tags to find yours.	

27th April 2017

click Connect and a window should open	
on Mac/Linux, open terminal navigate to your key folder	$ cd Key
make your private key private by modifying its permissions.	$ chmod 400 Name.pem
to avoid having to type the long name over and over, copy it from the connection window set a variable in bash.	Example: ssh -i "mynamehere.pem" ec2-user@ec2-52-209-161-135.eu-west-1.compute.amazonaws.com $ instance=ec2-user@ec2-ip-ip-ip.ip.location.compute. amazonaws.com refer to this variable later using $instance
Connect to your instance and type yes to allow the connection.	$ ssh -i "Name.pem" $instance Are you sure you want to continue connecting (yes/no)? yes
You should now be on the machine. Terminal should show:	[ec2-user@ip-ip-ip-ip ~]$

27th April 2017

Transferring files from an Amazon S3 Bucket to your instance

Create a text file in your home directory and use a new terminal window to transfer it to your instance using scp.	$ scp –i Key/myname.pem newFile.txt $instance:~
As an alternative, we can transfer files from an Amazon S3 bucket to the instance using Amazon's s3 copy command. Our bucket is called "ngcm1"	[ec2-user@ip-ip-ip-ip ~]$ aws s3 ls s3://ngcm1 [ec2-user@ip-ip-ip-ip ~]$ aws s3 cp s3://ngcm1/testFile.txt
in your terminal window for the instance, check that it's been transferred.	[ec2-user@ip-ip-ip-ip ~]$ ls
Code can be run on the instance in the same way Transfer "sim.py" to the instance from the S3 bucket and run it	[ec2-user@ip-ip-ip-ip ~]$ python sim.py
Use scp to transfer the output of the simulation out.csv onto your local machine.	$ scp –i Key/myname.pem $instance:~/out.csv out.csv
The Amazon Linux AMI uses yum as its preferred package manager. As you have root access to the instance you can install any supported packages, such as gcc.	[ec2-user@ip-ip-ip-ip ~]$ sudo yum install <package>

27th April 2017

A very simple web hosting example with EC2…

We are going to repurpose our running instances to host a static webpage. The free tier allows 24/7 running of a single t2.micro amazon instance for 12 months (750 hours / month free).

Step by step guidance is not provided for this section. We are here to help, the detailed instructions above should give you a good start, and AWS has many rollover information points and comprehensive documentation.

- In the management console, set up your instance to allow access from internet traffic over port 8000.
- Transfer the entire contents of the Web subdirectory on the ngcm1 S3 bucket onto the instance. (hint: aws s3 help)
- navigate to this folder on the instance and launch a web server
  $ python -m SimpleHTTPServer 8000 &
- In a web browser, navigate to the instance's public URL, and specify port 8000 to view your website.

If you like the look of Amazon Web Services and would like to explore the further capabilities of the platform, you can apply for a student account which includes $40 of free credit.

https://aws.amazon.com/education/awseducate/

Apply for a student account, follow the prompts and explore Amazon Web Services for yourself!