Distributed Computing, Grids and Clouds

Julian Bunn, 2015

INTRODUCTION TO CLOUD COMPUTING

Distributed Computing

- Run an application on one or more computers connected in a network
- Computers are autonomous and possibly heterogeneous
- Target problem is subdivided into separate tasks
- Computers communicate via messages
- cf Parallel Computing (tightly coupled Distributed Computing)

Grid Computing

- “Distributed Computing with Authentication” – certificates etc.
- Supported by e.g. Globus “toolkit”
- No central control
- “Virtual Organizations”
- Typically used in research and academic communities
- A bit long in the tooth and hard to use

Cloud Computing

- “Cloud” is a metaphor for the network
- Physical Infrastructure abstracted away
- Resources accessed as “Services”
- Resources are shared transparently – “Virtualised”
- Inherently scalable
- Guarantees of service
- Mainly commercial (Amazon, Google …) and Pay As You Go
- Simple to use

Cloud Services

- “Software as a Service” (SaaS) e.g. Gmail, Google Apps (App = Application)
- “Platform as a Service” (PaaS) e.g. Google App Engine, MS SQL Services
- “Infrastructure as a Service” (IaaS) e.g. Amazon EC2

- SaaS – user just uses the service
- PaaS – service appears as potentially infinitely large computer running unknown OS
- IaaS – Service hardware abstracted away

AMAZON EC2

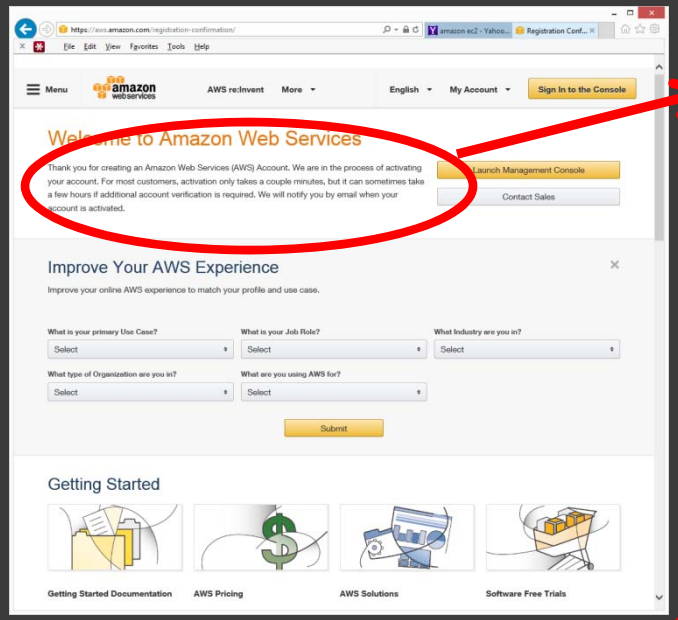
"Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers."

You can try Amazon EC2 for free (750 hours of "t2.micro" instances per month for a year).

