

EC2 Management Console

us-east-2.console.aws.amazon.com/ec2/v2/home

Services Resource Groups

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Launch Templates

Spot Requests

Reserved Instances

Dedicated Hosts

Capacity Reservations

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK STORE

Volumes

Snapshots

Lifecycle Manager

NETWORK & SECURITY

Security Groups

Elastic IPs

Launch Instance

Connect

Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Insta
	i-0055da59589301a8	t2.m
	i-01b5466e1abbc3bb6	t2.m
	i-031cf2768563ac544	t2.m

Instance: i-031cf2768563ac544 Public DNS: ec2-3-17-204-10.us-east-2.compute.amazonaws.com

Description

Status Checks

Monitoring

Tags

Instance ID: i-031cf2768563ac544

Instance state: running

Instance type: t2.micro

Elastic IPs

Availability zone: us-east-2c

Public DNS (IPv4): ec2-3-17-204-10.us-east-2.compute.amazonaws.com

IPv4 Public IP: 3.17.204.10

IPv6 IPs: -

Private DNS: p-172-31-45-118.us-east-2.compute.internal

Private IP: 172.31.45.118

Putty Configuration

Category: Session

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): 204-10.us-east-2.compute.amazonaws.com

Port: 22

Connection type: ☐ Raw ☐ Telnet ☐ Rlogin ☒ SSH ☐ Serial

Load, save or delete a stored session

Saved Sessions

Default Settings

aws\_1

pl02\_wif

rasp\_ethernet

rasp\_wif

Close window on exit

☐ Always ☐ Never ☒ Only on clean exit

Open Cancel

EC2 Management Console

us-east-2.console.aws.amazon.com/ec2/v2/home

Services Resource Groups

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Launch Templates

Spot Requests

Reserved Instances

Dedicated Hosts

Capacity Reservations

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK STORE

Volumes

Snapshots

Lifecycle Manager

NETWORK & SECURITY

Security Groups

Elastic IPs

Launch Instance

Connect

Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Insta
	i-0055da59589301a8	t2.m
	i-01b5466e1abbc3bb6	t2.m
	i-031cf2768563ac544	t2.m

Instance: i-031cf2768563ac544 Public DNS: ec2-3-17-204-10.us-east-2.compute.amazonaws.com

Description

Status Checks

Monitoring

Tags

Instance ID: i-031cf2768563ac544

Instance state: running

Instance type: t2.micro

Elastic IPs

Availability zone: us-east-2c

Public DNS (IPv4): ec2-3-17-204-10.us-east-2.compute.amazonaws.com

IPv4 Public IP: 3.17.204.10

IPv6 IPs: -

Private DNS: p-172-31-45-118.us-east-2.compute.internal

Private IP: 172.31.45.118

Putty Configuration

Category: SSH

Options controlling SSH authentication

☒ Display pre-authentication banner (SSH2 only)

☐ Bypass authentication entirely (SSH2 only)

Authentication methods

☒ Attempt authentication using Pageant

☐ Attempt TIS or CryptoCard auth (SSH-1)

☒ Attempt "keyboard-interactive" auth (SSH2)

Authentication parameters

☐ Allow agent forwarding

☐ Allow attempted changes of username in SSH2

Private key file for authentication:

C:\Users\nick2\Downloads\Py Projects\aws

Browse...

Open Cancel

EC2 Management Console

us-east-2.console.aws.amazon.com/ec2/v2/home

Services Resource Groups

EC2 Dashboard

Launch Instance

Filter by tags and attributes or search by keyword

Name	Instance ID	Insta
	i-0055da59589391a8	t2.m
	i-01b5466e1abbc3bb6	t2.m
	i-031cf2768563ac544	t2.m

Putty Configuration

Category: SSH

Options controlling SSH port forwarding

Port forwarding

☐ Local ports accept connections from other hosts

☐ Remote ports do the same (SSH-2 only)

Forwarded ports

D0081

Add new forwarded port

Source port: 3081

Destination: ☐ Local ☐ Remote ☒ Dynamic

☐ IPv4 ☐ IPv6

Open Cancel

Instance: i-031cf2768563ac544 Public DNS: ec2-3-17-204-10.us-east-2.compute.amazonaws.com

Description Status Checks Monitoring Tags

Instance ID	Instance state	Instance type	Elastic IPs
i-031cf2768563ac544	running	t2.micro	

