

Handed out: 09/26/2015

Due by 11:59 PM EST on Friday, 10/2/2015

1. Use AWS CLI commands unless otherwise noted.

- a. Select an EBS backed AMI (Linux) from Public using Amazon Console.

Name	AMI Name	AMI ID	Source	Owner	Visibility	Status	Creation Date	Platform
	getting-started...	ami-b232d0db	amazon/getting-...	amazon	Public	available	December 2, 2009 at 11:44:...	Other Linux

- b. Describe your EBS backed Linux AMI.

```
marnie@Marnies-MacBook-Air:~ marnie$ aws ec2 describe-images --image-ids ami-b232d0db
{
    "Images": [
        {
            "Architecture": "i386",
            "CreationDate": "2009-12-02T16:44:20.000Z",
            "Description": "Fedora 8 v1.14 i386 std-root lvm-swap lvm-storage getting-started-guide",
            "Hypervisor": "xen",
            "ImageId": "ami-b232d0db",
            "ImageLocation": "amazon/getting-started-with-ebs-boot",
            "ImageOwnerAlias": "amazon",
            "ImageType": "machine",
            "KernelId": "aki-94c527fd",
            "Name": "getting-started-with-ebs-boot",
            "OwnerId": "206029621532",
            "Public": true,
            "RamdiskId": "ari-96c527ff",
            "RootDeviceName": "/dev/sda1"
        }
    ]
}
```

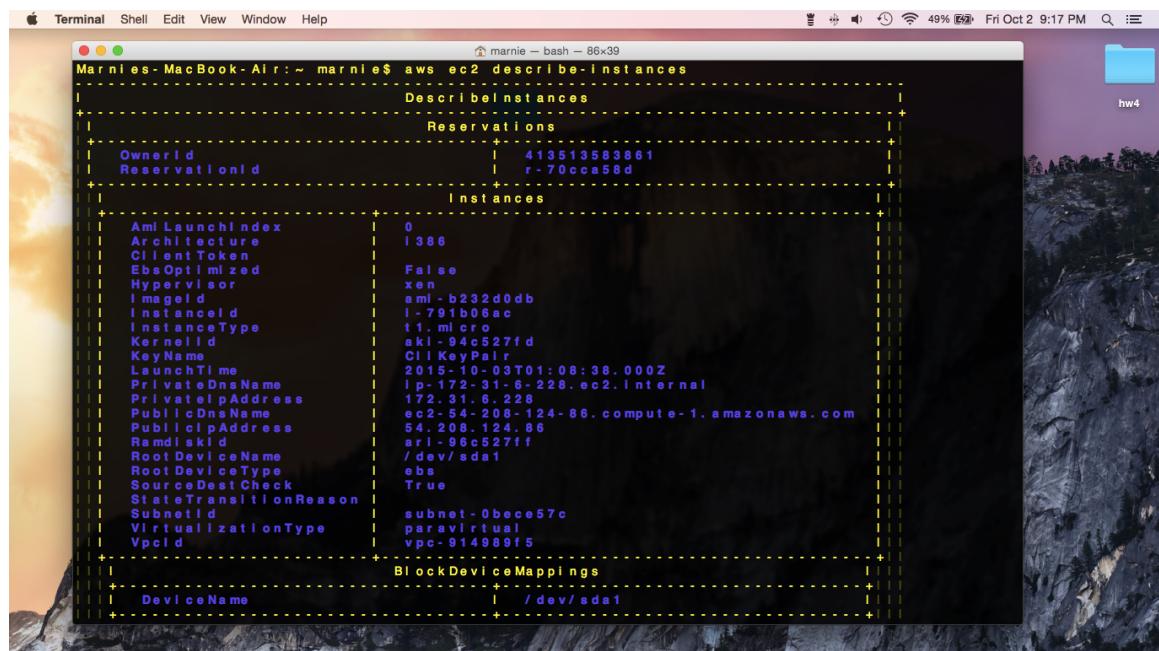
Marnie Scully CSCIE 90 HW#4

- c. Create two instances from your AMI.



```
marnie$ aws ec2 run-instances --image-id ami-b232d0db --count 2 --instance-type t1.micro --key-name CliKeyPair --security-groups RDPGroup
marnie$ aws ec2 run-instances --image-id ami-b232d0db --count 2
--instance-type t1.micro --key-name CliKeyPair --security-groups RDPGroup
+-----+
| RunInstances
+-----+
| OwnerId          | 413513583861
| ReservationId   | r-70cca58d
+-----+
| Instances
+-----+
| AmiLaunchIndex   | 0
| Architecture     | i386
| ClientToken      | False
| EbsOptimized     | xen
| Hypervisor       | ami-b232d0db
| ImageId          | i-791b06ac
| InstanceId       | t1.micro
| KernelId         | aki-94c527fd
| KeyName          | CliKeyPair
| LaunchTime       | 2015-10-03T01:08:38.000Z
| PrivateDnsName   | ip-172-31-6-228.ec2.internal
| PrivateIpAddress | 172.31.6.228
| PublicDnsName    | 
| PublicIpAddress  | 
| RamdiskId        | ari-96c527ff
| RootDeviceName   | /dev/sda1
| RootDeviceType   | ebs
| SourceDestCheck  | True
| StateTransitionReason | 
| SubnetId          | subnet-0bece57c
| VirtualizationType | paravirtual
| VpcId             | vpc-914989f5
+-----+
| Monitoring
+-----+
| State
+-----+
| NetworkInterfaces
+-----+
```

- d. Describe your instances and jot down the public DNS, the Volume ID and Root Device Name for both. Record the Device Name used by that volume (drive). Device names have patterns like /dev/sda1 or /dev/sdf.



```
marnie$ aws ec2 describe-instances
marnie$ aws ec2 describe-instances
+-----+
| DescribeInstances
+-----+
| Reservations
+-----+
| OwnerId          | 413513583861
| ReservationId   | r-70cca58d
+-----+
| Instances
+-----+
| AmiLaunchIndex   | 0
| Architecture     | i386
| ClientToken      | False
| EbsOptimized     | xen
| Hypervisor       | ami-b232d0db
| ImageId          | i-791b06ac
| InstanceId       | t1.micro
| KernelId         | aki-94c527fd
| KeyName          | CliKeyPair
| LaunchTime       | 2015-10-03T01:08:38.000Z
| PrivateDnsName   | ip-172-31-6-228.ec2.internal
| PrivateIpAddress | 172.31.6.228
| PublicDnsName    | ec2-54-208-124-86.compute-1.amazonaws.com
| PublicIpAddress  | 54.208.124.86
| RamdiskId        | ari-96c527ff
| RootDeviceName   | /dev/sda1
| RootDeviceType   | ebs
| SourceDestCheck  | True
| StateTransitionReason | 
| SubnetId          | subnet-0bece57c
| VirtualizationType | paravirtual
| VpcId             | vpc-914989f5
+-----+
| BlockDeviceMappings
+-----+
| DeviceName       | /dev/sda1
+-----+
```

My Instance Information:

Instance 1

AMI ID: ami-b232d0db

Instance ID: i-791b06ac

Volume ID: vol-0247f3e2

Device Name /dev/sda1:

Public DNS: ec2-54-208-124-86.compute-1.amazonaws.com

Public IP: 54.208.124.86

Instance 2

AMI ID: ami-b232d0db

Instance ID: i-781b06ad

Volume ID: vol-fa41f51a

Device Name: dev/sda1

Public DNS: ec2-54-208-221-69.compute-1.amazonaws.com

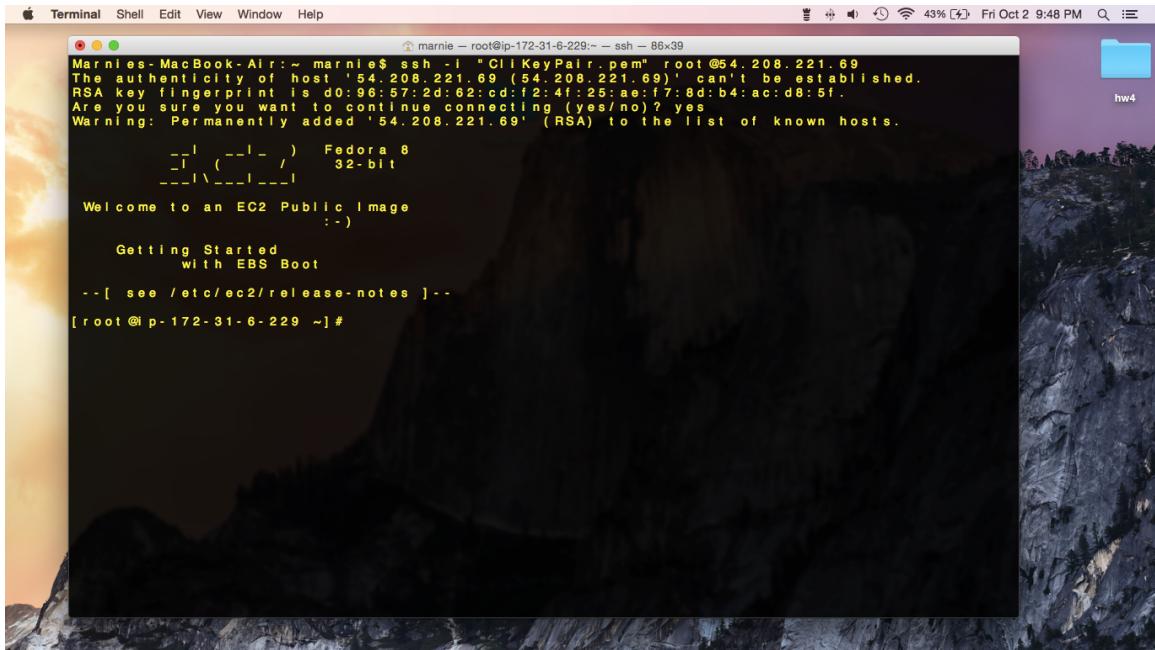
Public IP: 54.208.221.69

- e. Demonstrate that you can log into both instances.

Instance 1

Marnie Scully CSCIE 90 HW#4

Instance 2



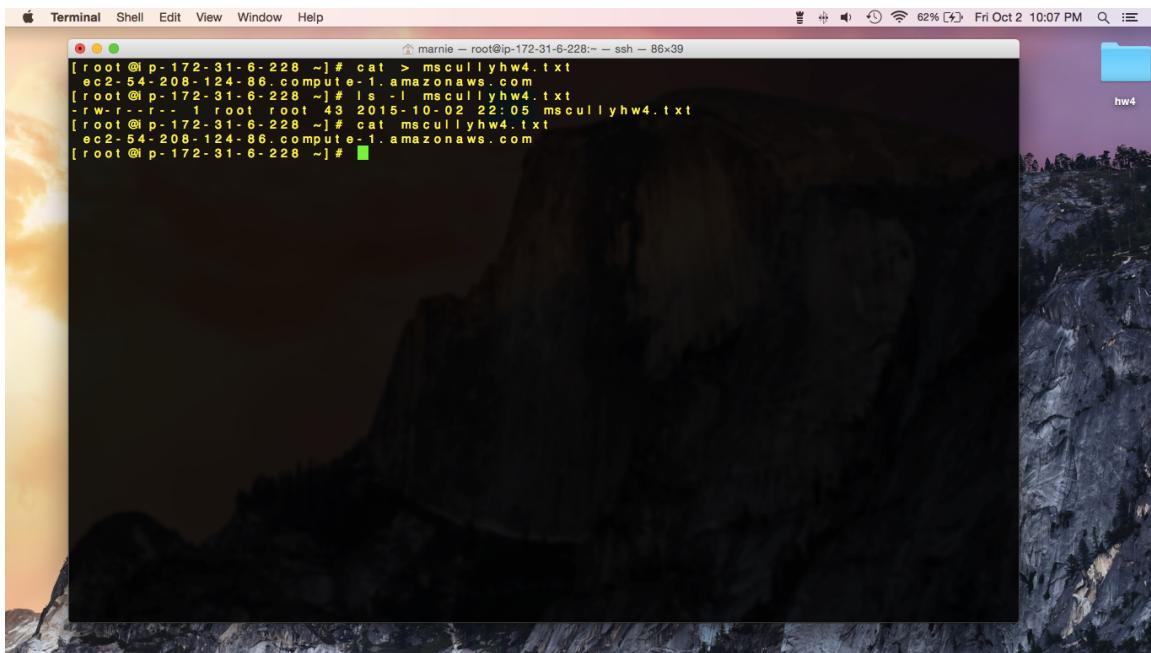
```
marnie - root@ip-172-31-6-229:~ - ssh - 86x39
Marnie's MacBook-Air:~ marnie$ ssh -i "CLI KeyPair.pem" root@54.208.221.69
The authenticity of host '54.208.221.69 (54.208.221.69)' can't be established.
RSA key fingerprint is d0:96:57:2d:62:cd:12:4f:25:se:f7:8d:b4:ac:d8:5f.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '54.208.221.69' (RSA) to the list of known hosts.

      _\   _ )   Fedora 8
     _\ / /_ 32-bit
     _\_\_\_/_\_
Welcome to an EC2 Public Image
      : - )
Getting Started
      with EBS Boot
-- [ see /etc/ec2/release-notes ] --
[root @ip-172-31-6-229 ~]#
```

Go through the process of demonstrating that you can move the EBS volume from one instance to another and back.

The following steps are for performing detach and attach volume operations:

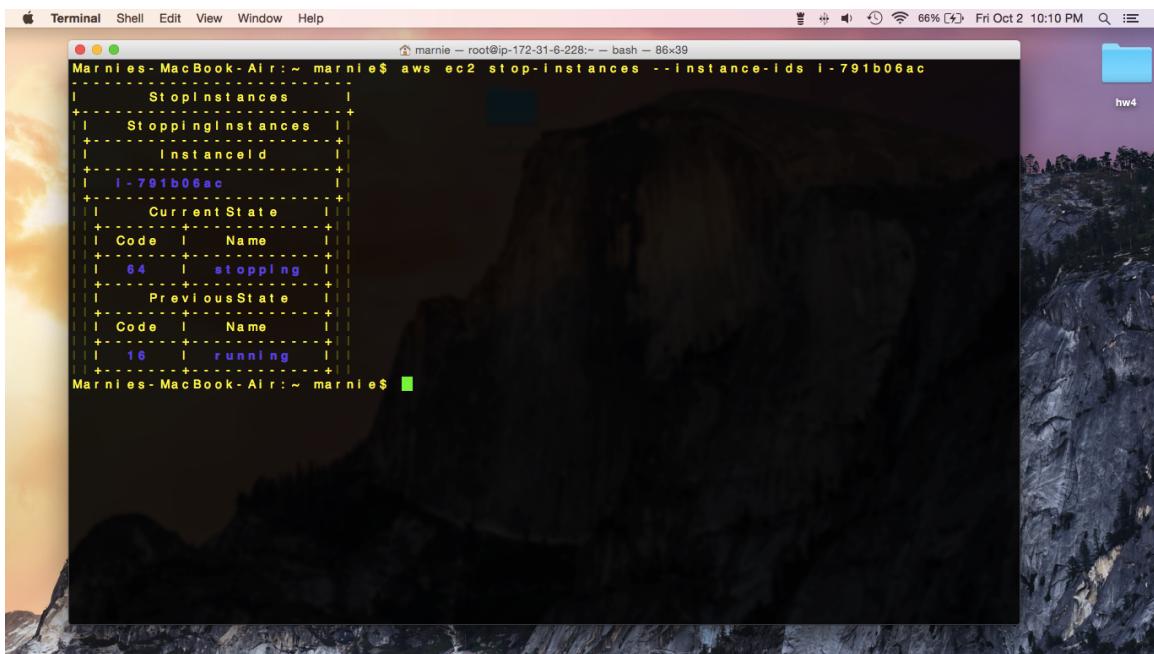
- f. First, add a single file in a directory (on one instance only) called: mscullyhw4.txt. and Write the DNS name of that instance into your file.