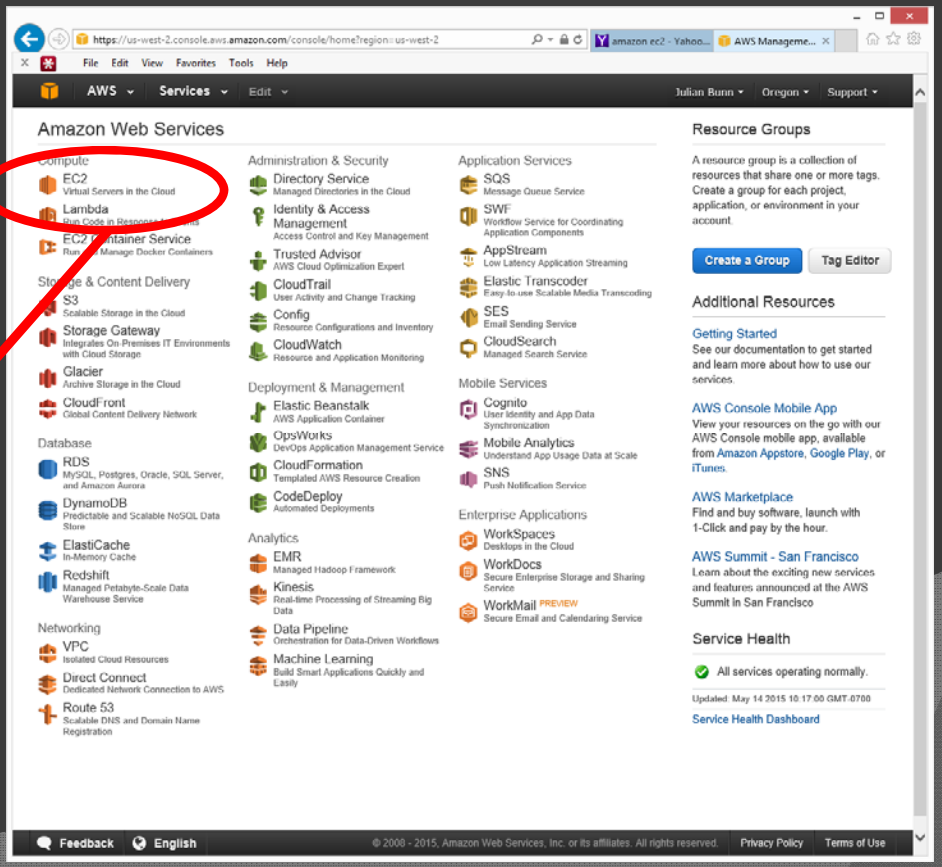
SETTING UP FOR EC2

- Create an account
  - Go to http://aws.amazon.com/ec2/
  - Click “Try AWS for Free”
  - Create a new user using your email address
  - You will need to enter a credit card number (I used a gift card with a few $ on it)
  - Amazon will call your ‘phone and ask you to enter the on-screen PIN
  - Sign up for the basic free support plan