Public DNS (IPv4) ec2-3-17-204-10.us-east-2.compute.amazonaws.com

IPv4 Public IP 3.17.204.10

IPv6 IPs -

Private DNS p-172-31-45-118.us-east-2.compute.internal

Private IP 172.31.45.118

EC2 Management Console

us-east-2.console.aws.amazon.com/ec2/v2/home

Services Resource Groups

EC2 Dashboard

Launch Instance

Filter by tags and attributes or search by keyword

Name	Instance ID	Insta
	i-0055da59589391a8	t2.m
	i-01b5466e1abbc3bb6	t2.m
	i-031cf2768563ac544	t2.m

Putty Configuration

Category: Session

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address) ubuntu@ec2-3-17-204-10.us-east-2.compute

Port 22

Connection type ☐ Raw ☐ Telnet ☐ Rlogin ☒ SSH ☐ Serial

Load, save or delete a stored session

Saved Sessions

Default Settings

aws\_1

aws\_2

aws\_3

pi02\_wifi

rasp\_ethernet

rasp\_wifi

Load Save Delete

Close window on exit ☐ Always ☐ Never ☒ Only on clean exit

Open Cancel

Instance: i-031cf2768563ac544 Public DNS: ec2-3-17-204-10.us-east-2.compute.amazonaws.com

Description Status Checks Monitoring Tags

Instance ID	Instance state	Instance type	Elastic IPs
i-031cf2768563ac544	running	t2.micro	

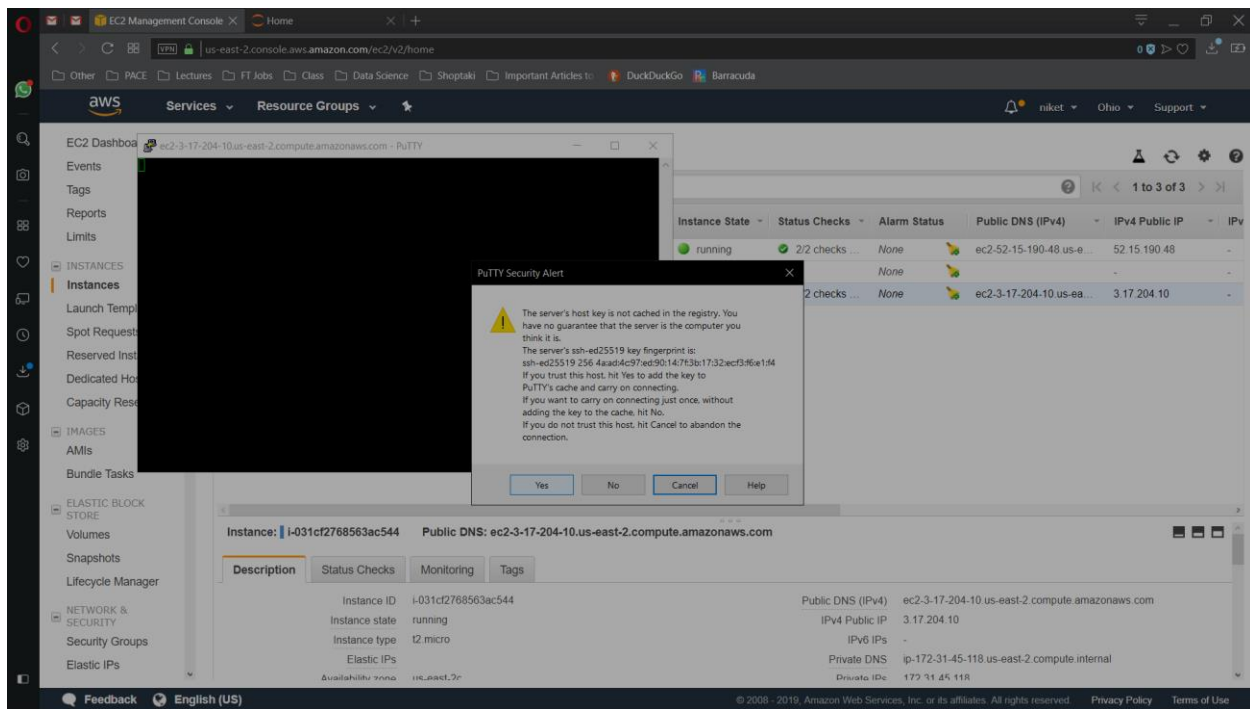
Public DNS (IPv4) ec2-3-17-204-10.us-east-2.compute.amazonaws.com

