```
Step 1. Python code will be attached
From python console
From EC2 Dashboard
From Volumes
SSH connection instruction
Step 2. SSH access
Step 3. Take a snapshot and terminate the first EC2
New snapshot
Terminate old EC2
Step 4. Create a new EC2 instance,
Create a new EC2 instance
Create an EBS volume from the snapshot
Final verification of new EBS mount
```

Step 1. Python code will be attached

From python console

```
ec2-user@ip-172-31-3-143:/proc

>>> l=launch_instance(key_name="keyPair0")

ERROR!

Creating keypair keyPair0

Group name= SecurityGroup:python2

Security group python2 already authorized
waiting for instance...

Instance is now running

Instance IP is 54.67.106.44
```

```
siming.meng@USABOS147305L ~/.aws
$ fg
python
inst=|[0]
res=|[1]inst=|[0]
>>> res=|[1]
>>> ec2=inst.connection
>>> azone=inst.placement
>>> azone
u'us-west-1a'
>>> ec2=inst.connection
>>> azone=inst.placement
>>> azone
u'us-west-1a'
>>> vol=ec2.create_volume(2,azone)
>>> vol
Volume:vol-06acff83bcd7113b6
>>> vol.attach(inst.id,'/dev/sdf')
```

From EC2 Dashboard



From Volumes



SSH connection instruction

Connect To Your Instance

- - A Java SSH Client directly from my browser (Java required)

To access your instance:

- 1. Open an SSH client. (find out how to connect using PuTTY)
- 2. Locate your private key file (keyPair0.pem). The wizard automatically detects the key you used to launch the instance.
- 3. Your key must not be publicly viewable for SSH to work. Use this command if needed:

```
chmod 400 keyPair0.pem
```

4. Connect to your instance using its Public DNS:

```
ec2-54-67-106-44.us-west-1.compute.amazonaws.com
```

Example:

ssh -i "keyPair0.pem" ec2-user@ec2-54-67-106-44.us-west-1.compute.amazonaws.com

Step 2. SSH access

```
[ec2-user@ip-172-31-3-143 proc]$ sudo !m
sudo mke2fs -F -j /dev/sdf
mke2fs 1.42.12 (29-Aug-2014)
Creating filesystem with 524288 4k blocks and 131072 inodes
Filesystem UUID: cd1e942d-654e-4525-908d-7dbfad5acca5
Superblock backups stored on blocks:
         32768, 98304, 163840, 229376, 294912
Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
[ec2-user@ip-172-31-3-143 proc]$ sudo mkdir /mnt/data-store
[ec2-user@ip-172-31-3-143 proc]$ sudo mount /dev/sdf /mnt/data-store
[ec2-user@ip-172-31-3-143 proc]$ df -T
                          1K-blocks
                                         Used Available Use% Mounted on
Filesystem
                Type
                              8123812 1222456
                                                            16% /
/dev/xvda1
                 ext4
                                                  6801108
                 devtmpfs
                                                              1% /dev
                              501092
                                             60
                                                    501032
devtmpfs
tmpfs
                 tmpfs
                              509668
                                             0
                                                    509668
                                                              0% /dev/shm
/dev/xvdf
                             1998672
                                           3140
                                                   1890676
                                                              1% /mnt/data-store
                ext3
[ec2-user@ip-172-31-3-143 proc]$ cd /mnt/data-store/
[ec2-user@ip-172-31-3-143 data-store]$ chmod 777 .
chmod: changing permissions of '.': Operation not permitted
[ec2-user@ip-172-31-3-143 data-store]$ sudo !ch
sudo chmod 777 .
[ec2-user@ip-172-31-3-143 data-store]$ ls -xap
   ../ lost+found/
how are you?
[ec2-user@ip-172-31-3-143 data-store]$ ls
cats.txt lost+found
 ec2-user@ip-172-31-3-143 data-store]$ more cats.txt
now are you?
```

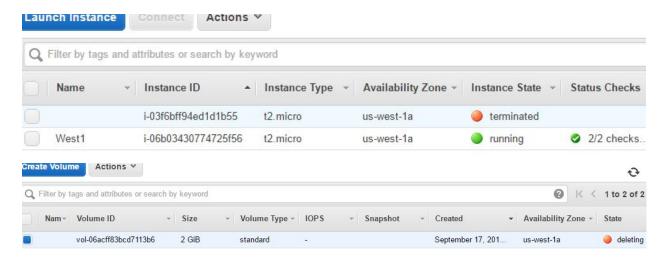
Step 3. Take a snapshot and terminate the first EC2

```
>>> vol.create_snapshot('ucsc-aws-class')
Snapshot:snap-0084de3c5a56e404f
>>>
>>> ec2=boto.connect_ec2()
>>> ec2=boto.ec2.connect_to_region('us-west-1')
>>> ec2
EC2Connection:ec2.us-west-1.amazonaws.com
>>> instlist=ec2.get_all_instances()
>>> instlist
[Reservation:r-0e98e6b8efe5eeab7, Reservation:r-0df772c324dae42f8, Reservation:r-0e4f84733fde9fd5b]
>>> res
Reservation:r-0e4f84733fde9fd5b
>>> inst=res.instances[0]
>>> inst.id
u'i-03f6bff94ed1d1b55'
>>> inst.stop()
>>> inst.stop()
>>> inst.terminate()
```