Use the Management Console

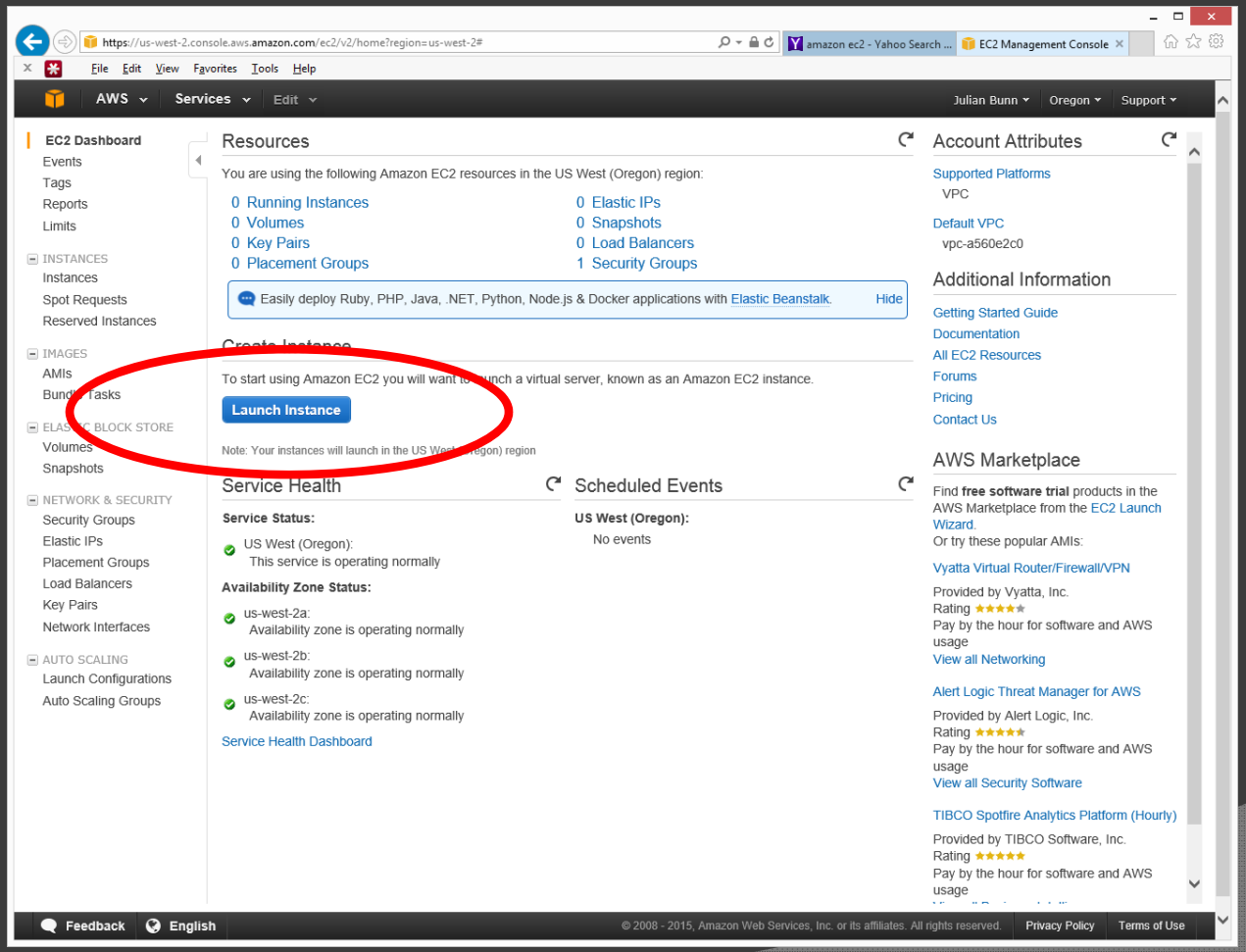


(There is sometimes a delay before you can use the Management Console.)

