

Pre-Course Exercise 2

Start up an instance on Amazon EC2 and get Apache web server running

Prior Knowledge

Unix Command Line Shell

Learning Objectives

Understand about EC2 instances

Start an instance using the web interface

Configure the AWS command line

Manage instances from a command line

Understand Security Groups

Software Requirements

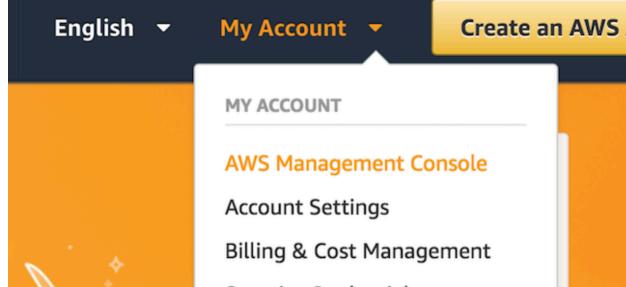
(see separate document for installation of these)

- AWS CLI

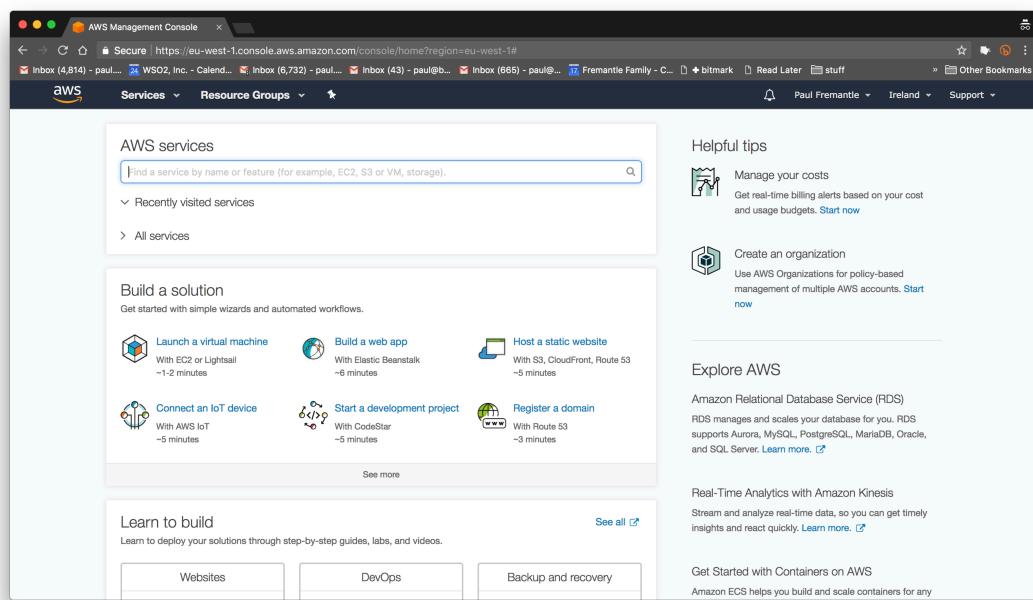
Part A: Starting an Instance from the Web Console.

1. You have been provided with an Ubuntu VM. Start that up.
2. The course is also providing time and resources on the Amazon AWS/EC2 cloud for the duration of the course.

3. Open up a browser window and navigate to <https://aws.amazon.com/>
4. Click on the menu item My Account-> AWS Management Console



5. Log in with your credentials
6. You should see a screen like this:



7. In the top right corner click on Oregon and change to **EU (Ireland)** (unless it is already on Ireland!)

8. Expand All Services and click on the link EC2

The screenshot shows the AWS EC2 Dashboard. On the left, there's a sidebar with links like 'Events', 'Tags', 'Reports', 'Limits', 'INSTANCES', 'Instances', 'Spot Requests', 'Reserved Instances', 'Commands', 'IMAGES', 'AMIs', 'Bundle Tasks', 'ELASTIC BLOCK STORE', 'Volumes', 'Snapshots', 'NETWORK & SECURITY', 'Security Groups', 'Elastic IPs', 'Placement Groups', and 'Key Pairs'. The main area is titled 'Resources' and shows resource counts: 0 Running Instances, 1 Volumes, 1 Key Pairs, 0 Placement Groups, 0 Elastic IPs, 2 Snapshots, 0 Load Balancers, and 2 Security Groups. Below this, there's a 'Create Instance' section with a 'Launch Instance' button. To the right, there's an 'Account Attributes' section with links for supported platforms (VPC), default VPC (vpc-42fb9527), and additional information like getting started guides, documentation, forums, pricing, and contact us. There's also an 'AWS Marketplace' section.