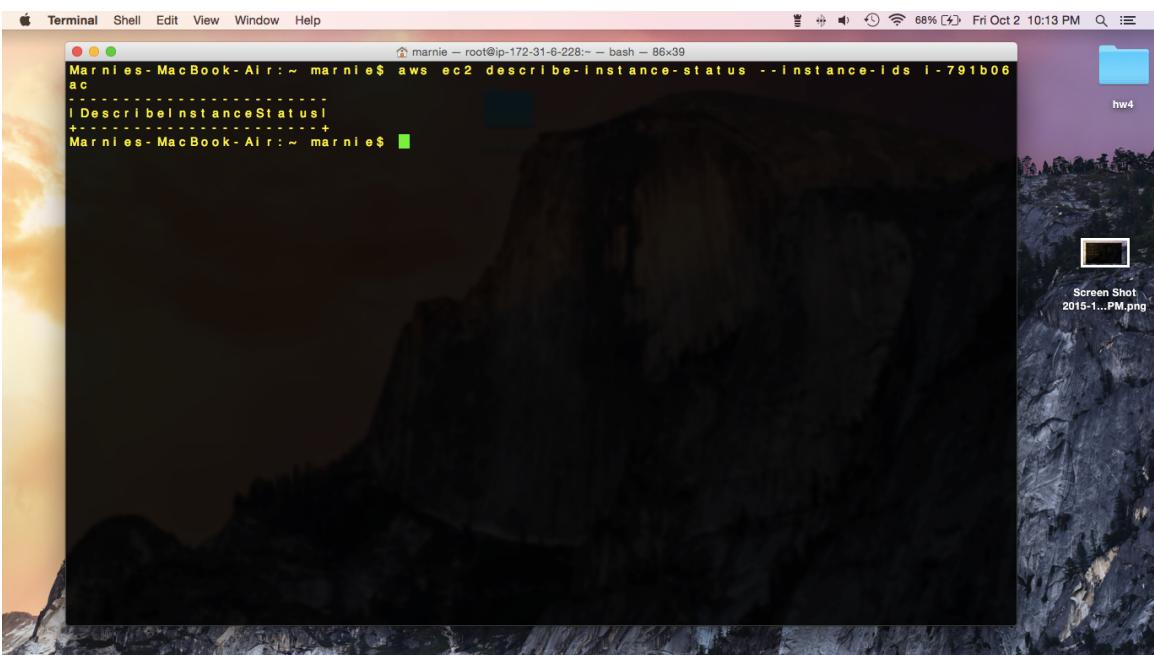
```
marnie - root@ip-172-31-6-228:~ - ssh - 86x39
[root @ ip-172-31-6-228 ~]# cat > mscullyhw4.txt
ec2-54-208-124-86.compute-1.amazonaws.com
[root @ ip-172-31-6-228 ~]# ls -l mscullyhw4.txt
-rw-r--r-- 1 root root 43 2015-10-02 22:05 mscullyhw4.txt
[root @ ip-172-31-6-228 ~]# cat mscullyhw4.txt
ec2-54-208-124-86.compute-1.amazonaws.com
[root @ ip-172-31-6-228 ~]#
```

Marnie Scully CSCIE 90 HW#4

g. Stop that instance (do not terminate it).



```
Marnie's-MacBook-Air:~ marnie$ aws ec2 stop-instances --instance-ids i-791b06ac
+-----+
| StoppingInstances |
+-----+
| InstanceId          |
| i-791b06ac          |
+-----+
| CurrentState        |
+-----+
| Code    | Name   |
+-----+-----+
| 64     | stopping |
+-----+
| PreviousState       |
+-----+
| Code    | Name   |
+-----+-----+
| 18     | running |
+-----+
Marnie's-MacBook-Air:~ marnie$
```



```
Marnie's-MacBook-Air:~ marnie$ aws ec2 describe-instance-status --instance-ids i-791b06ac
+-----+
| DescribeInstanceStatus |
+-----+
Marnie's-MacBook-Air:~ marnie$
```

Marnie Scully CSCIE 90 HW#4

h. Next, detach the volume.



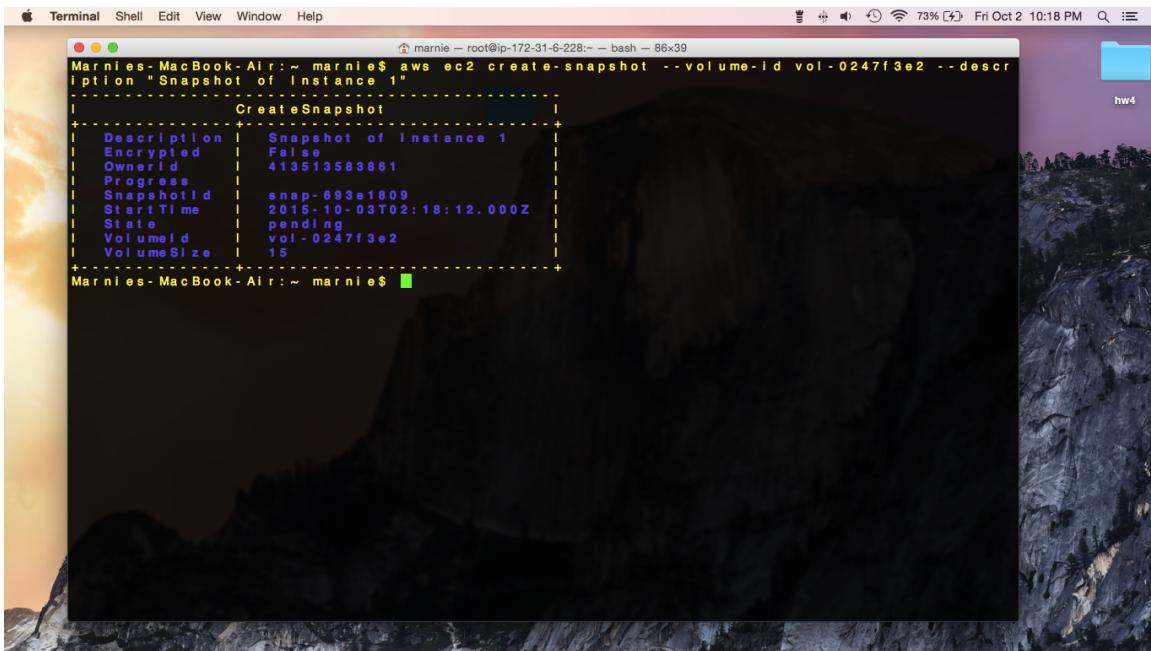
A screenshot of a macOS Terminal window titled "marnie". The window shows the command "aws ec2 detach-volume --volume-id vol-0247f3e2" being run. The output is a table titled "DetachVolume" with the following data:

AttachTime	Device	InstanceId	State	VolumeId
2015-10-03T01:08:41.000Z	/dev/sda1	i-791b06ac	detaching	vol-0247f3e2

The terminal window has a dark background with a mountain landscape image. The title bar shows the date and time as "Fri Oct 2 10:15 PM". A file icon labeled "hw4" is visible in the top right corner of the desktop.

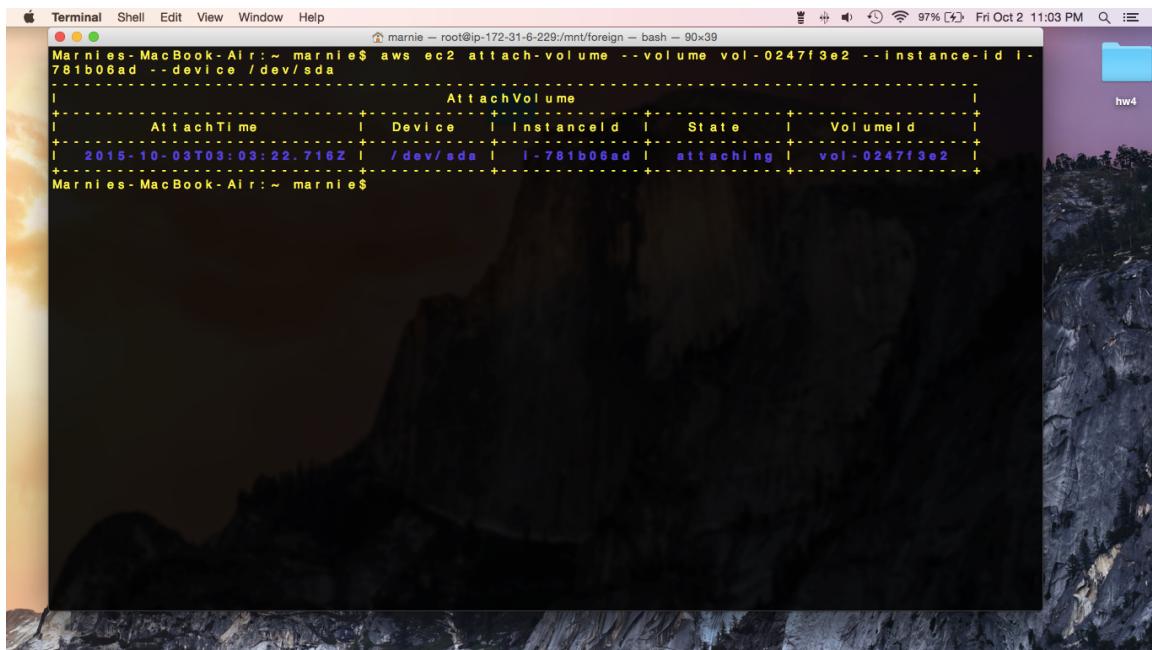
Marnie Scully CSCIE 90 HW#4

- i. Create a snapshot of the volume you just detached.



```
Marnie's-MacBook-Air:~ marnie$ aws ec2 create-snapshot --volume-id vol-0247f3e2 --description "Snapshot of Instance 1"
CreateSnapshot
+-----+
| Description | Snapshot of Instance 1
| Encrypted   | False
| OwnerId    | 413513583861
| Progress    |
| SnapshotId | snap-693e1809
| StartTime   | 2015-10-03T02:18:12.000Z
| State       | pending
| VolumeId   | vol-0247f3e2
| VolumeSize  | 15
+-----+
Marnie's-MacBook-Air:~ marnie$
```

- j. Subsequently, attach that same volume to the other instance.

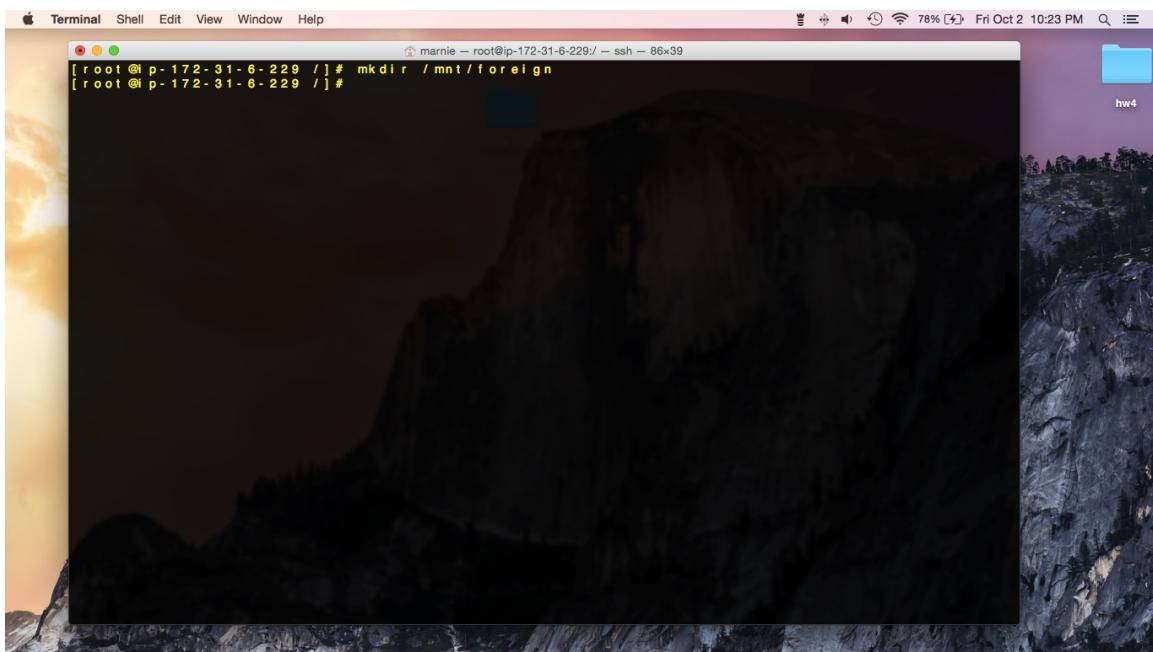


```
Marnie's-MacBook-Air:~ marnie$ aws ec2 attach-volume --volume vol-0247f3e2 --instance-id i-781b06ad --device /dev/sda
AttachVolume
+-----+
| AttachTime      | Device   | InstanceId | State   | VolumeId |
| 2015-10-03T03:03:22.716Z | /dev/sda | i-781b06ad | attaching | vol-0247f3e2 |
+-----+
Marnie's-MacBook-Air:~ marnie$
```

Marnie Scully CSCIE 90 HW#4

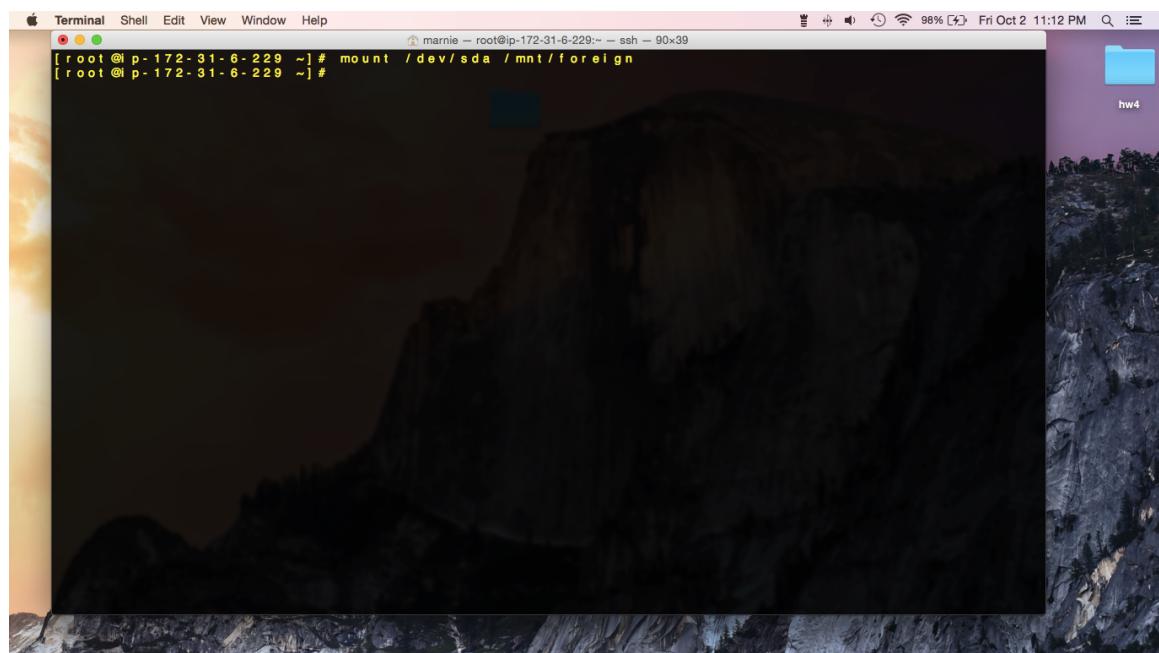
- k. Map that volume to a directory e.g. /mnt/foreign by first creating the directory on the new instance:

```
# mkdir /mnt/foreign
```