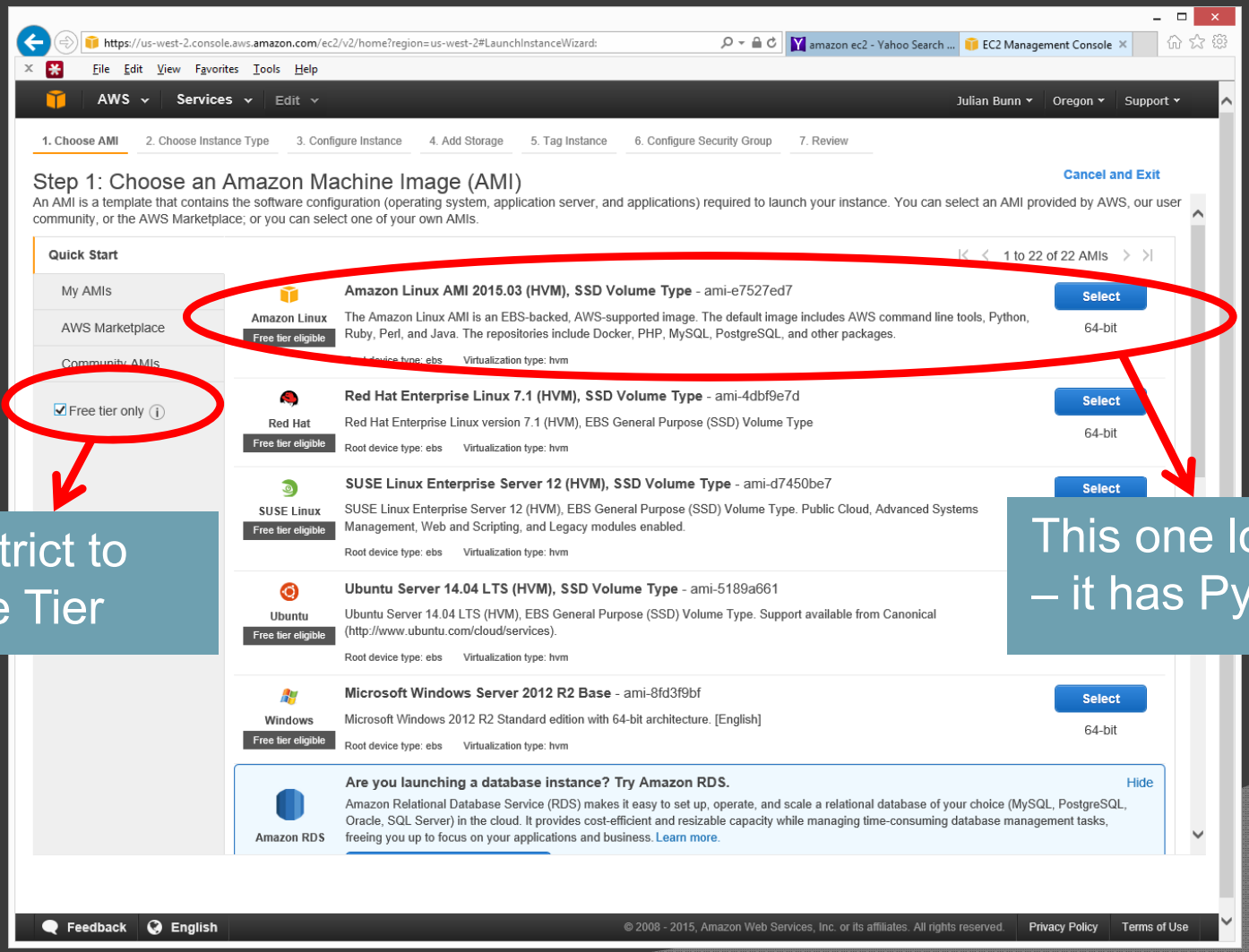


Select the “EC2” service

Launch an EC2 Instance



Choose the Machine Image type

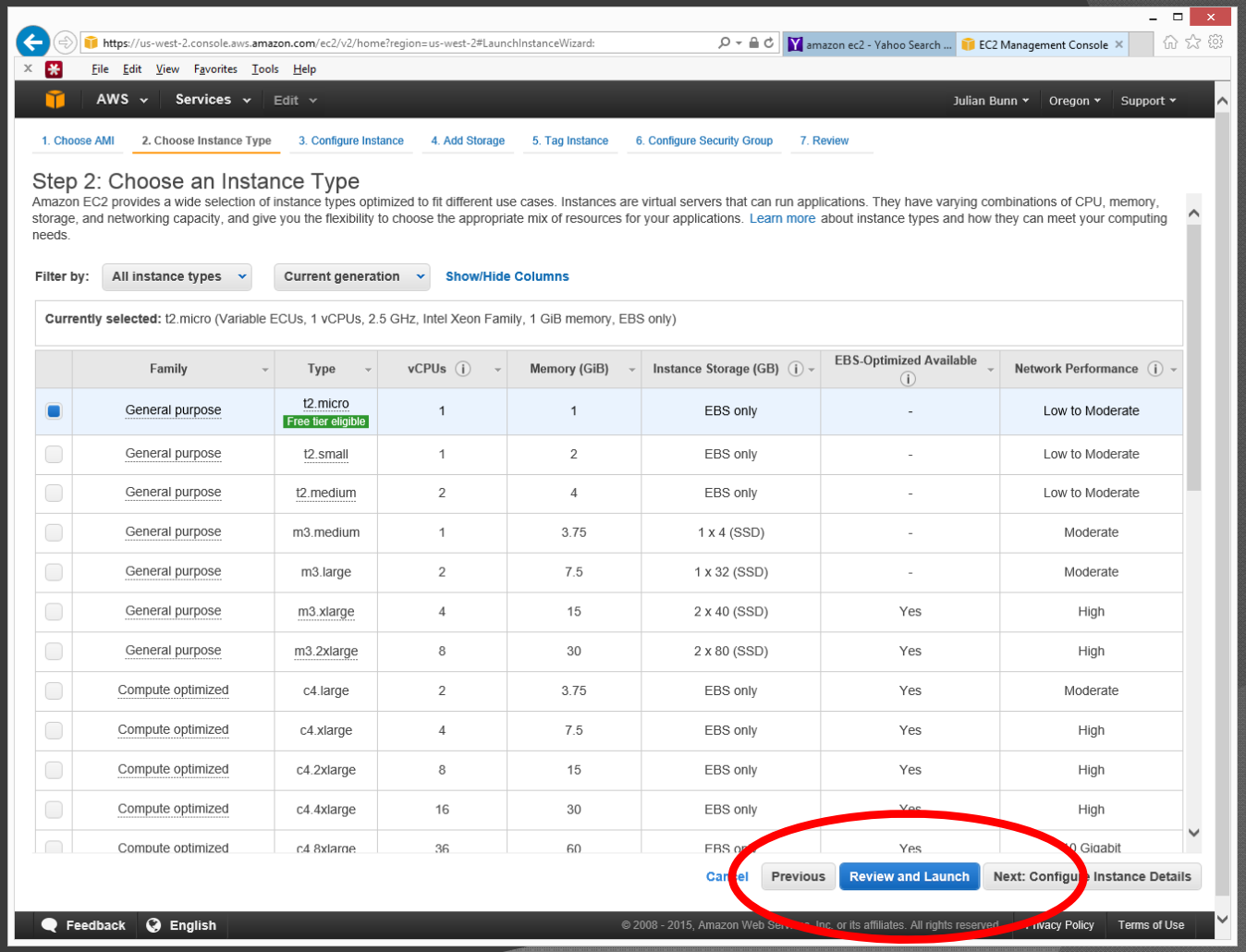