9. Click on the blue button: Launch Instance

10. Choose “Ubuntu Server 16.04 LTS (HVM), SSD Volume Type”

The screenshot shows the AWS Lambda console. It displays the AMI details for 'Ubuntu Server 16.04 LTS (HVM), SSD Volume Type - ami-6d48500b'. It includes a 'Select' button, a 'Free tier eligible' badge, and a note about Canonical support. Below this, it shows 'Root device type: ebs' and 'Virtualization type: hvm'.

11. Choose the instance type t2.micro.

12. Click Next: Configure Instance Details

Next: Configure Instance Details

13. Click Next: Add Storage

14. Click **Next: Add Tags**

15. Now click: **Next: Configure Security Group**

16. Change the name of the security group to **simple**

Assign a security group: Create a new security group
 Select an existing security group

Security group name: simple

Description: launch-wizard-29 created 2017-11-24T12:44:48.839+00:00

| Type <i>i</i> | Protocol <i>i</i> | Port Range <i>i</i> | Source <i>i</i> |
|---------------|-------------------|---------------------|---------------------------|
| SSH | TCP | 22 | Custom <i>▼</i> 0.0.0.0/0 |

Add Rule

Hint: There is a security warning about the security rule. The default rule allows Secure Shell (SSH) access from any IP address. If you know your company or personal internet connection comes from a specific IP address you can improve security by restricting to that.

Note this is NOT the IP address you get by looking at the local machine's configuration, but the publicly visible IP address that the Amazon cloud sees from you. You can see what your IP is by typing "what's my IP" into Google.

However, I am not sure if the current network sends messages from different IPs or the same and therefore we will leave this as-is despite the warning.

17. Click Review and Launch

You should see something very like this:

AMI Details

Ubuntu Server 14.04 LTS (HVM), SSD Volume Type - ami-47a23a30
Free tier eligible
Ubuntu Server 14.04 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

Instance Type

| Instance Type | ECUs | vCPUs | Memory (GiB) | Instance Storage (GB) | EBS-Optimized Available | Network Performance |
|---------------|----------|-------|--------------|-----------------------|-------------------------|---------------------|
| t2.micro | Variable | 1 | 1 | EBS only | - | Low to Moderate |

Security Groups

Security group name: oxclo02
Description: launch-wizard-1 created 2015-11-16T09:27:30.852+00:00

| Type | Protocol | Port Range | Source |
|------|----------|------------|-----------|
| SSH | TCP | 22 | 0.0.0.0/0 |

Tags

| Key | Value |
|------|---------|
| Name | oxclo02 |

Buttons: Edit AMI, Edit instance type, Edit security groups, Edit instance details, Edit storage, Edit tags, Cancel, Previous, Launch

18. Click Launch

19. You will be prompted with a new window to decide on the correct key pair to secure this instance with. Since this is the first time you are using EC2, you need to create a key pair. Change the dropdown box to **Create a new key pair**.

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

Create a new key pair

Key pair name
bigkp

Download Key Pair

You have to download the **private key file (*.pem file)** before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

Cancel Launch Instances

20. Use **bigkp** as the name of the keypair.

21. Click **Download Key Pair**. This will save a file to your ~/Downloads directory.

22. Click **Launch Instances**

You should see something like:

The screenshot shows the AWS Launch Status page. At the top, a green box says "Your instances are now launching" with a checkmark icon. Below it, a message says "The following instance launches have been initiated: i-a475401d" and a link to "View launch log". A blue box below says "Get notified of estimated charges" with a speech bubble icon, and a link to "Create billing alerts". Under "How to connect to your instances", it says "Your instances are launching, and it may take a few minutes until they are in the running state, when they will be ready for you to use. Usage hours on your new instances will start immediately after they are launched." It also says "Click View Instances to monitor your instances' status. Once your instances are in the running state, you can connect to them from the Instances screen. Find out how to connect to your instances." A section titled "Here are some helpful resources to get you started" lists links to "How to connect to your Linux instance", "Amazon EC2: User Guide", "Learn about AWS Free Usage Tier", and "Amazon EC2: Discussion Forum".