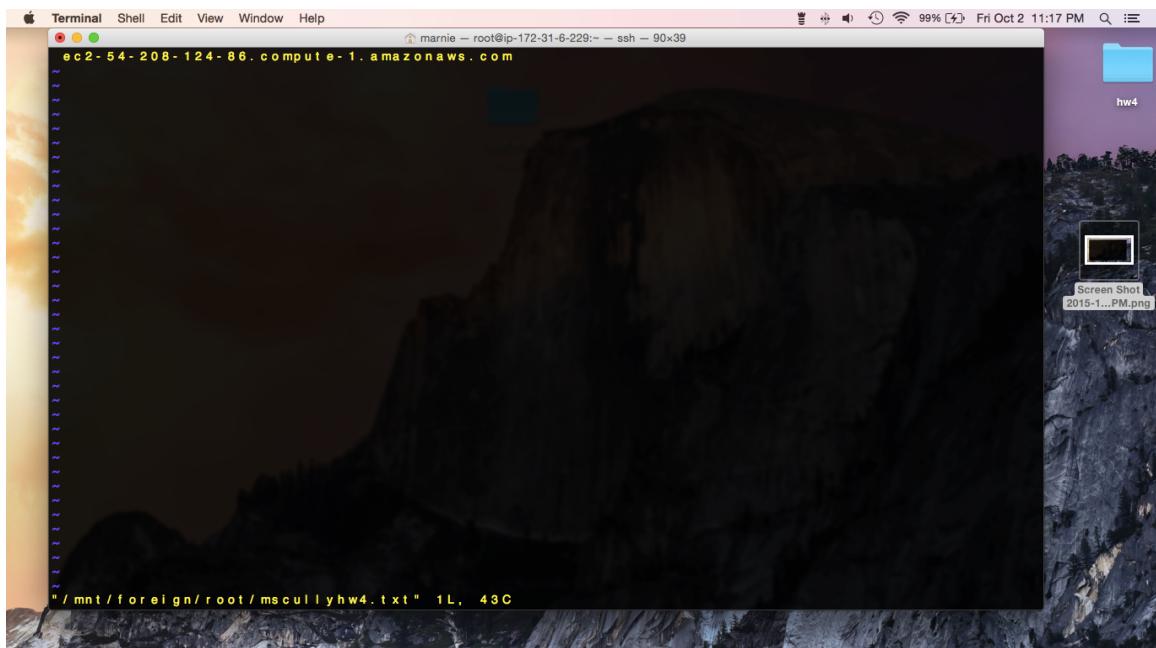
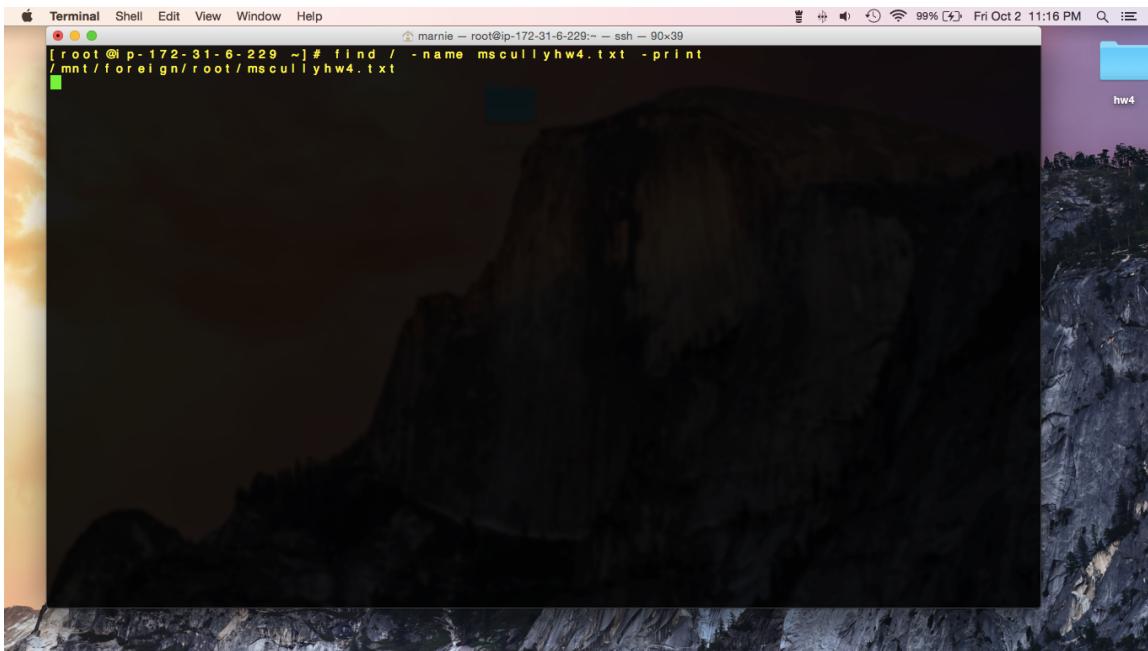
- l. Then MOUNT the attached drive to that directory, by issuing the command:

```
# mount /dev/sda /mnt/foreign
```



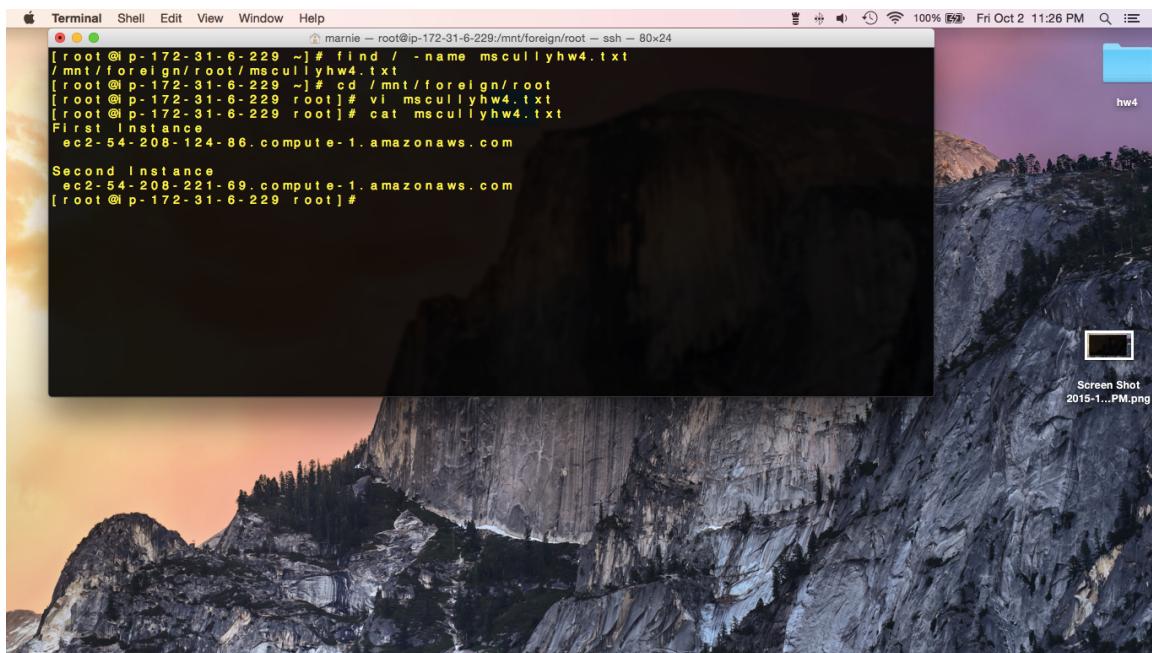
Marnie Scully CSCIE 90 HW#4

m. Verify that, on the second instance, you can see the file you added on the original instance (mscullyhw4.txt). Hint: Unix find / -name mskullyhw4.txt -print then show the contents of the file.

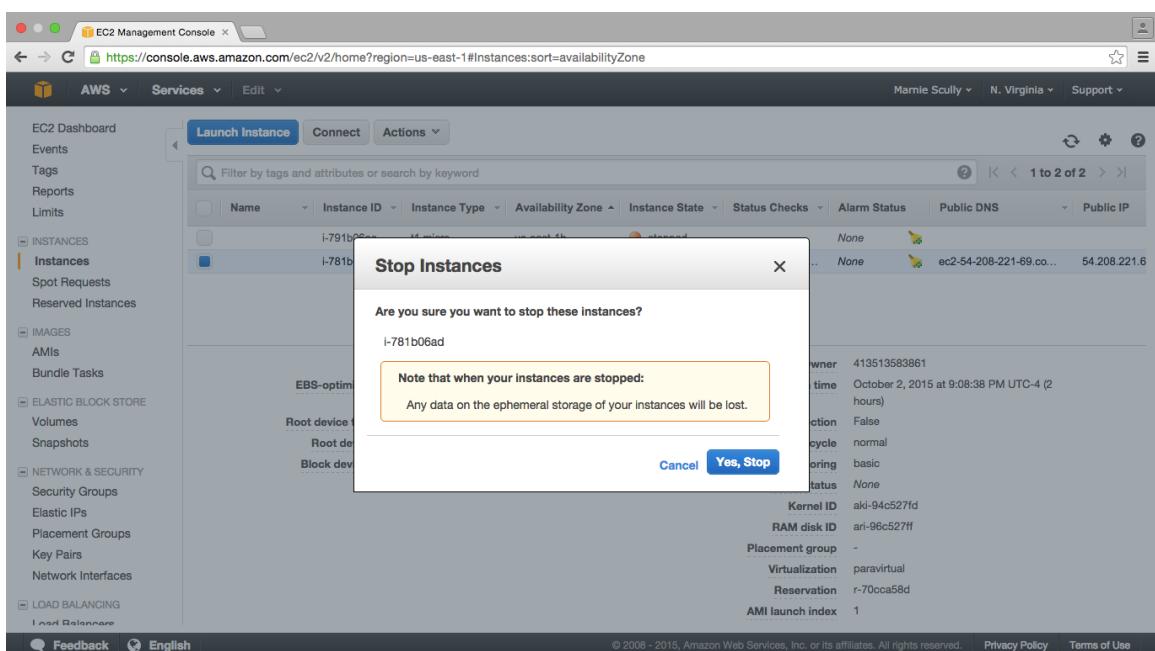


Marnie Scully CSCIE 90 HW#4

- n. Change that file (mscullyhw4.txt). For example, add the DNS name of the current host to it.



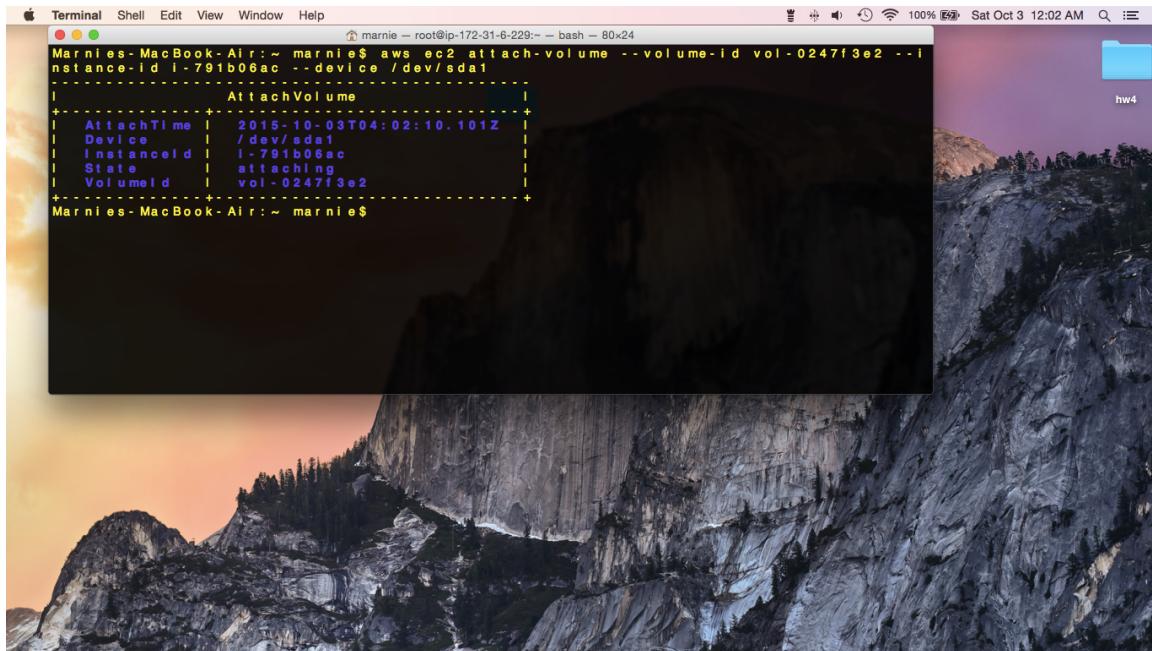
- o. Detach the volume from the active instance, this time do it the easy way through the AWS Console GUI. First stop the instance!