Restrict to Free Tier

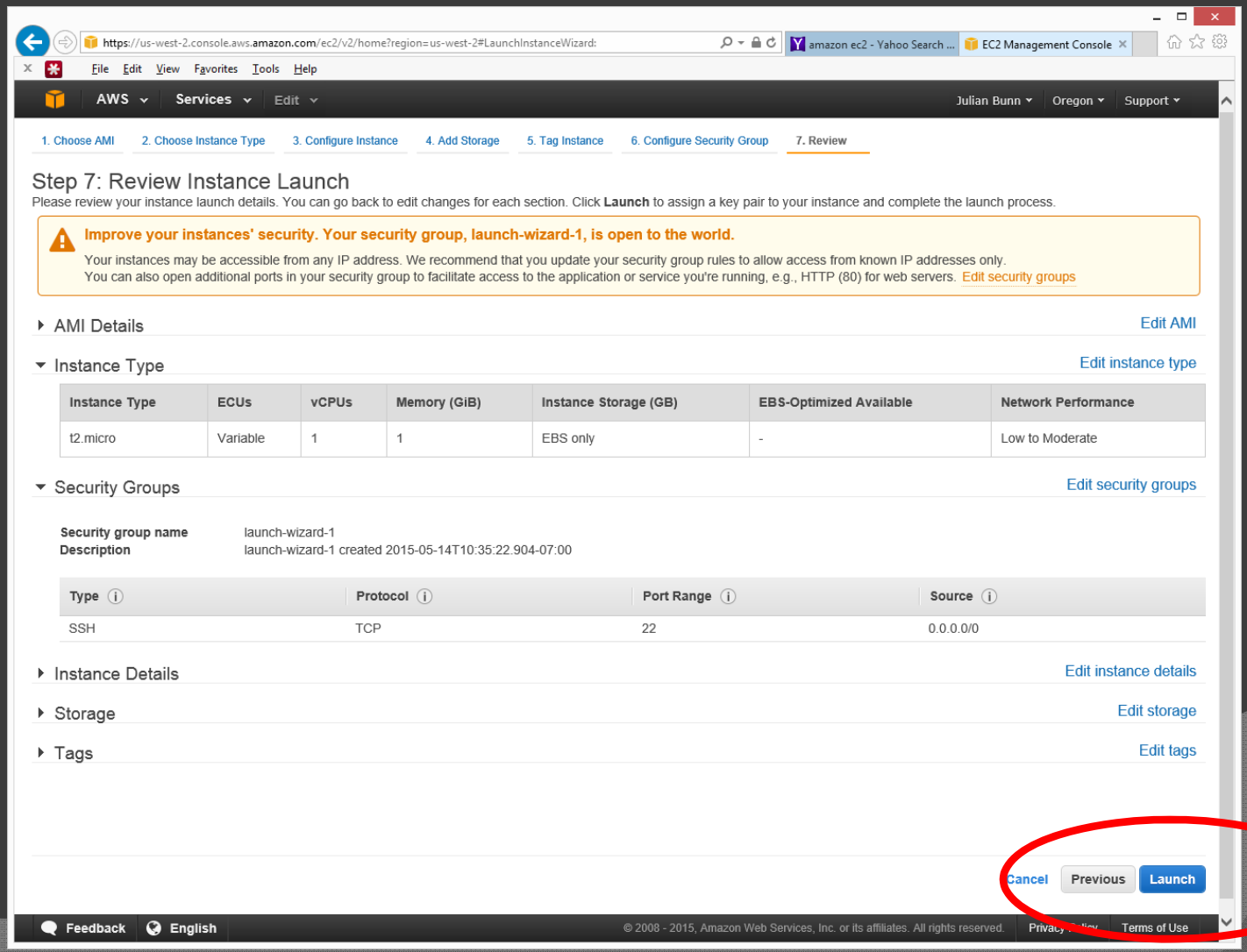
This one looks fine – it has Python

Choose the Instance type

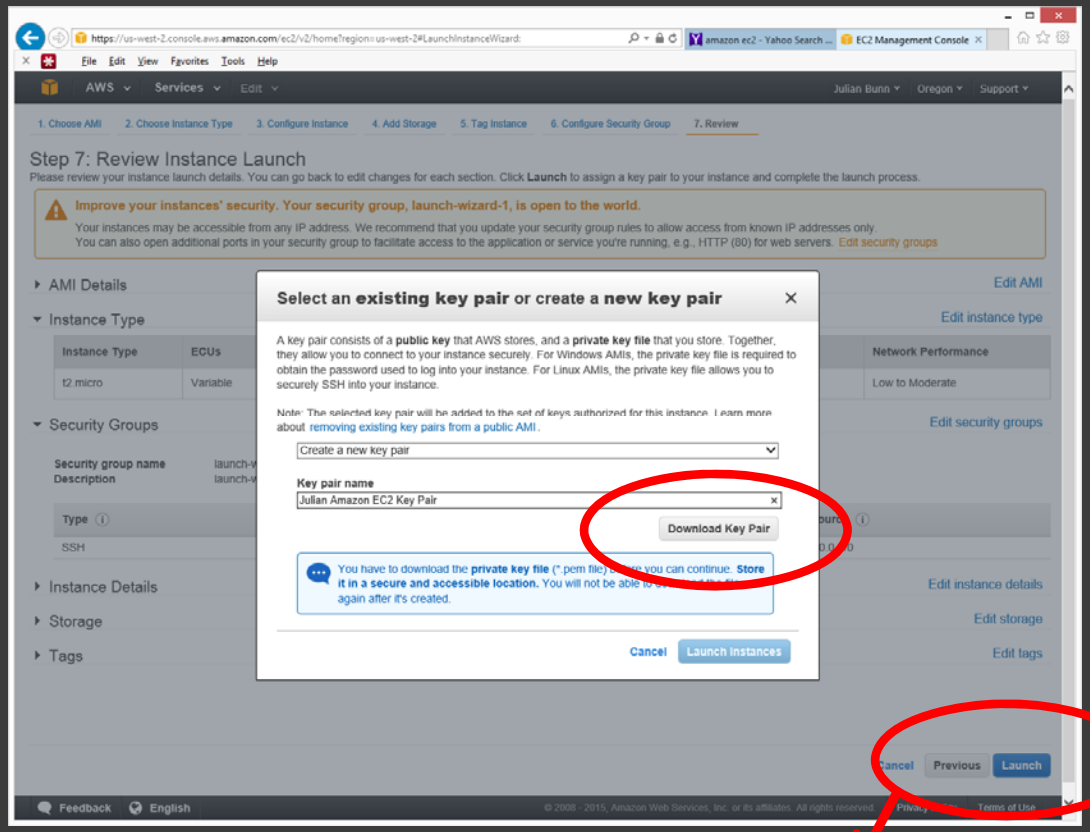