23. Click on the blue instance ID link (e.g. **i-a475401d** in the screenshot above)

You will see a dashboard like:

The screenshot shows the AWS Instances dashboard. At the top, there are buttons for "Launch Instance", "Connect", and "Actions". Below is a search bar with "search : i-0fa3d4032833ea933" and a "Add filter" button. The main table has columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, Public DNS (IPv4), and IPv4. One row is shown: "i-0fa3d4032833ea933", "t2.micro", "eu-west-1c", "pending", "Initializing", "None", "ec2-54-154-120-147.eu-west-1.compute.amazonaws.com", "54.120.147.120". Below the table, a detailed view for "Instance: i-0fa3d4032833ea933" shows "Public DNS: ec2-54-154-120-147.eu-west-1.compute.amazonaws.com". It has tabs for "Description", "Status Checks", "Monitoring", and "Tags". Under "Description", it shows "Instance ID: i-0fa3d4032833ea933", "Instance state: pending", and "Instance type: t2.micro". Under "Status Checks", it shows "Public DNS (IPv4): ec2-54-154-120-147.eu-west-1.compute.amazonaws.com", "IPv4 Public IP: 54.154.120.147", and "IPv6 IPs: -".

24. Make sure you are running the Ubuntu VM, and start a fresh terminal window (Ctrl-Alt-T, or find Terminal graphically)

25. Check if there is already a ~/keys directory.

If not, then make a directory to store your private key:
`mkdir ~/keys`

26. Copy your private key to the new directory:
`cp ~/Downloads/bigkp.pem ~/keys/`

27. Before you can use the key you need to change the permissions on it.
Type:

```
chmod 400 ~/keys/bigkp.pem
```

28. Check to see if the status checks on your instance are now complete.
 Refresh the browser window:

| Instance State | Status Checks | Alarm Status | Public DNS | Public IP |
|--|---|--------------|--|--------------|
| ● running | ✓ 2/2 checks ... | None | ⚠ ec2-52-30-233-95.eu-w... | 52.30.233.95 |

29. Copy the Public IP Address from the browser window (e.g. 52.30.233.95 in my case)

30. Try to SSH into the machine. Replace your key file name and the IP address below!

```
ssh -i ~/keys/bigkp.pem ubuntu@ww.xx.yy.zz
```

31. As this is the first time you are accessing this host, the key on the server side is not known. You should see something like:

```
The authenticity of host '52.30.233.95 (52.30.233.95)' can't be
established.
ECDSA key fingerprint is
SHA256:7GhOakN9Pj3vWAegV0uYhPVI9qqVEe9RlNM0wcut01E.
Are you sure you want to continue connecting (yes/no)?
```

Type **yes** and hit Enter.

You will see something like:

```
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-1020-aws x86_64)
```