Marnie Scully CSCIE 90 HW#4

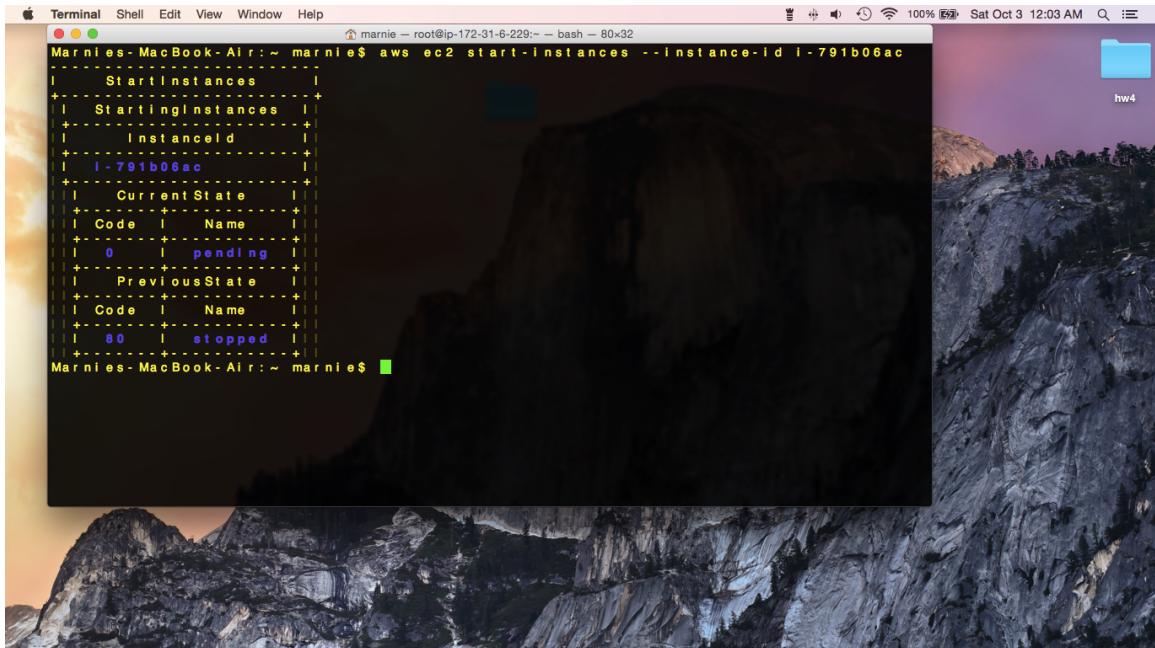
The screenshot shows the AWS EC2 Management Console. On the left, there's a sidebar with navigation links like EC2 Dashboard, Events, Tags, Reports, Limits, Instances, AMIs, Bundle Tasks, Elastic Block Store (with Volumes selected), Security Groups, Network & Security, and Load Balancing. The main area shows a table of volumes. A modal dialog box titled "Detach Volume" is open, asking "Are you sure you want to detach this volume? vol-0247f3e2". Below the dialog, the volume details are shown: Volume ID: i-781b06ad, Size: 15 GiB, Volume Type: standard, State: in-use, Attachment information: /dev/sda1 (attached). To the right of the dialog, the volume's status is listed: Alarm status: None, Snapshot: snap-a08912c9, Availability Zone: us-east-1b, Encrypted: Not Encrypted, KMS Key ID: None, KMS Key Aliases: None, and KMS Key ARN: None. At the bottom of the dialog are "Cancel" and "Yes, Detach" buttons.

- p. Attach the volume back to the original instance. Be careful to use the original device name e.g., /dev/sda1.



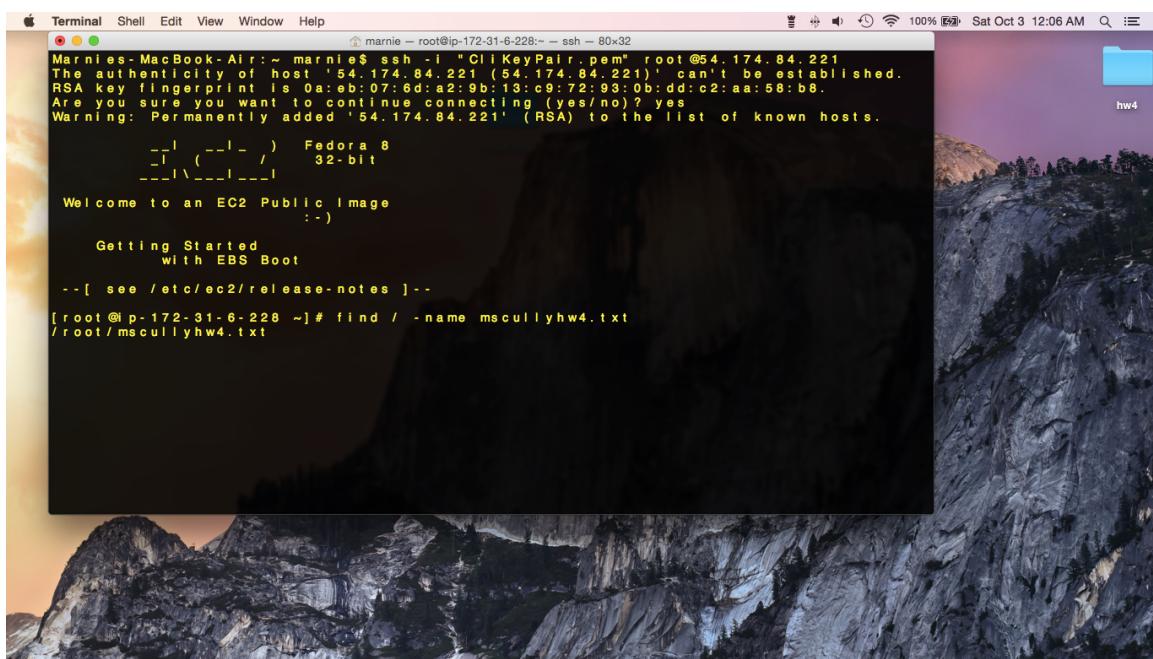
Marnie Scully CSCIE 90 HW#4

- q. Start the original instance.



```
marnie$ aws ec2 start-instances --instance-id i-791b06ac
StartingInstances
StartingInstances
InstanceID
i-791b06ac
CurrentState
Code Name
0 pending
PreviousState
Code Name
80 stopped
Marnie$
```

- r. Verify that you can see modifications made while the volume was attached to the other instance. Log into your instance and open your file mscullyhw4.txt. Hint: find / -name anynamehere.txt -print. Also show the contents of your file.



```
marnie$ ssh -i "CliKeyPair.pem" root@54.174.84.221
The authenticity of host '54.174.84.221 (54.174.84.221)' can't be established.
RSA key fingerprint is 0a:eb:07:6d:a2:9b:13:c9:72:93:0b:dd:c2:aa:58:b8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '54.174.84.221' (RSA) to the list of known hosts.

--( -- )  Fedora 8
--( / 32-bit
--( -- )

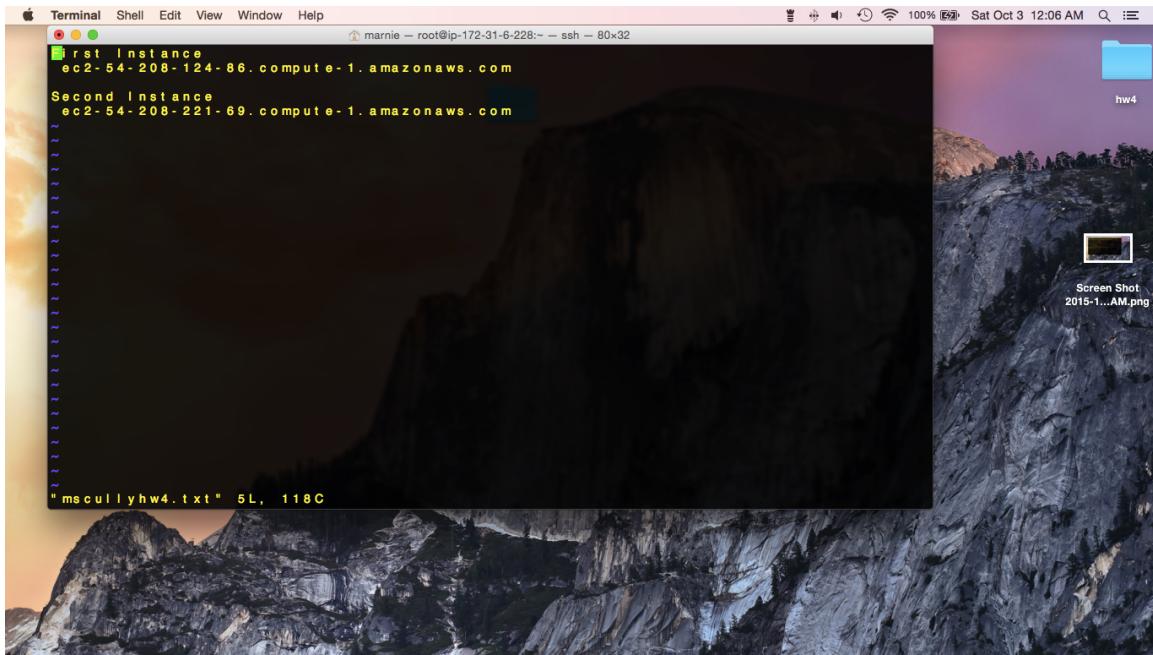
Welcome to an EC2 Public Image
:)

Getting Started
with EBS Boot

--[ see /etc/ec2/release-notes ]--

[root@ip-172-31-6-228 ~]# find / -name mscullyhw4.txt
/root/mscullyhw4.txt
```

Marnie Scully CSCIE 90 HW#4

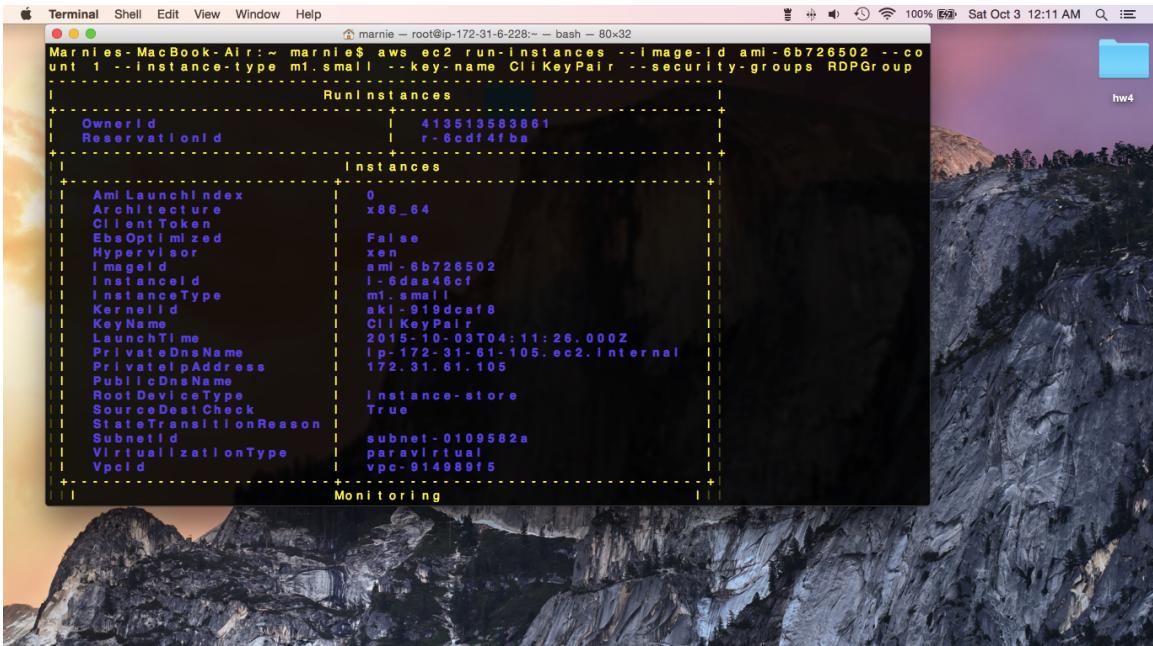


[Total Points: 50]

2. Use AWS CLI commands for all steps.

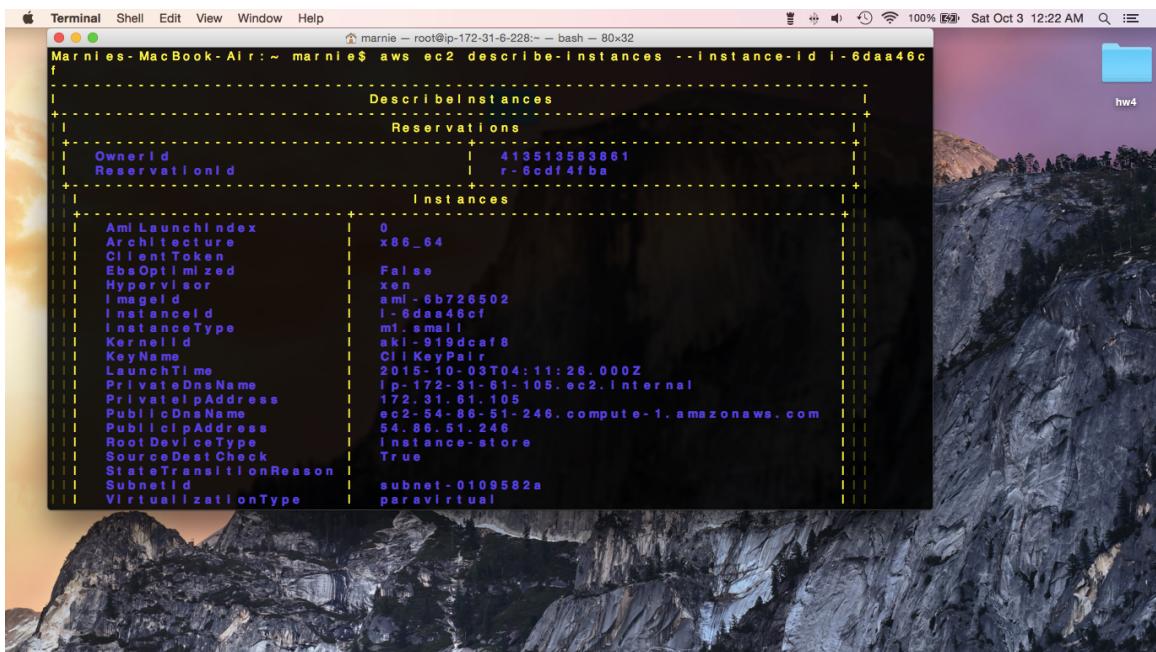
Create an instance of Amazon Linux ami-6b726502 and create a new user ec2test on that instance following lecture Sept 26, 2015 (Adding non-root Linux User). Demonstrate that you can connect to the instance as user ec2test and create a file.

- Create an instance from the selected AMI.