IPv4 Public IP 3.17.204.10

IPv6 IPs -

Private DNS p-172-31-45-118.us-east-2.compute.internal

Private IP 172.31.45.118



## Download Anaconda 3 installer

wget [https://repo.continuum.io/archive/Anaconda3-4.4.0-Linux-x86\\_64.sh](https://repo.continuum.io/archive/Anaconda3-4.4.0-Linux-x86_64.sh)

```
ubuntu@ip-172-31-45-118:~$ wget https://repo.continuum.io/archive/Anaconda3-4.4.0-Linux-x86_64.sh
--2019-02-22 00:08:01-- https://repo.continuum.io/archive/Anaconda3-4.4.0-Linux-x86_64.sh
Resolving repo.continuum.io (repo.continuum.io)... 104.16.19.10, 104.16.18.10, 2606:4700::6810:120a, ...
Connecting to repo.continuum.io (repo.continuum.io)|104.16.19.10|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 523283080 (499M) [application/x-sh]
Saving to: 'Anaconda3-4.4.0-Linux-x86_64.sh'

Anaconda3-4.4.0-Linux-x 100%[=====>] 499.04M 61.4MB/s in 7.6s

2019-02-22 00:08:08 (65.5 MB/s) - 'Anaconda3-4.4.0-Linux-x86_64.sh' saved [523283080/523283080]
```

## Install Anaconda3 by typing:

```
bash Anaconda3-4.4.0-Linux-x86_64.sh
```

```
ubuntu@ip-172-31-45-118:~$ bash Anaconda3-4.4.0-Linux-x86_64.sh

Welcome to Anaconda3 4.4.0 (by Continuum Analytics, Inc.)

In order to continue the installation process, please review the license
agreement.
Please, press ENTER to continue
>>> █
```

```
installing: anaconda-4.4.0-np112py36_0 ...
installing: conda-4.3.21-py36_0 ...
installing: conda-env-2.6.0-0 ...
Python 3.6.1 :: Continuum Analytics, Inc.
creating default environment...
installation finished.
Do you wish the installer to prepend the Anaconda3 install location
to PATH in your /home/ubuntu/.bashrc ? [yes|no]
[no] >>> yes █
```

To switch your environment to use Python 3 type the command:

```
which python /usr/bin/python
```

```
source .bashrc
```

```
ubuntu@ip-172-31-45-118:~$ which python /usr/bin/python3
/usr/bin/python3
ubuntu@ip-172-31-45-118:~$ source .bashrc
ubuntu@ip-172-31-45-118:~$ █
```

## Create your Jupyter/Ipython password:

```
ipython
```

```
from IPython.lib import passwd
```

```
passwd()
```

**save your SHA hash for future reference by copying and pasting it to a text file, you'll need this later**

```
ubuntu@ip-172-31-45-118:~$ ipython
Python 3.6.1 |Anaconda 4.4.0 (64-bit)| (default, May 11 2017, 13:09:58)
Type "copyright", "credits" or "license" for more information.

IPython 5.3.0 -- An enhanced Interactive Python.
?                -> Introduction and overview of IPython's features.
%quickref        -> Quick reference.
help             -> Python's own help system.
object?         -> Details about 'object', use 'object??' for extra details.

In [1]: from IPython.lib import passwd

In [2]: passwd()
Enter password:
Verify password:
Out[2]: 'sha1:1934e8c03760:6487537af42bb75cd7c3a094d167135d83cac375'

In [3]: exit
```

```
sha1:1934e8c03760:6487537af42bb75cd7c3a094d167135d83cac375
```

**Configure Jupyter/Ipypython server to access your notebooks from local computer via internet browser:**

```
jupyter notebook --generate-config
```

```
ubuntu@ip-172-31-45-118:~$ jupyter notebook --generate-config
Writing default config to: /home/ubuntu/.jupyter/jupyter_notebook_config.py
ubuntu@ip-172-31-45-118:~$
```

Generate key & cert into the jupyter configuration folder:

```
openssl req -x509 -nodes -days 365 -newkey rsa:1024 -keyout ~/.jupyter/key.key -out  
~/.jupyter/cert.pem
```

```
ubuntu@ip-172-31-45-118:~$ jupyter notebook --generate-config
Writing default config to: /home/ubuntu/.jupyter/jupyter_notebook_config.py
ubuntu@ip-172-31-45-118:~$ cd .
./      ../      .cache/  .gnupg/  .ipython/ .jupyter/ .ssh/
ubuntu@ip-172-31-45-118:~$ cd .jupyter/
ubuntu@ip-172-31-45-118:~/.jupyter$ openssl req -x509 -nodes -days 365 -newkey rsa:1024 -keyo
ut ~/.jupyter/key.key -out ~/.jupyter/cert.pem
Generating a 1024 bit RSA private key
.....+++++
.....+++++
writing new private key to '/home/ubuntu/.jupyter/key.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:
ubuntu@ip-172-31-45-118:~/.jupyter$
```