t2.micro



Launch the Instance



Create a Key Pair for access



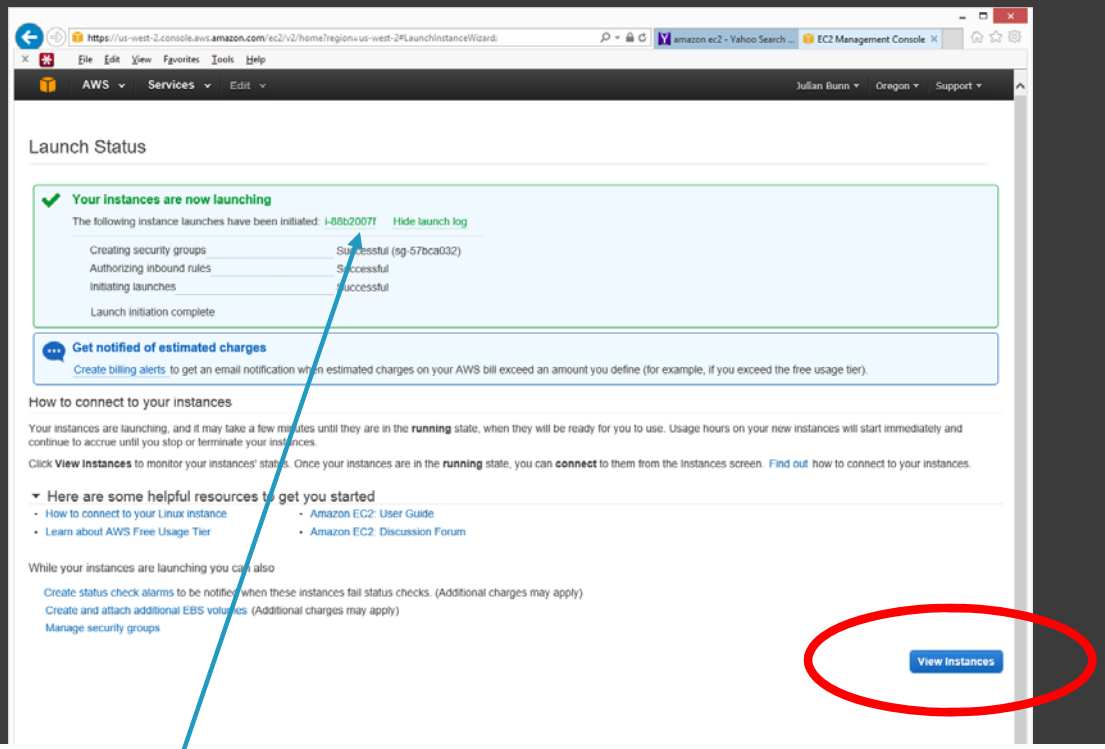
Once you have the Key Pair downloaded, launch your Instance

