

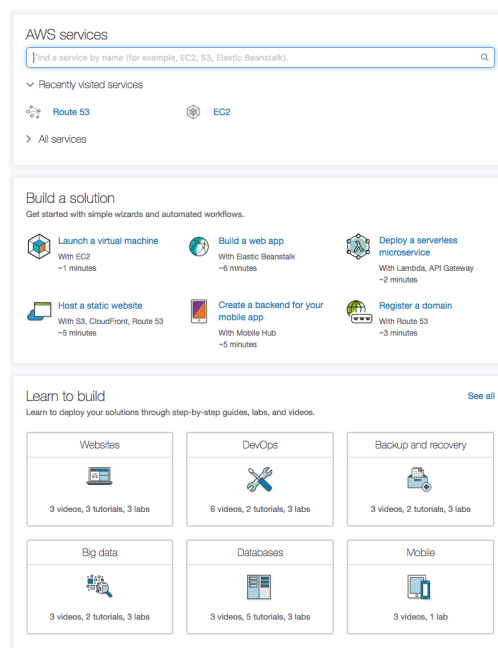
Starting and Provisioning an EC2 Instance

Aim

This course will be taught using Jupyter [1] notebooks hosted on an Amazon Web Services (AWS) Elastic Cloud Compute (EC2) instance. The aim of these instructions is to: Start an EC2 instance with the correct access permissions and use the key provided to log in, run some pre-defined scripts to provision the EC2 instance with Anaconda Python [2] and the tools required for the course and, finally, to start the Jupyter notebook server and connect to it using a web browser.

First: Starting an EC2 instance

1. Point your browser at <https://aws.amazon.com/> and click "sign into the console" in the top right hand corner.
- 2) Log in with the credentials you generated using the pre-course instructions, once authenticated this will take you to the console which will look *similar* to this:



- 3) Type "ec2" into the text box under "AWS services" and click the first option in the drop down box. This will land you on a page that looks similar to this (of course you will not have existing snapshots etc...):

EC2 Dashboard

- Events
- Tags
- Reports
- Limits
- INSTANCES
 - Instances
 - Spot Requests
 - Reserved Instances
 - Scheduled Instances
 - Dedicated Hosts
- IMAGES
 - AMIs
 - Bundle Tasks
- ELASTIC BLOCK STORE
 - Volumes
 - Snapshots
- NETWORK & SECURITY
 - Security Groups
 - Elastic IPs
 - Placement Groups
 - Key Pairs
 - Network Interfaces
- LOAD BALANCING
 - Load Balancers
 - Target Groups
- AUTO SCALING
 - Launch Configurations
 - Auto Scaling Groups
- SYSTEMS MANAGER
 - SERVICES
 - Run Command
 - State Manager
 - Automations
 - Patch Baselines
 - SHARED RESOURCES
 - Managed Instances
 - Activations

Resources

You are using the following Amazon EC2 resources in the US East (N. Virginia) region:

- 0 Running Instances
- 0 Elastic IPs
- 0 Dedicated Hosts
- 2 Snapshots
- 6 Volumes
- 0 Load Balancers
- 6 Key Pairs
- 7 Security Groups
- 0 Placement Groups

Just need a simple virtual private server? Get everything you need to jumpstart your project - compute, storage, and networking - for a low, predictable price. Try Amazon Lightsail for free.

Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

[Launch Instance](#)

Note: Your instances will launch in the US East (N. Virginia) region

Service Health

Service Status:

- US East (N. Virginia): This service is operating normally

Availability Zone Status:

- us-east-1a: Availability zone is operating normally
- us-east-1c: Availability zone is operating normally
- us-east-1d: Availability zone is operating normally
- us-east-1e: Availability zone is operating normally

[Service Health Dashboard](#)

Scheduled Events

US East (N. Virginia): No events

3. If it is not expanded out already (ie there is a "+" sign) click on "Instances" and in the menu below it click on "Instances" and you will be greeted by this screen:

[Launch Instance](#) [Connect](#) [Actions](#)

Filter by tags and attributes or search by keyword

You do not have any running instances in this region.

First time using EC2? Check out the [Getting Started Guide](#).

Click the [Launch Instance](#) button to start your own server.

[Launch Instance](#)

You can now click on "Launch Instance" on either blue and white button.

4. You will be greeted by a page that looks like this:

Quick Start

My AMIs

AWS Marketplace

Community AMIs

☐ Free tier only

Amazon Linux AMI 2016.09.1 (HVM), SSD Volume Type - ami-9be6f38c

The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.