## Edit your Jupyter configuration file:

```
nano .jupyter/jupyter_notebook_config.py
```

```
ubuntu@ip-172-31-45-118:~/.jupyter$ nano jupyter_notebook_config.py
c = get_config()

# Kernel config
c.IPKernelApp.pylab = 'inline' # if you want plotting support
always in your notebook

# Notebook config
import os
```

```

c.NotebookApp.keyfile = os.path.expanduser('~') + '/.jupyter/key.key'
c.NotebookApp.certfile = os.path.expanduser('~') + '/.jupyter/cert.pem'
#location of your certificate file
c.NotebookApp.ip = '0.0.0.0'
c.NotebookApp.open_browser = False #so that the ipython
notebook does not opens up a browser by default
c.NotebookApp.password = u'
sha1:1934e8c03760:6487537af42bb75cd7c3a094d167135d83cac375'
#edit this with the SHA hash that you generated after typing in
Step 9
# This is the port we opened in Step 3.
c.NotebookApp.port = 8081

```

```

GNU nano 2.9.3 jupyter_notebook_config.py

c = get_config()

# Kernel config
c.IPKernelApp.pylab = 'inline' # if you want plotting support always in your notebook

# Notebook config
import os
c.NotebookApp.keyfile = os.path.expanduser('~') + '/.jupyter/key.key'
c.NotebookApp.certfile = os.path.expanduser('~') + '/.jupyter/cert.pem'
#location of your certificate file
c.NotebookApp.ip = '0.0.0.0'
c.NotebookApp.open_browser = False #so that the ipython notebook does not opens up a browse$
c.NotebookApp.password = u' sha1:1934e8c03760:6487537af42bb75cd7c3a094d167135d83cac375' #ed$
# This is the port we opened in Step 3.
c.NotebookApp.port = 8081

# Configuration file for jupyter notebook

```

## Create a folder for your notebooks and start Jupyter Notebook:

```

mkdir Notebooks
cd Notebooks

```

```

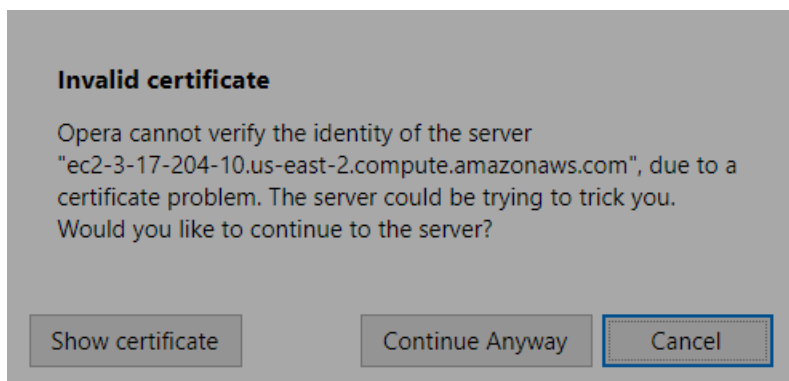
jupyter notebook

```

```
ubuntu@ip-172-31-45-118:~$ mkdir Notebooks
ubuntu@ip-172-31-45-118:~$ cd Notebooks/
ubuntu@ip-172-31-45-118:~/Notebooks$ jupyter notebook
[I 00:32:53.727 NotebookApp] Writing notebook server cookie secret to /run/user/1000/jupyter/
notebook_cookie_secret
[I 00:32:53.770 NotebookApp] Serving notebooks from local directory: /home/ubuntu/Notebooks
[I 00:32:53.771 NotebookApp] 0 active kernels
[I 00:32:53.771 NotebookApp] The Jupyter Notebook is running at: https://0.0.0.0:8081/
[I 00:32:53.771 NotebookApp] Use Control-C to stop this server and shut down all kernels (twi
ce to skip confirmation).
```

In browser:

<https://ec2-3-17-204-10.us-east-2.compute.amazonaws.com:8081/>



Click Continue Anyway