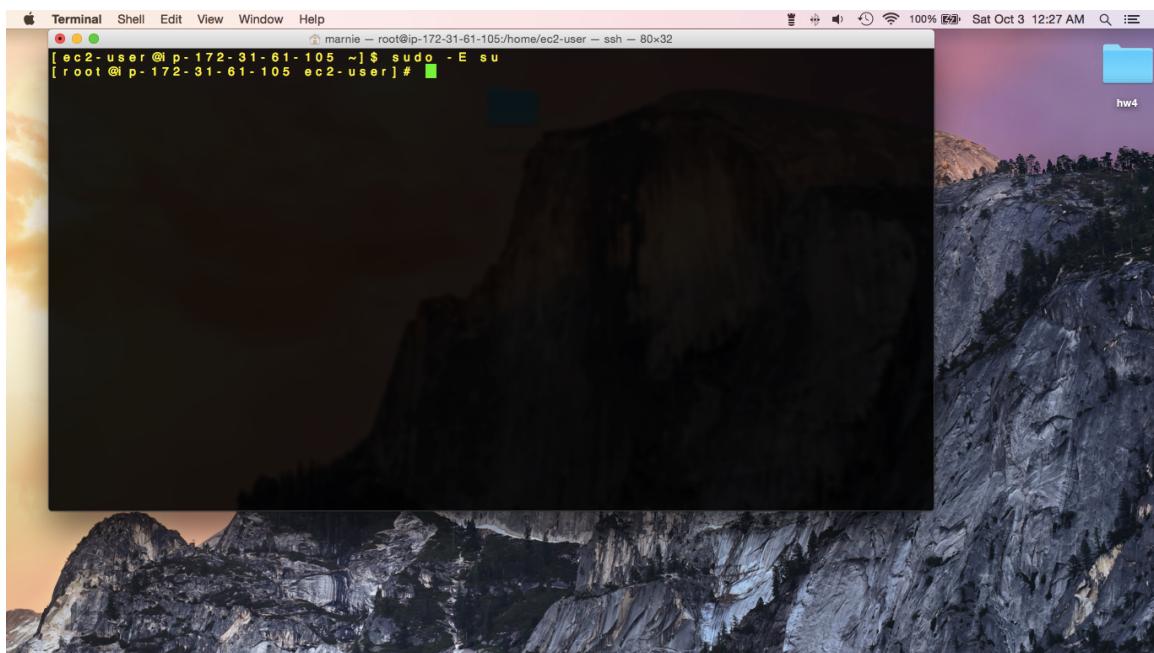
Marnie Scully CSCIE 90 HW#4

b. Describe the instance.



```
marnie - root@ip-172-31-6-228:~ bash 80x32
Marnie's MacBook-Air:~ marnie$ aws ec2 describe-instances --instance-id i-6daa46c
{
    "Reservations": [
        {
            "OwnerId": "413513583861",
            "ReservationId": "r-6cdf4fba"
        }
    ],
    "Instances": [
        {
            "AmiLaunchIndex": 0,
            "Architecture": "x86_64",
            "ClientToken": "False",
            "Hypervisor": "xen",
            "ImageId": "ami-6b726502",
            "InstanceId": "i-6daa46cf",
            "InstanceType": "mt.small",
            "KernelId": "aki-919d4af8",
            "KeyName": "CLI KeyPair",
            "LaunchTime": "2015-10-03T04:11:26.000Z",
            "PrivateDnsName": "ip-172-31-61-105.ec2.internal",
            "PrivateIpAddress": "172.31.61.105",
            "PublicDnsName": "ec2-54-86-51-246.compute-1.amazonaws.com",
            "PublicIpAddress": "54.86.51.246",
            "RootDeviceType": "instance-store",
            "SourceDestCheck": "True",
            "StateTransitionReason": null,
            "SubnetId": "subnet-0109582a",
            "VirtualizationType": "paravirtual"
        }
    ]
}
```

c. Login to the instance as root.

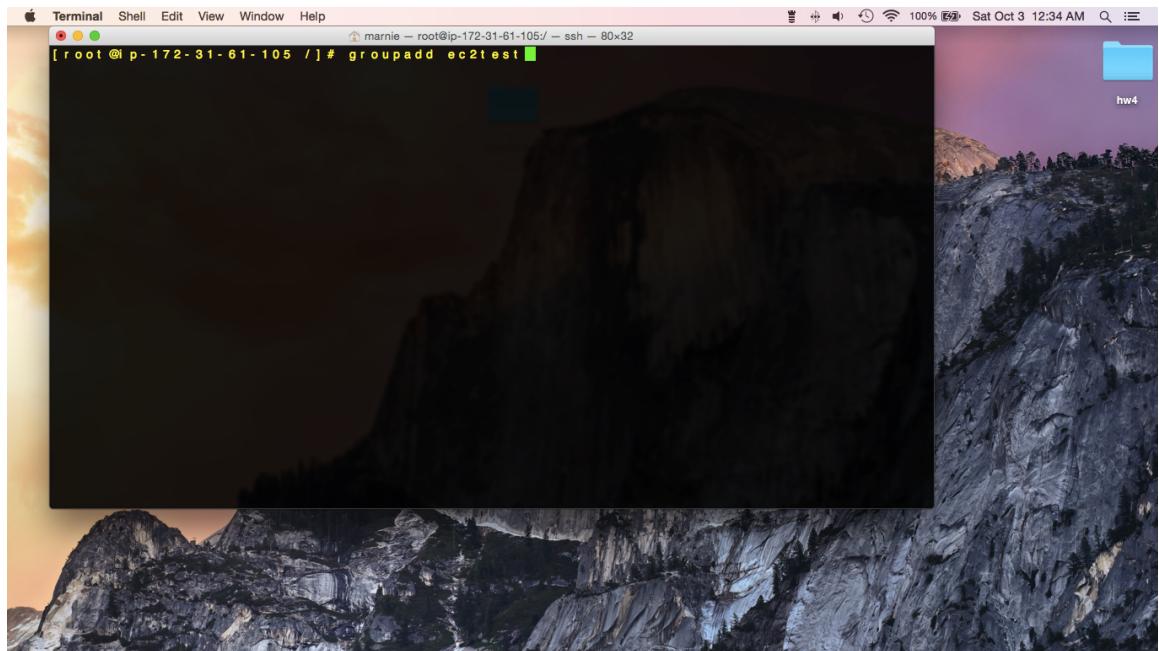


```
marnie - root@ip-172-31-61-105:~ ssh 80x32
[ec2-user@ip-172-31-61-105 ~]$ sudo -E su
[root@ip-172-31-61-105 ec2-user]# 
```

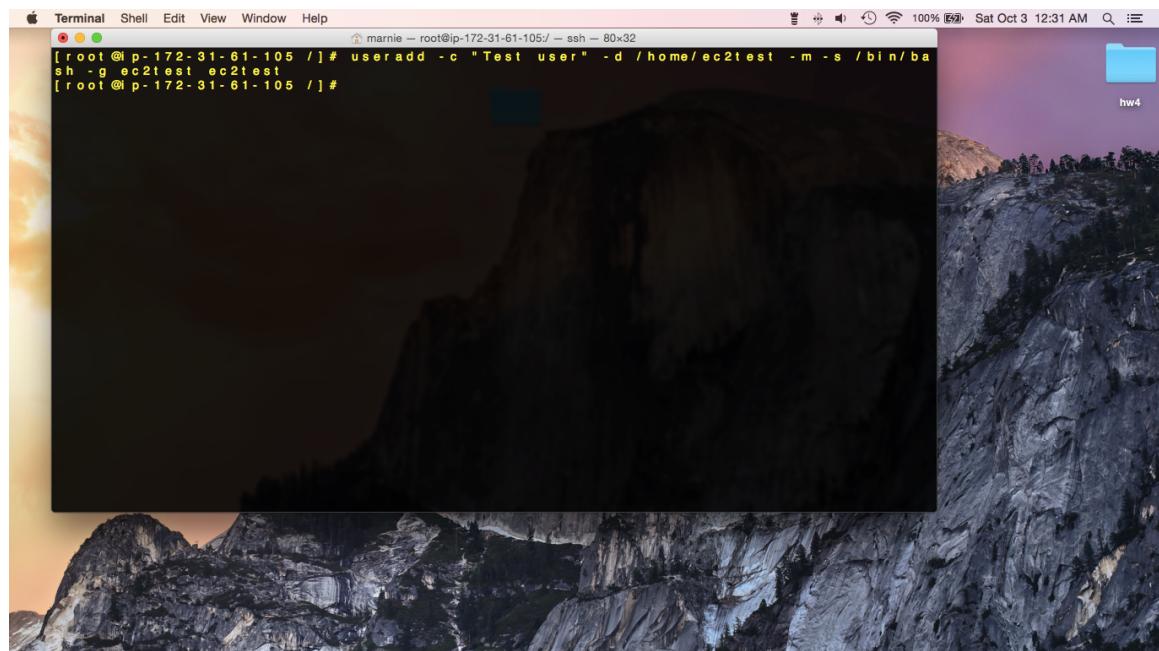
Marnie Scully CSCIE 90 HW#4

- d. Create new “ec2test” user: Create new group “ec2test” first. Add your new user to the group.

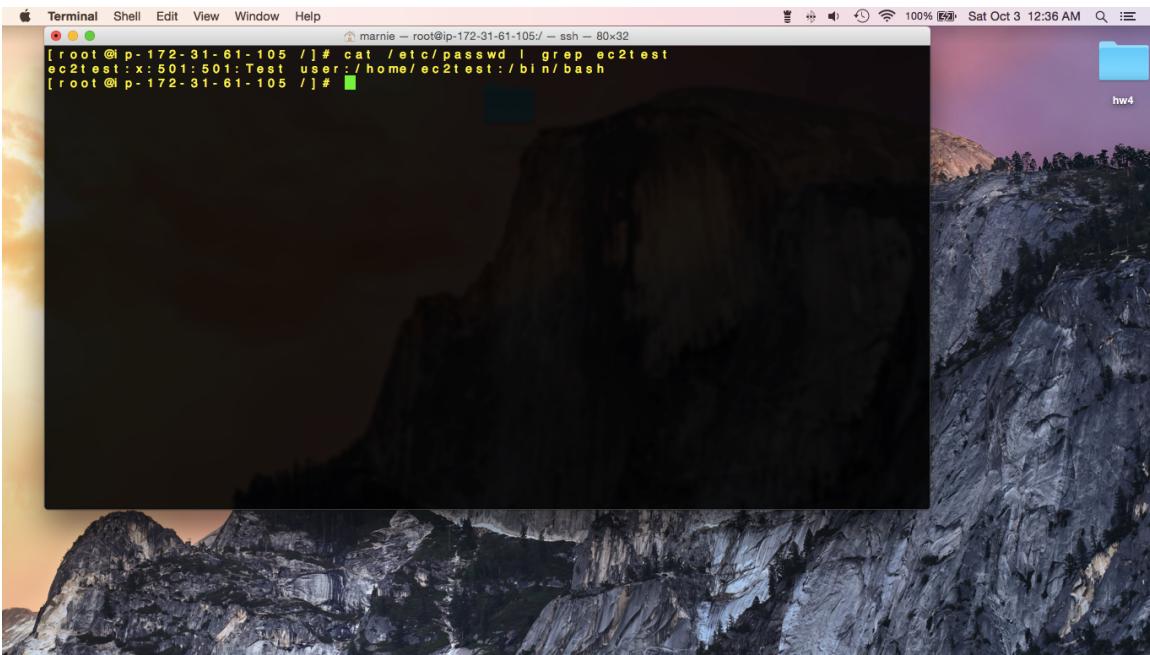
Create New Group ec2test



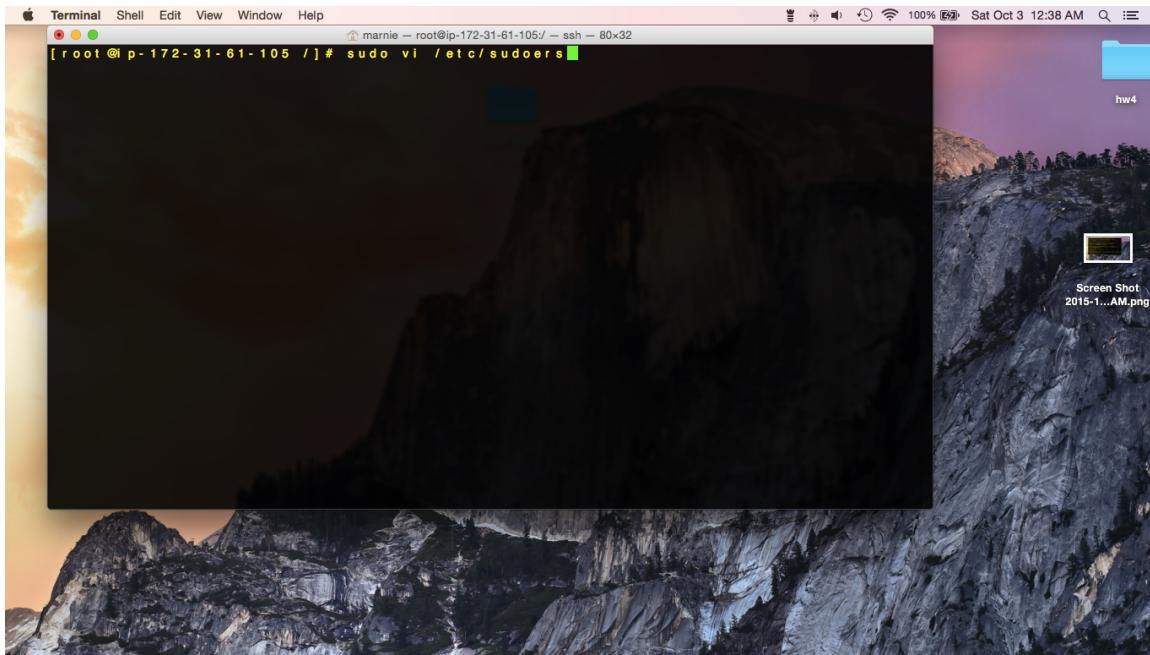
Add your new user to the group



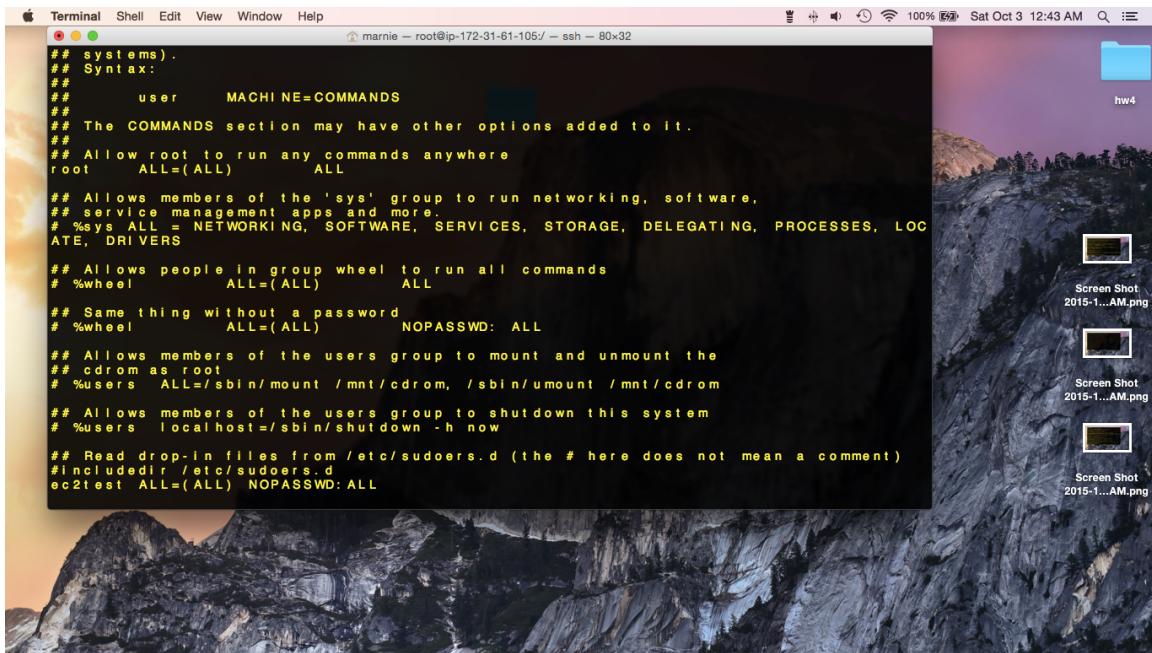
Marnie Scully CSCIE 90 HW#4



e. Make new user "ec2test" sudo user. (hint: /etc/sudoers)