Root device type: ebs Virtualization type: hvm

Red Hat Enterprise Linux 7.3 (HVM), SSD Volume Type - ami-b63769a1

Red Hat Enterprise Linux version 7.3 (HVM), EBS General Purpose (SSD) Volume Type

Root device type: ebs Virtualization type: hvm

SUSE Linux Enterprise Server 12 SP2 (HVM), SSD Volume Type - ami-fde4e6ea

SUSE Linux Enterprise Server 12 Service Pack 2 (HVM), EBS General Purpose (SSD) Volume Type. Public Cloud, Advanced Systems Management, Web and Scripting, and Legacy modules enabled.

Root device type: ebs Virtualization type: hvm

Ubuntu Server 16.04 LTS (HVM), SSD Volume Type - ami-e13739f6

Ubuntu Server 16.04 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

click select next to the "Ubuntu Server 16.04 LTS (HVM), SSD Volume Type" option. The next page asks you to click a radio button next to the "size" of the machine you want to start. **While you are**

experimenting use "t2.micro" which gives you a 1GB 1CPU EC2 instance. **For the course** 1GB will not cut it, select "m4.large". The on demand pricing [3] is 10.8 cents an hour.

Currently selected: m4.large (6.5 ECUs, 2 vCPUs, 2.4 GHz, Intel Xeon E5-2676v3, 8 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate
<input type="checkbox"/>	General purpose	t2.micro	1	1	EBS only	-	Low to Moderate
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate
<input type="checkbox"/>	General purpose	t2.large	2	8	EBS only	-	Low to Moderate
<input type="checkbox"/>	General purpose	t2.xlarge	4	16	EBS only	-	Moderate
<input type="checkbox"/>	General purpose	t2.xlarge	8	32	EBS only	-	Moderate
<input checked="" type="checkbox"/>	General purpose	m4.large	2	8	EBS only	Yes	Moderate
<input type="checkbox"/>	General purpose	m4.xlarge	4	16	EBS only	Yes	High
<input type="checkbox"/>	General purpose	m4.2xlarge	8	32	EBS only	Yes	High
<input type="checkbox"/>	General purpose	m4.4xlarge	16	64	EBS only	Yes	High
<input type="checkbox"/>	General purpose	m4.10xlarge	40	160	EBS only	Yes	10 Gigabit
<input type="checkbox"/>	General purpose	m4.16xlarge	64	256	EBS only	Yes	20 Gigabit
<input type="checkbox"/>	General purpose	m3.medium	1	3.75	1 x 4 (SSD)	-	Moderate
<input type="checkbox"/>	General purpose	m3.large	2	7.5	1 x 32 (SSD)	-	Moderate
<input type="checkbox"/>	General purpose	m3.xlarge	4	15	2 x 40 (SSD)	Yes	High
<input type="checkbox"/>	General purpose	m3.2xlarge	8	30	2 x 80 (SSD)	Yes	High

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Instance Details](#)

Once your selection has been made click the **gray** button "Next: Configure Instance Details".

5) The next page can be left as all defaults... Click the **gray** button "Next: Add Storage"

6) Again, this can be left as all defaults. Click the **gray** button "Next: Add Tags"

7) *Again*, no need to add tags for the purposes of this course... Once more click the **gray** button "Next: configure Security Group"

8) Now we have work to do! We need to configure our instance to be able to serve the Jupyter notebook via HTTPS on port 8888. You will see a page similar to that below. Make sure "Create a new security group" is checked, later you can select "select and existing group" to save you time! Enter a simple name for the Security group name like "j_sever". Enter something descriptive for the Description.

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

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:

Description:

Type	Protocol	Port Range	Source
SSH	TCP	22	Custom 0.0.0.0/0

[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.

We need to configure to allow connections from any IP to ports 443 (HTTPS) and 8888 (where Jupyter listens). Click "Add Rule". This will create a new row. On the new row the drop down box on the left will default to "Custom TCP Rule". Click it and select "HTTPS". Then click "Add Rule" again but this time, in the new row leave the drop down on "Custom TCP Rule". In that same, third, row enter "8888" into "Port Range" column and "0.0.0.0/0" in the source column. Once done it should look like this:

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

Security group name:

l_server

Description:

Server config for Jupyter Notebooks

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	
SSH	TCP	22	Custom 0.0.0.0/0	✕
HTTPS	TCP	443	Custom 0.0.0.0/0	✕
Custom TCP Rule	TCP	8888	Custom 0.0.0.0/0	✕

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.

Cancel

Previous

Review and Launch

Now we can click the **blue** button "Review and Launch".

[1]

[2]

[3] <https://aws.amazon.com/ec2/pricing/on-demand/>