New snapshot

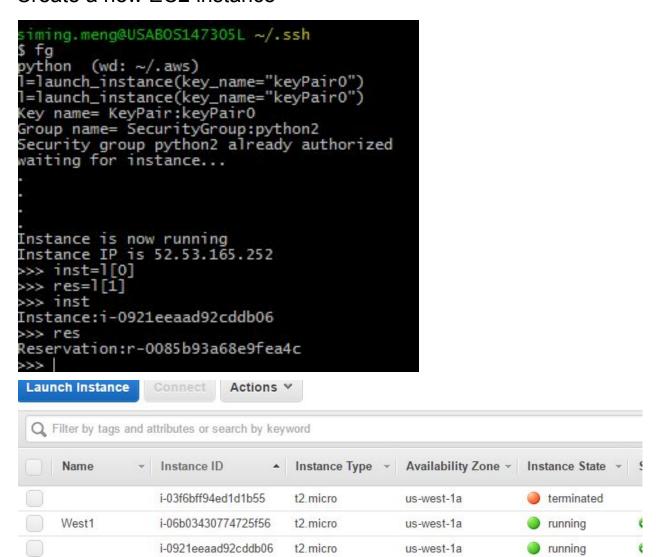


Terminate old EC2



Step 4. Create a new EC2 instance,

Create a new EC2 instance



Create an EBS volume from the snapshot

```
>> pprint (snapshots)
[Snapshot:snap-0084de3c5a56e404f]
>>> pprint (snapshots[0])
Snapshot:snap-0084de3c5a56e404f
>>> pprint (snapshots[0].name)
Traceback (most recent call last):
File "<stdin>", line 1, in <module>
AttributeError: 'Snapshot' object has no attribute 'name'
>>> dir (snapshots[0])
SyntaxError: invalid syntax

>>> snapshots[0]['volume_id']
Traceback (most recent call last):
   File "<stdin>", line 1, in <module>
TypeError: 'Snapshot' object has no attribute '__getitem__'
>>> snapshots[0].volume_id
u'vol-06acff83bcd7113b6'
>>> snapshots[0].id
u'snap-0084de3c5a56e404f'
>>> snapShotId = snapshots[0].id
>>> snapShotId
u'snap-0084de3c5a56e404f'
>>> vol=ec2.create_volume(2,azone, snapShotId)
>>> vol
Volume:vol-0d7b3001164277cd6
>>> vol.attach(inst.id,'/dev/sdf')
u'attaching'
```

Final verification of new EBS mount

```
>>> ec2=boto.connect_ec2()
>>> ec2=boto.ec2.connect_to_region('us-west-1')
>>> azone=inst.placement
 >> azone
u'us-west-1a'
>>> filters = {
... 'description': snapshotName
 ...,
>>> snapshots = ec2.get_all_snapshots(filters=filters)
>>> snapShotId = snapshots[0].id
 >> snapShotId
u'snap-0084de3c5a56e404f'
>>> vol=ec2.create_volume(2,inst.placement, snapShotId)
>>> vol.attach(inst.id,'/dev/sdf')
u'attaching
 >> vol
/olume:vol-00976e2fb3f35d36a
[1]+ Stopped
                                            python (wd: ~/.aws)
(wd now: ~/.ssh)
 iming.meng@USABOS147305L ~/.ssh
$ ssh -i "keyPairO.pem" ec2-user@ec2-52-53-242-41.us-west-1.compute.amazonaws.com
The authenticity of host 'ec2-52-53-242-41.us-west-1.compute.amazonaws.com (52.53.242.41)' car
ECDSA key fingerprint is SHA256:8GVaupJEhN]FxJ8kkSCsaZLhNg5NeuvvmSEpv1cNAkw.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-52-53-242-41.us-west-1.compute.amazonaws.com,52.53.242.41' (E0
                               Amazon Linux AMI
nttps://aws.amazon.com/amazon-linux-ami/2016.03-release-notes/
13 package(s) needed for security, out of 26 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-13-78 ~]$ cat /proc/partitions
major minor #blocks name
 202
                 0
                         8388608 xvda
Used Available Use% Mounted on
                                                                             16% /
1% /dev
0% /dev/shm
1% /mnt/dz
                      devtmpfs
devtmpfs
                                       501092
                                                         60
                                                                  501032
tmpfs
/dev/xvdf
                      tmpfs
                                       509668
                                                          0
                                                                  509668
                      ext3
                                      1998672
                                                      3144
                                                                 1890672
[ec2-user@ip-172-31-13-78 ~]$ ls /mnt/dz/
ats.txt lost+found
 ec2-user@ip-172-31-13-78 ~]$ cat /mnt/dz/cats.txt
 low are you?
 ec2-user@ip-172-31-13-78 ~]$ |
```