Marnie Scully CSCIE 90 HW#4



```
marnie - root@ip-172-31-61-105:/ - ssh - 80x32
## systems).
## Syntax:
##      user      MACHINE=COMMANDS
## The COMMANDS section may have other options added to it.
## Allow root to run any commands anywhere
root    ALL=(ALL)        ALL

## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# %sys  ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOC
ATE, DRIVERS

## Allows people in group wheel to run all commands
# %wheel        ALL=(ALL)        ALL

## Same thing without a password
# %wheel        ALL=(ALL)        NOPASSWD: ALL

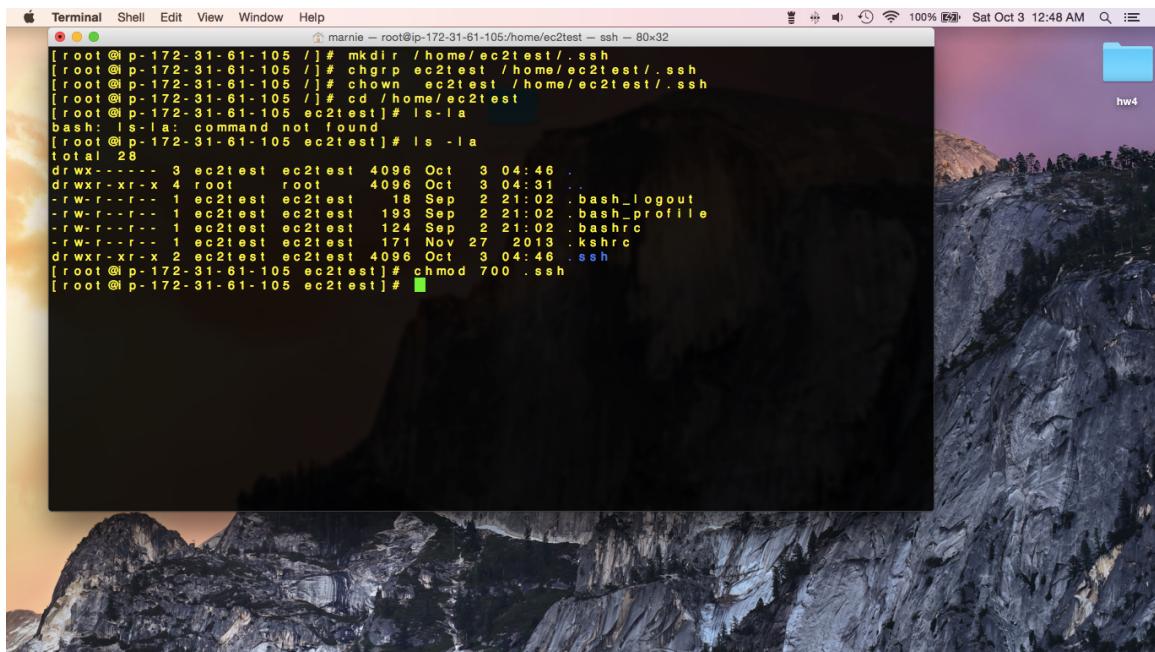
## Allows members of the users group to mount and umount the
## cdrom as root
# %users   ALL=/sbin/mount /mnt/cdrom, /sbin/umount /mnt/cdrom

## Allows members of the users group to shutdown this system
# %users   localhost=/sbin/shutdown -h now

## Read drop-in files from /etc/sudoers.d (the # here does not mean a comment)
#includedir /etc/sudoers.d
ec2test  ALL=(ALL)        NOPASSWD: ALL
```

f. Create directory .ssh for “ec2test” user.

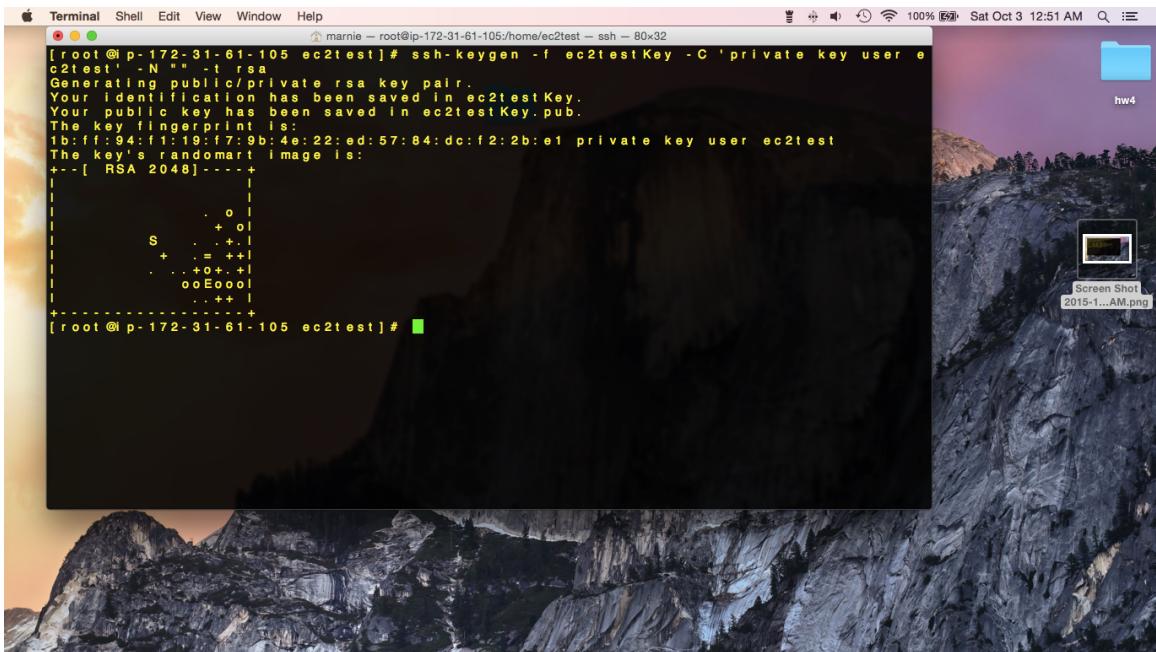
Give ec2test ownership and change the group to ec2test as well. Change permissions on .ssh directory



```
root@ip-172-31-61-105 ~]# mkdir /home/ec2test/.ssh
[root@ip-172-31-61-105 ~]# chgrp ec2test /home/ec2test/.ssh
[root@ip-172-31-61-105 ~]# chown ec2test /home/ec2test/.ssh
[root@ip-172-31-61-105 ~]# cd /home/ec2test
[root@ip-172-31-61-105 ec2test]# ls -la
bash: ls: command not found
[root@ip-172-31-61-105 ec2test]# ls -la
total 28
drwxr----- 3 ec2test ec2test 4096 Oct  3  04:46 .
drwxr-xr-x  4 root   root   4096 Oct  3  04:31 ..
-rw-r--r--  1 ec2test ec2test 16 Sep  2 21:02 .bash_logout
-rw-r--r--  1 ec2test ec2test 193 Sep  2 21:02 .bash_profile
-rw-r--r--  1 ec2test ec2test 224 Sep  2 21:02 .bashrc
-rw-r--r--  1 ec2test ec2test 114 Nov 22 2015 .kshrc
drwxr-xr-x  2 ec2test ec2test 4096 Oct  3  04:46 .ssh
[root@ip-172-31-61-105 ec2test]# chmod 700 .ssh
[root@ip-172-31-61-105 ec2test]#
```

Marnie Scully CSCIE 90 HW#4

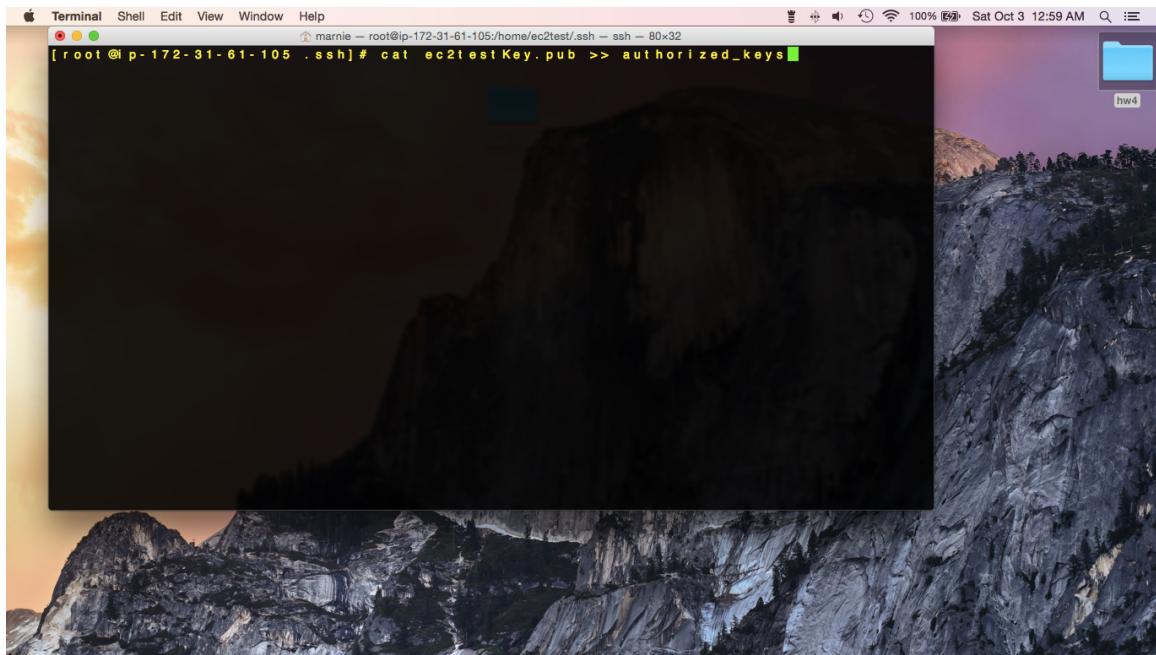
g. Generate key pair.



A screenshot of a Mac OS X terminal window titled "Terminal". The window shows the command `ssh-keygen -f ec2testKey -C 'private key user ec2test' -N "" -t rsa` being run by root. The output indicates that a public/private RSA key pair has been generated, with the private key saved as `ec2testKey` and the public key as `ec2testKey.pub`. The key's fingerprint is displayed as a series of hex digits. The desktop background features a scenic view of a mountain range at sunset.

```
marnie — root@ip-172-31-61-105:/home/ec2test — ssh — 80x32
[root@ip-172-31-61-105 ~]# ssh-keygen -f ec2testKey -C 'private key user ec2test' -N "" -t rsa
Generating public/private rsa key pair.
Your identification has been saved in ec2testKey.
Your public key has been saved in ec2testKey.pub.
The key fingerprint is:
1b:ff:94:f1:19:f7:9b:4e:22:ed:57:84:dc:f2:2b:e1 private key user ec2test
The key's randomart image is:
+-- [ RSA 2048 ] --+
| |
| |
| . o |
| + o |
| S . + . |
| + = ++ |
| . . +o+.+ |
| ooEooo |
| .++ |
+-----+
[root@ip-172-31-61-105 ~]#
```

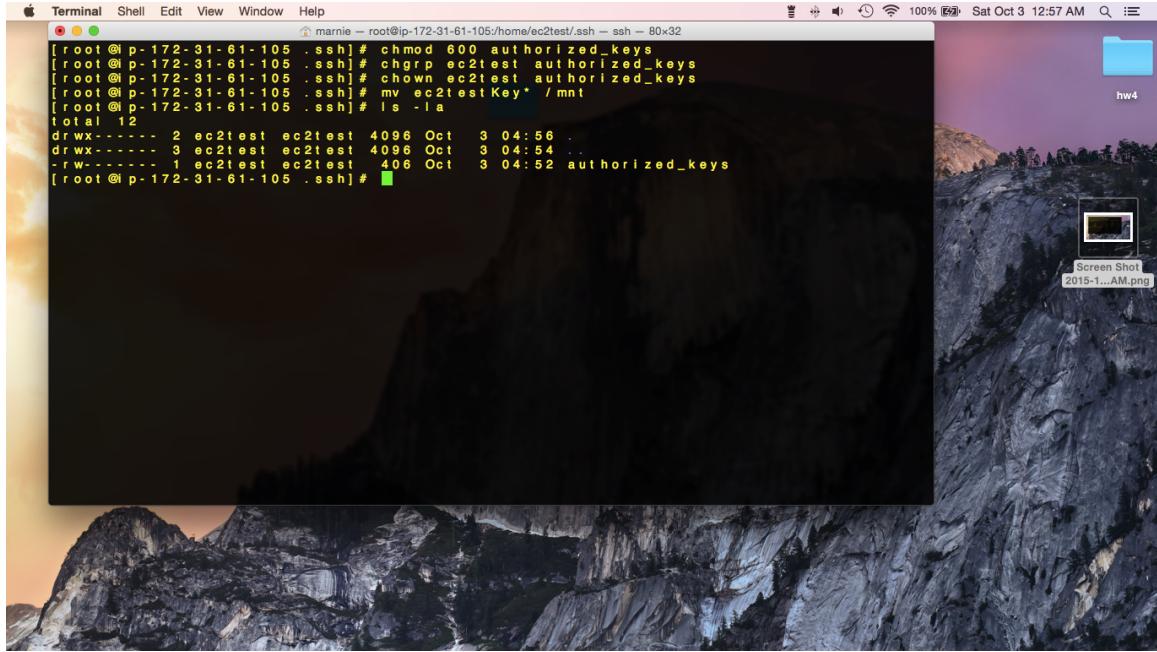
h. Push public key into file: authorized_keys.



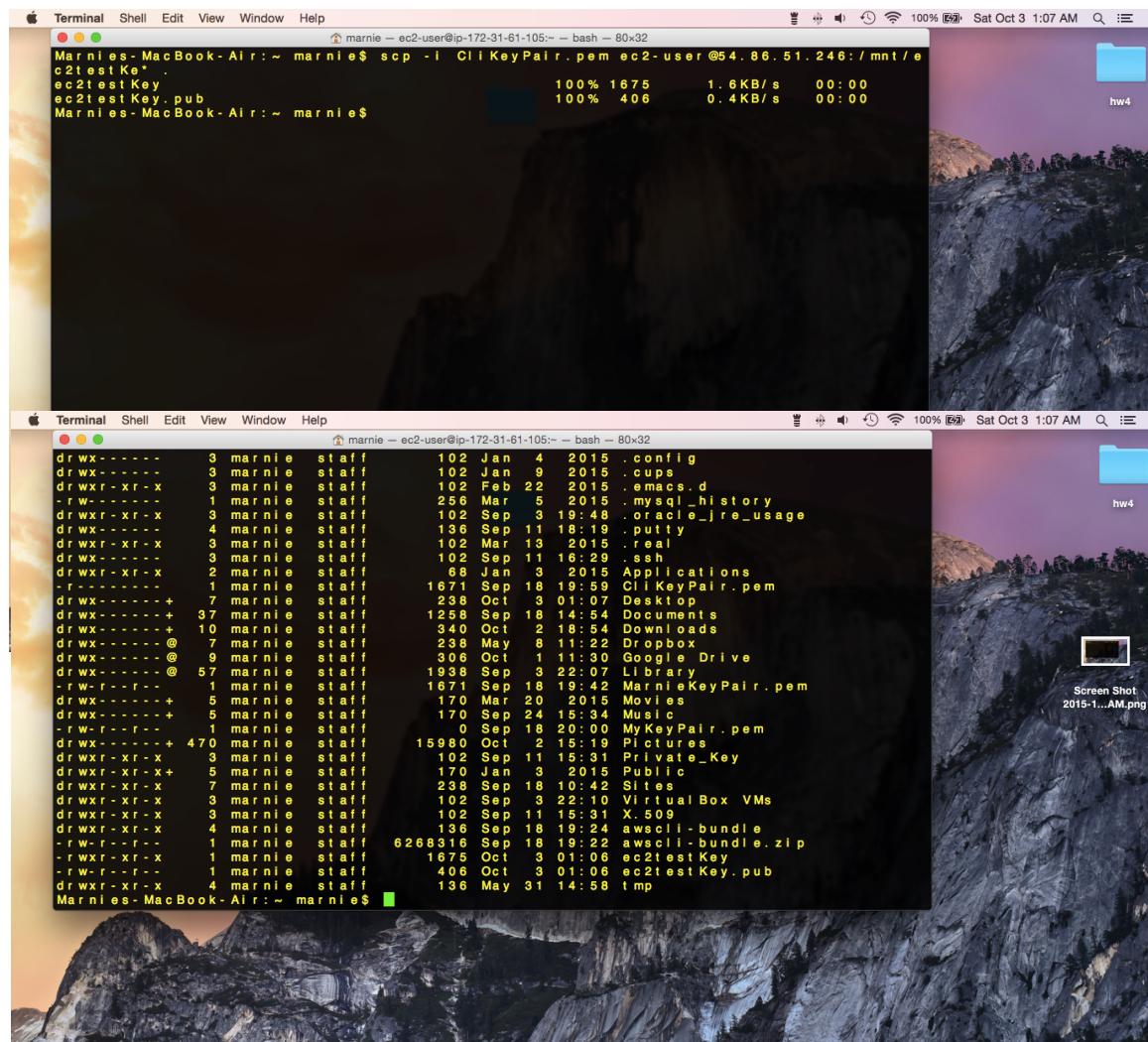
A screenshot of a Mac OS X terminal window titled "Terminal". The window shows the command `cat ec2testKey.pub >> authorized_keys` being run by root. The desktop background features a scenic view of a mountain range at sunset.

```
marnie — root@ip-172-31-61-105 ~.ssh# cat ec2testKey.pub >> authorized_keys
```