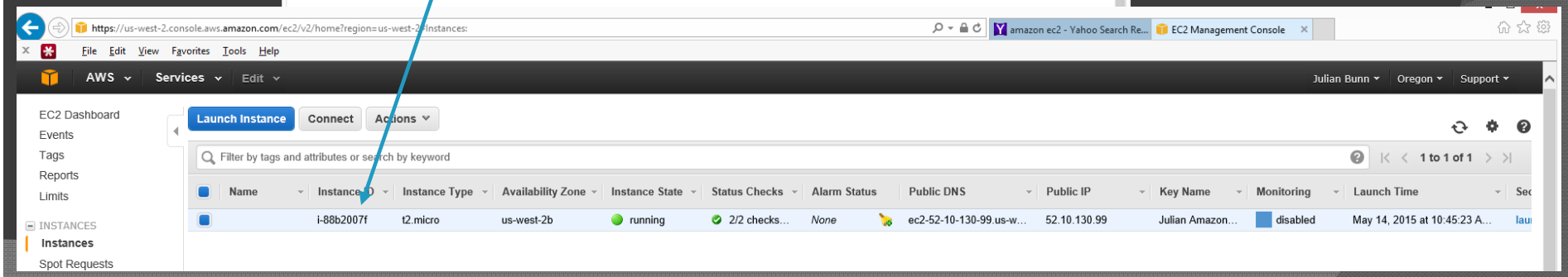
You save the Key/Pair file (.pem) on your local machine.

It's a text file:

-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEAq7tMKYNXAdjwFtuItbRszAzXAfY6i+fk41rwcN0ZUXFDfmoqkdysOZbJB0Rb
bm/qroEWS2LLGCmEFnxDNBpIqIMJZ9EUOj75XoMTYJoXykQZWZxNV6pdGxrULgdnUCbtCFdajgm/
gGuINcHj/Mdu450YdgPnWMKSC9+bBMFMIX11mPWVuGx+thH8jfV90c2510WniCzNGafwzhfSX2uF
oaUJUZd3/VEKjZ6AnuQj0TcSR6xGHdpxtbelfomdeqoKwnAYr4Nn2H94JZUgNj+z2dy9onZusgKW
esQMyMyW2ND+FtPI6bqO9VyAAHzuqqTORhDjeNbL/G93PFyHs6ZX4QIDAQABAoIBADm9cL8odNGf
MiJNT+rgLhBxZL05H7g4kOHMirk37bJPZcUURxnh21pFk3PJDQDVyk9AAdH49pY6Jzo0L5hOI84D
uN7aEnX/BHM9mP9K8CLGwMs7XVnTYxig7aRIQvlVDS9Xx4+oQWYZS2QGu/QazfLjt/y+FUPnr5kk
Zoezwl8kUYFogTacS3rsvWqzvLSvtfwgUJzeGnSKqAL+sVPSrZ2Xl/CKZhsHBvy/eGYdUybncIde
XMd1BLHymP/os9Xcq+RCYhatu3XFHZdvoIZz33AjGiVXFuo+0nOMhE4To6iQ8sLF1ZpSk+E2n5V4
qiiEqr8ZVbaSzAGCEi0gfJ4+Q00CgYEA3FJScgt3RLrK+BJqIk9VSy72fPbX/7CoaJoQG5HaY+kC
LHdfPsA0VXPr/zChsIEaYKsmw/pfsh35h6UmE8JNBOD31l0FYhjG2RceA/719sMYGDjZC1w0EcQC
2AEByDyZsaDutRsjnKmAiyHc9fK5q44HUmXYOzDsCA19zP9Ne9cCgYEAx4qf/syTbRYH5MK5r1Dt
TOLclQn2h/lIQHryM10p8r87ktVWD/qKRoYlF05iIRom2rq0vrRnPX8BONrYMjI0sET2TFN9ClgH
rUI+XnLWMD7JFLTrTmJeYtrl94MwdHnvq68VLsl26STUMrjdJ9ajYWIlHia0Okjhje7t29eBEwcC
gYAEoDXa7yKxatAyNukBArE85CVoYQQ0B8lud6x0E/PoXl30ddCpGLAsfU70IweAUPB
GaADmRQTIiDH2XFarddPF/YU2MdojhHb/HylBYyJElaHWPY2Us8WxS7068wXQVqeD3EDetKl12/O
ikzdDgZcN8q84lujojieQQKBgQClH8FA4FgLiJSlqMJylt0MqBsTuCPr5+TzFArDmI/zEm8hFiuu
81xj+fRNsg6gZ/zJOoIB4IYg3K3NN8R63XMefyL2jIxGeBMqPExSamN27y4FVaX3ao3C7nQpe8Wo
TZOZFjzlQe2gx1pWUwaoRgVTwhzsc4jnRpJomtT4/QASFwKBgQDR0CM30mW6BjAHXEZbydTVcCZM
jxYi2JfXkP0TniSK0Hqk3QD9iLdouYroG3QxprfV+p1bHaVz/N2xZZ/YAZ9sySv86WO0NHtSOXEx
3UZ/wVynpoN58YoPmr2/pAKQXqfydLA4sY2x5OlDlaA3aN73gHAcSr/O1ABms3oCW5LlfQ==
-----END RSA PRIVATE KEY-----

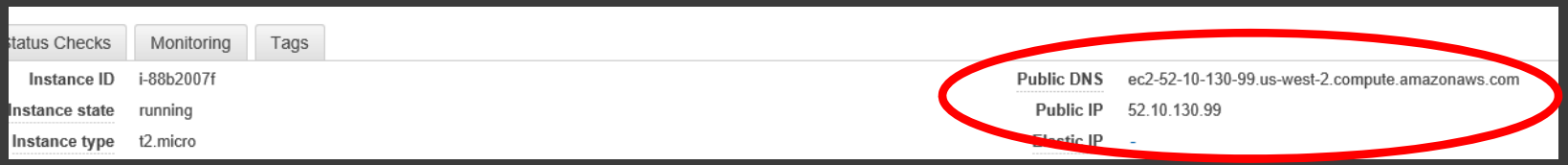
Check Launch and View Instances





ssh to your Instance

- Look for the public IP/DNS of your Instance



- From a command prompt on your local machine, ssh to your Instance:

```
C:\Users\Julian>ssh -i JulianAmazonEC2KeyPair.pem ec2-user@52.10.130.99

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2015.03-release-notes/
11 package(s) needed for security, out of 36 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-27-10 ~]$
```

Using your Instance - Example

```
[ec2-user@ip-172-31-27-10 ~]$ cat > test.py
print 'Hello World!'
[ec2-user@ip-172-31-27-10 ~]$ python test.py
Hello World!
```

Google App Engine

- Run your Web Applications on Google's Infrastructure
- Idea: upload your Web App, and it's ready to serve users
- App can be written in Java, JavaScript, Ruby, Python
- Persistent storage, automatic scaling and load balancing, authentication
- Task Queues (work outside Web requests)
- Resources used measured in GBytes and CPU Hours

App Engine Datastore

- Distributed storage with a Query Engine and Transactions
- Unlike traditional relational database
- Objects (“entities”) have kind and properties
- Queries return entities of a kind sorted by values of properties
- No schema

App Engine Services and Tasks

- URL Fetch
- Mail
- Memcache (in-memory key/value pairs)
- Image manipulation
- (Others you create in code)
- Tasks
  - Cron-like
  - Asynchronous to Web requests

Alternatives?

- “AppScale” – from UC Santa Barbara
- Open Source implementation of App Engine
- Executes in Eucalyptus (IaaS and compatible with e.g. Amazon EC2)
- Could be deployed in a private cloud