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

```
Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud
```

```
0 packages can be updated.
0 updates are security updates.
```

```
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.
```

```
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```

32. Congratulations – you have a cloud instance running.

PART B – Using the AWS Command Line to terminate the instance

33. The AWS Command Line (AWS CLI) is available as part of the Python PIP installed code. PIP is a package manager for Python.

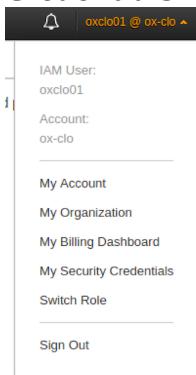
34. In a fresh Ubuntu Terminal Window (*make sure you are not doing this on your cloud server by mistake!*)

35. Now you can configure the AWS command line with your credentials

36. First you need to create an Access Key and Secret Key.

37. Go to the AWS Console

38. In the top right corner, click on your username, then choose **My Security Credentials:**



39. You will be warned as follows.

Choose Continue to Security Credentials.

You are accessing the security credentials page for your AWS account. The account credentials provide unlimited access to your AWS resources.

To help secure your account, follow an [AWS best practice](#) by creating and using AWS Identity and Access Management (IAM) users with limited permissions.

[Continue to Security Credentials](#)

[Get Started with IAM Users](#)

Don't show me this message again

40. You should see:

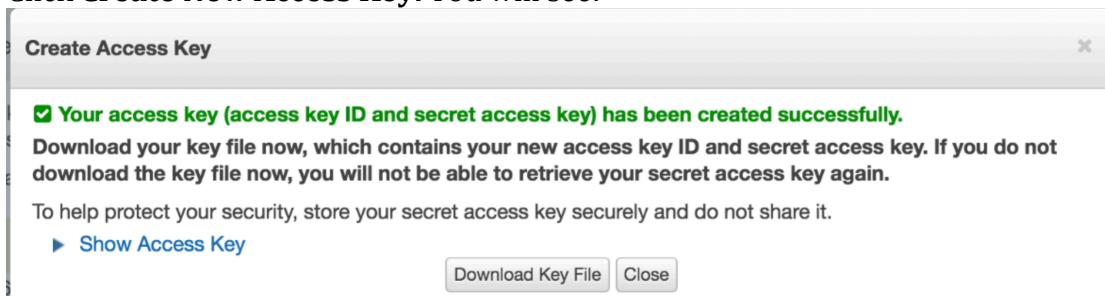
Your Security Credentials

Use this page to manage the credentials for your AWS account. To manage credentials for AWS Identity
To learn more about the types of AWS credentials and how they're used, see [AWS Security Credentials](#)

- + Password
- + Multi-factor authentication (MFA)
- + Access keys (access key ID and secret access key)
- + CloudFront key pairs
- + X.509 certificate
- + Account identifiers

41. Expand AccessKeys

42. Click **Create New Access Key**. You will see:



43. Click **Download Key File**

It should download a file called **rootkey.csv**

44. **You need to make a note of these credentials or download them, because the secret key will not be available again.**

45. Now we can use these keys to configure the AWS CLI. In a terminal window type:

```
aws configure
```

- a. When prompted
AWS Access Key ID [None]:

Type the Access Key ID from the text file or CSV (cut and paste)

- b. Do the same for the Secret Access Key.
- c. For the region choose Ireland: **eu-west-1**
- d. For the output format, type **json**

Hint: You now have three credentials for AWS:

- Your userid/password
- An Access Key/Secret Key for controlling EC2/AWS through command line, third-party tools and apps, and any Web Service APIs
- An SSH Private Key pair for accessing the actual instances that you startup.

46. Now let's use the CLI to terminate your instance.

47. From the AWS Web-based console, go back to the EC2 page, and then choose Running Instances. Find your running EC2 instance and find the id of your running instance:

The screenshot shows the AWS EC2 Instances page. The top navigation bar includes the AWS logo, Services dropdown, and Resource Groups dropdown. The main interface has a sidebar on the left with options like EC2 Dashboard, Events, Tags, Reports, Limits, and a collapsed Instances section containing Instances, Spot Requests, Reserved Instances, and Scheduled Instances. The Instances section is currently selected, indicated by an orange border. The main content area displays a table of running instances. The first instance in the table is highlighted with a red box around its Instance ID column, which contains the value "i-05046e663deb9fa13". The table also includes columns for Name, Instance ID, Instance Type (t2.micro), Availability Zone (eu-west-1b), and Status (running).

48. Now use the AWS CLI to terminate:
Replacing the instance ID with your own, type:

```
aws ec2 terminate-instances --instance-ids i-05046e663deb9fa13
```

49. You should see log like:

```
aws ec2 terminate-instances --instance-ids i-0fa3d4032833ea933
{
    "TerminatingInstances": [
        {
            "InstanceId": "i-0fa3d4032833ea933",
            "CurrentState": {
                "Code": 32,
                "Name": "shutting-down"
            },
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
}
```

50. Your SSH session to the server will die, and the server will no longer be running.

51. **Congratulations!** You have completed both of this exercise.