Marnie Scully CSCIE 90 HW#4

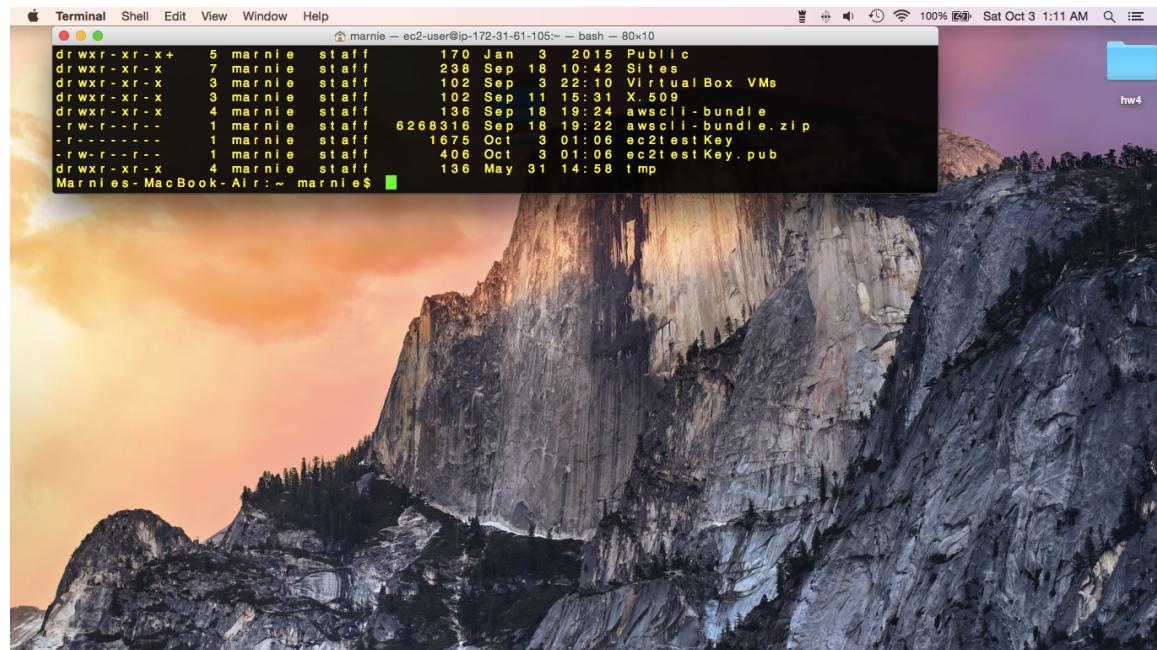


- i. Copy keys to your local Windows or Mac system.

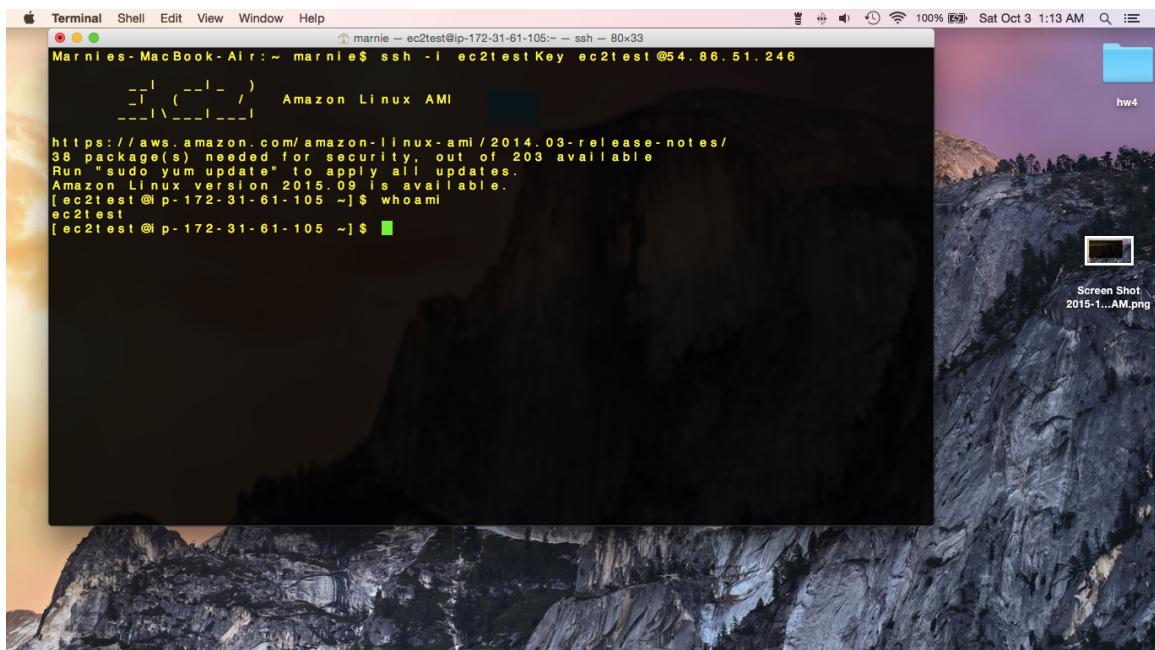


Marnie Scully CSCIE 90 HW#4

Change permissions of the key

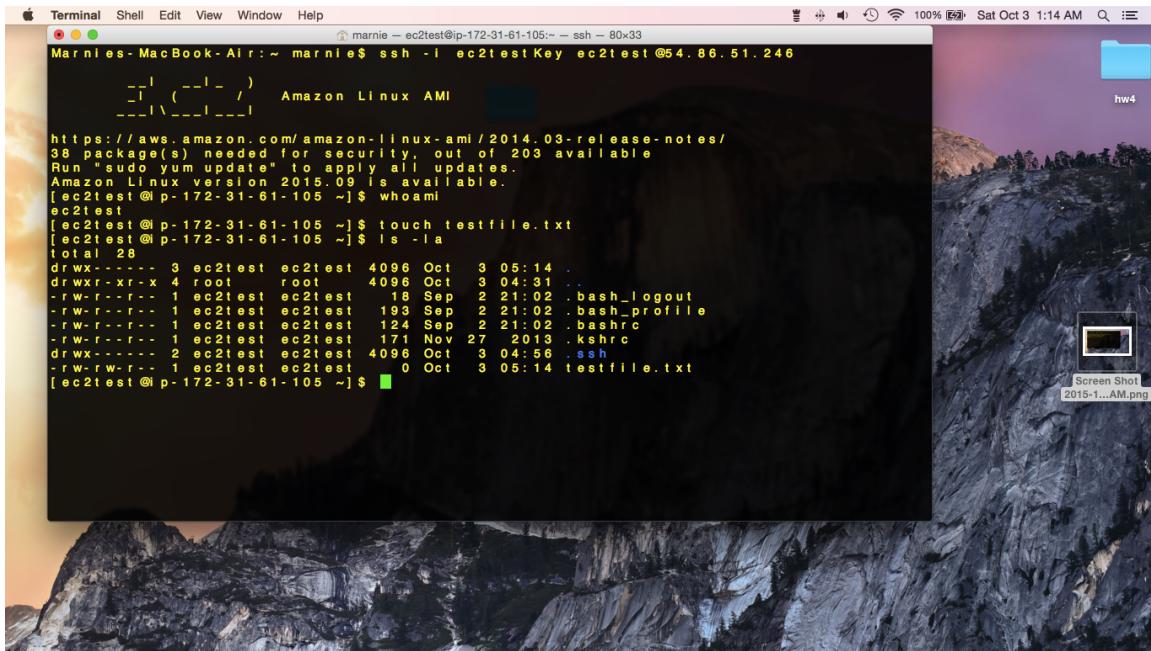


j. Connect to your instance as user "ec2test"



Marnie Scully CSCIE 90 HW#4

- k. Create a text file on your instance.
- l. Show permissions of the text file.



A screenshot of a Mac OS X desktop. In the foreground, a terminal window is open with the following command and output:

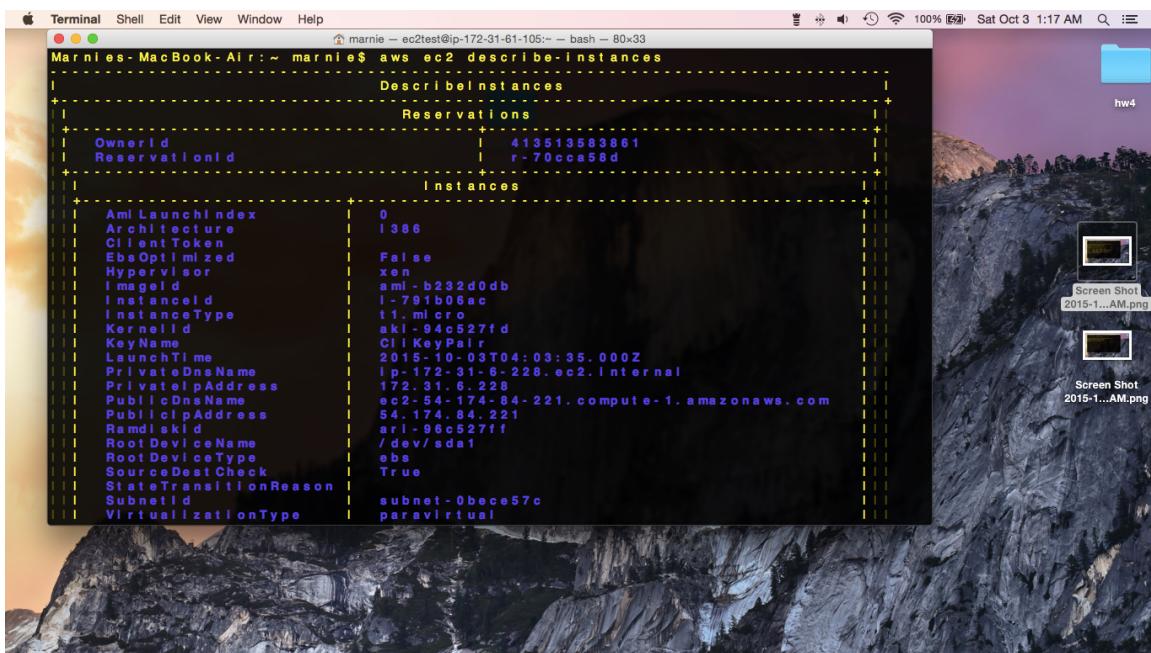
```
marnie$ ssh -i ec2testKey ec2test@54.86.51.246
Last login: Sat Oct 3 09:45:28 UTC 2015 from 54.86.51.246
[marnie ~]$ ls -la
total 28
drwx----- 3 ec2test ec2test 4096 Oct  3  05:14 .
drwxr-xr-x  4 root   root   4096 Oct  3  04:31 ..
-rw-r--r--  1 ec2test ec2test  18 Sep  2 21:02 .bash_logout
-rw-r--r--  1 ec2test ec2test 193 Sep  2 21:02 .bash_profile
-rw-r--r--  1 ec2test ec2test 124 Sep  2 21:02 .bashrc
-rw-r--r--  1 ec2test ec2test 171 Nov 27 2013 .kshrc
drwx----- 2 ec2test ec2test 4096 Oct  3  04:56 .ssh
-rw-rw-r--  1 ec2test ec2test    0 Oct  3  05:14 testfile.txt
[marnie ~]$
```

The terminal window has a dark background with white text. The desktop background is a scenic view of a mountain range at sunset. A file browser window titled "hw4" is visible in the top right corner, showing a folder icon and a thumbnail of a screenshot labeled "Screen Shot 2015-10-03 09.45.28 AM.png".

[Total Points: 40]

3. Use AWS CLI tools unless otherwise noted.
 - a. Show from your AWS Console your EC2 instances,

Instance i-791b06ac



A screenshot of a Mac OS X desktop. In the foreground, a terminal window is open with the following command and output:

```
marnie$ aws ec2 describe-instances
{
  "Reservations": [
    {
      "OwnerId": "413513583861",
      "ReservationId": "r-70ccaa8d",
      "Instances": [
        {
          "AmiLaunchIndex": 0,
          "Architecture": "x86_64",
          "ClientToken": null,
          "EbsOptimized": false,
          "Hypervisor": "xen",
          "ImageId": "ami-b232d0db",
          "InstanceId": "i-791b06ac",
          "InstanceType": "t1.micro",
          "KernelId": "aki-94c527fd",
          "KeyName": "ClfKeyPair",
          "LaunchTime": "2015-10-03T04:03:35.000Z",
          "PrivateDnsName": "ip-172-31-6-228.ec2.internal",
          "PrivateIpAddress": "172.31.6.228",
          "PublicDnsName": "ec2-54-174-84-221.compute-1.amazonaws.com",
          "PublicIpAddress": "54.174.84.221",
          "RootDeviceName": "ari-98c527ff",
          "RootDeviceType": "ebs",
          "SourceDestCheck": true,
          "StateTransitionReason": "subnet-0bece57c",
          "SubnetId": null,
          "VirtualizationType": "paravirtual"
        }
      ]
    }
  ]
}
```

The terminal window has a dark background with white text. The desktop background is a scenic view of a mountain range at sunset. A file browser window titled "hw4" is visible in the top right corner, showing a folder icon and two thumbnails of screenshots labeled "Screen Shot 2015-10-03 09.45.28 AM.png" and "Screen Shot 2015-10-03 09.45.29 AM.png".

Marnie Scully CSCIE 90 HW#4

Instance i-781b06ad

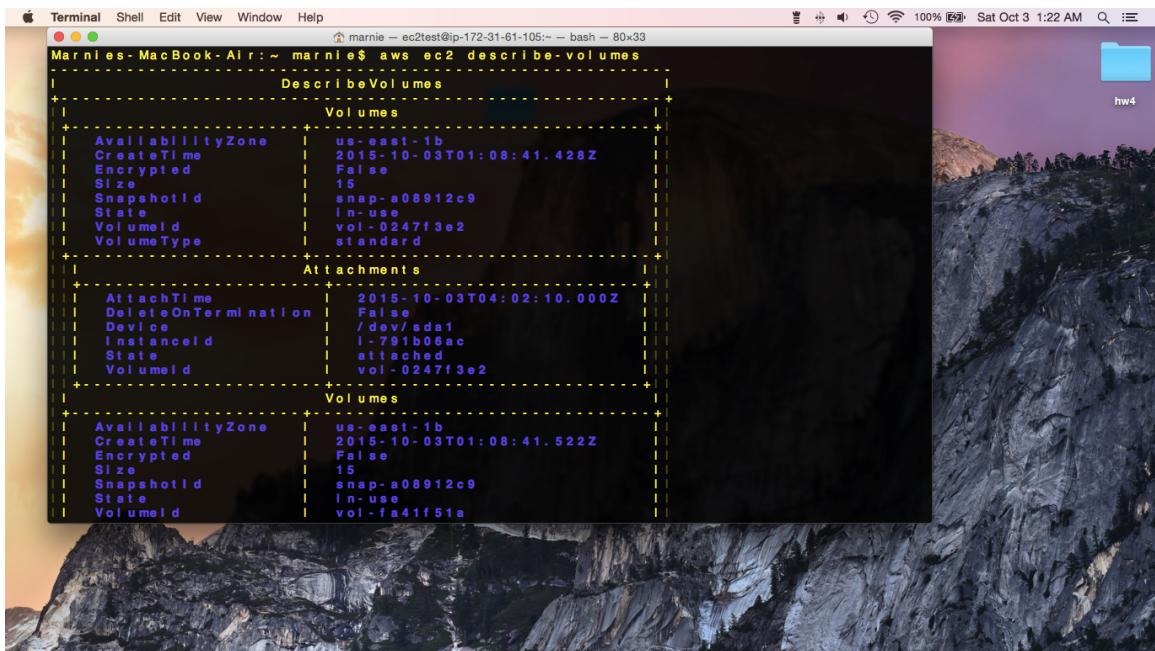
```
marnie@ec2test:~$ aws ec2 describe-instances --instance-ids i-781b06ad
{
    "Instances": [
        {
            "AmiLaunchIndex": 1,
            "Architecture": "x86_64",
            "ClientToken": "False",
            "Hypervisor": "xen",
            "ImageId": "ami-b232d0db",
            "InstanceId": "i-781b06ad",
            "InstanceType": "t1.micro",
            "KernelId": "aki-94c527fd",
            "KeyName": "CLI KeyPair",
            "LaunchTime": "2015-10-03T03:45:49.000Z",
            "PrivateDnsName": "ip-172-31-6-229.ec2.internal",
            "PrivateIpAddress": "172.31.6.229",
            "PublicDnsName": "ec2-54-172-215-144.compute-1.amazonaws.com",
            "PublicIpAddress": "54.172.215.144",
            "RamdiskId": "ari-96c527ff",
            "RootDeviceName": "/dev/sda1",
            "RootDeviceType": "ebs",
            "SourceDestCheck": "True",
            "StateTransitionReason": "",
            "SubnetId": "subnet-0bece57c",
            "VirtualizationType": "paravirtual",
            "VpcId": "vpc-914989f5"
        }
    ],
    "BlockDeviceMappings": [
        {
            "DeviceName": "/dev/sda1",
            "Ebs": {}
        }
    ]
}
```

Instance i-6daa46cf

```
marnie@ec2test:~$ aws ec2 describe-instances --instance-ids i-6daa46cf
{
    "Instances": [
        {
            "AmiLaunchIndex": 0,
            "Architecture": "x86_64",
            "ClientToken": "False",
            "Hypervisor": "xen",
            "ImageId": "ami-6b726502",
            "InstanceId": "i-6daa46cf",
            "InstanceType": "m1.small",
            "KernelId": "aki-919dc0af",
            "KeyName": "CLI KeyPair",
            "LaunchTime": "2015-10-03T04:11:26.000Z",
            "PrivateDnsName": "ip-172-31-61-105.ec2.internal",
            "PrivateIpAddress": "172.31.61.105",
            "PublicDnsName": "ec2-54-86-51-246.compute-1.amazonaws.com",
            "PublicIpAddress": "54.86.51.246",
            "RootDeviceType": "instance-store",
            "SourceDestCheck": "True",
            "StateTransitionReason": "",
            "SubnetId": "subnet-0109582a",
            "VirtualizationType": "paravirtual",
            "VpcId": "vpc-914989f5"
        }
    ],
    "Monitoring": {
        "State": "disabled"
    },
    "NetworkInterfaces": [
        {
            "Description": "eth0",
            "MacAddress": "12:2b:70:41:f0:fb"
        }
    ]
}
```

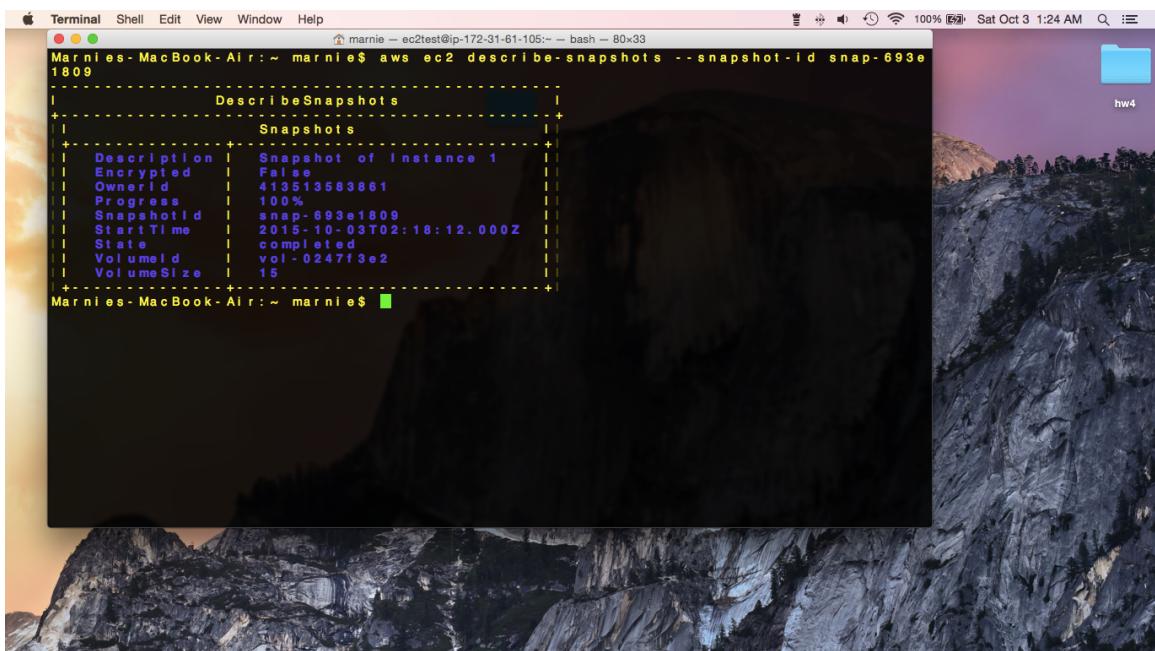
Marnie Scully CSCIE 90 HW#4

volumes



```
Terminal Shell Edit View Window Help
marnie — ec2test@ip-172-31-61-105:~ bash — 80x33
Marnie's MacBook Air:~ marnie$ aws ec2 describe-volumes
{
    "DescribeVolumes": {
        "Volumes": [
            {
                "AvailabilityZone": "us-east-1b",
                "CreateTime": "2015-10-03T01:08:41.428Z",
                "Encrypted": false,
                "Size": 15,
                "SnapshotId": "snap-a08912c9",
                "State": "in-use",
                "VolumeId": "vol-0247f3e2",
                "VolumeType": "standard"
            },
            {
                "Attachments": [
                    {
                        "AttachTime": "2015-10-03T04:02:10.000Z",
                        "DeleteOnTermination": false,
                        "Device": "/dev/sda1",
                        "InstanceId": "i-791b06ac",
                        "State": "attached",
                        "VolumeId": "vol-0247f3e2"
                    }
                ],
                "Volumes": [
                    {
                        "AvailabilityZone": "us-east-1b",
                        "CreateTime": "2015-10-03T01:08:41.522Z",
                        "Encrypted": false,
                        "Size": 15,
                        "SnapshotId": "snap-a08912c9",
                        "State": "in-use",
                        "VolumeId": "vol-fa41f51a"
                    }
                ]
            }
        ]
    }
}
```

and snapshots that remain running.

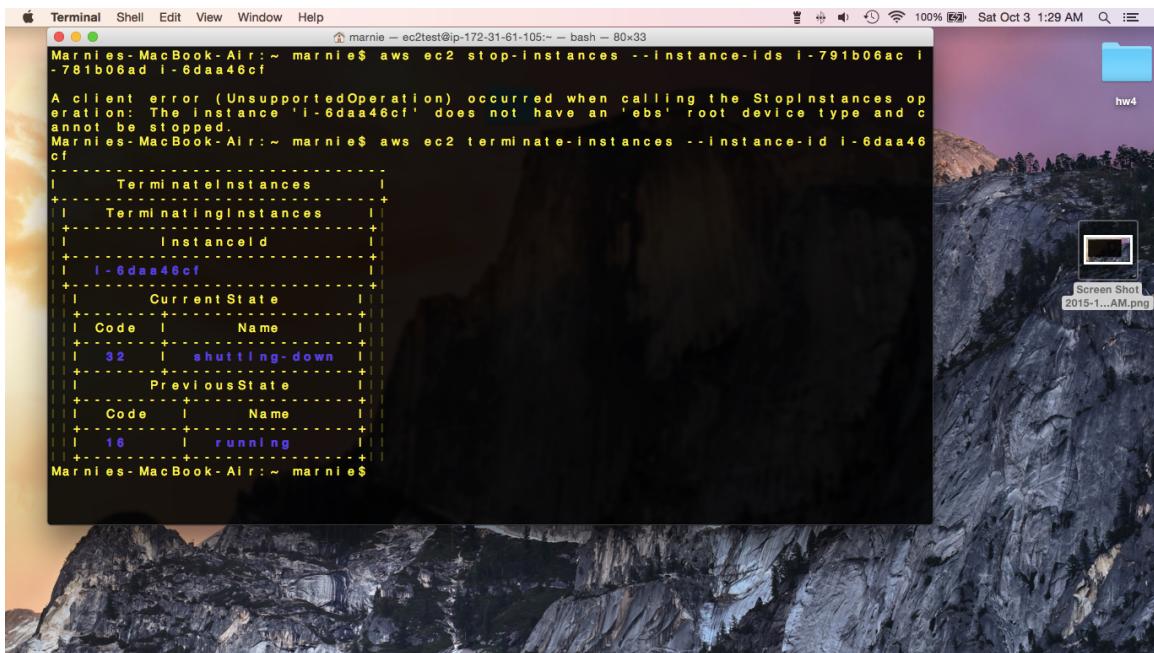


```
Terminal Shell Edit View Window Help
marnie — ec2test@ip-172-31-61-105:~ bash — 80x33
Marnie's MacBook Air:~ marnie$ aws ec2 describe-snapshots --snapshot-id snap-693e1809
{
    "DescribeSnapshots": {
        "Snapshots": [
            {
                "Description": "Snapshot of Instance 1",
                "Encrypted": false,
                "OwnerId": "413513583861",
                "Progress": "100%",
                "SnapshotId": "snap-693e1809",
                "StartTime": "2015-10-03T02:18:12.000Z",
                "State": "completed",
                "VolumeId": "vol-0247f3e2",
                "VolumeSize": 15
            }
        ]
    }
}
Marnie's MacBook Air:~ marnie$
```

Marnie Scully CSCIE 90 HW#4

- b. Stop your EC2 instances and delete volumes and your snapshot once you are done with the assignment using AWS CLI tools.

Stop the EBS instances and terminate the Instance Store instance

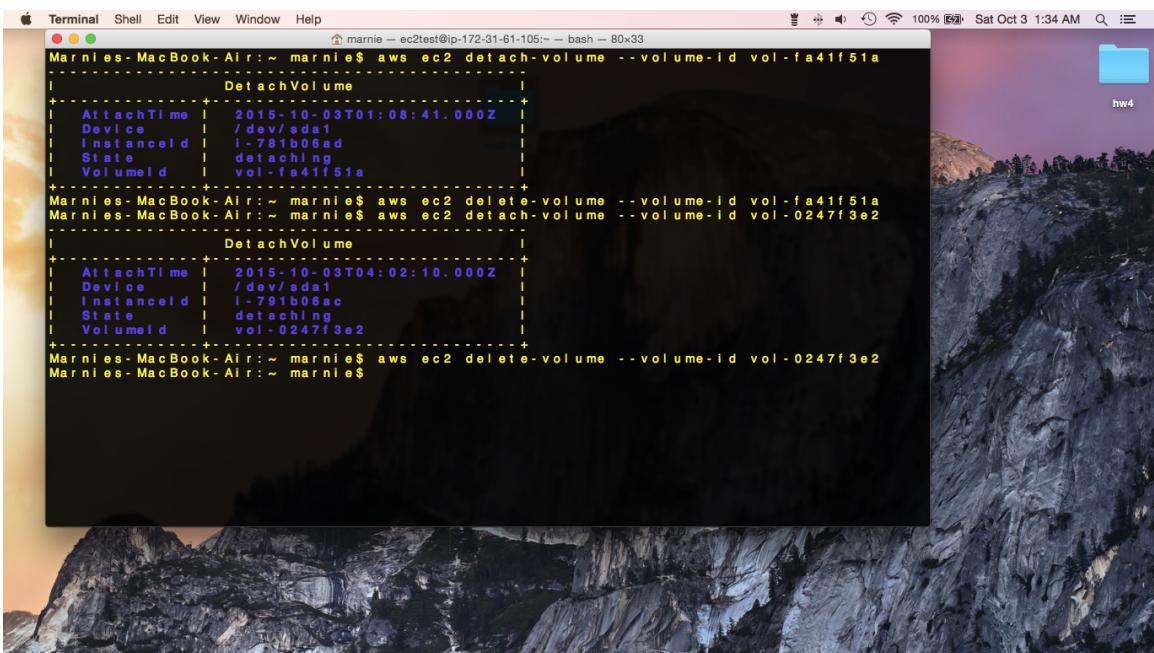


A screenshot of a Mac OS X desktop. In the foreground, a terminal window is open with the following command and output:

```
marnie$ aws ec2 stop-instances --instance-ids i-781b06ac i-781b06ad i-6daa46cf
A client error (UnsupportedOperation) occurred when calling the StopInstances operation: The instance 'i-6daa46cf' does not have an 'ebs' root device type and cannot be stopped.
marnie$ aws ec2 terminate-instances --instance-id i-6daa46cf
[Terminal output redacted]
marnie$
```

The desktop background is a scenic view of a mountain range at sunset. A file browser window titled "hw4" is visible in the top right corner, containing a folder icon and a thumbnail of a screenshot named "Screen Shot 2015-10-03 01:34 AM.png".

Detach and Delete Volumes

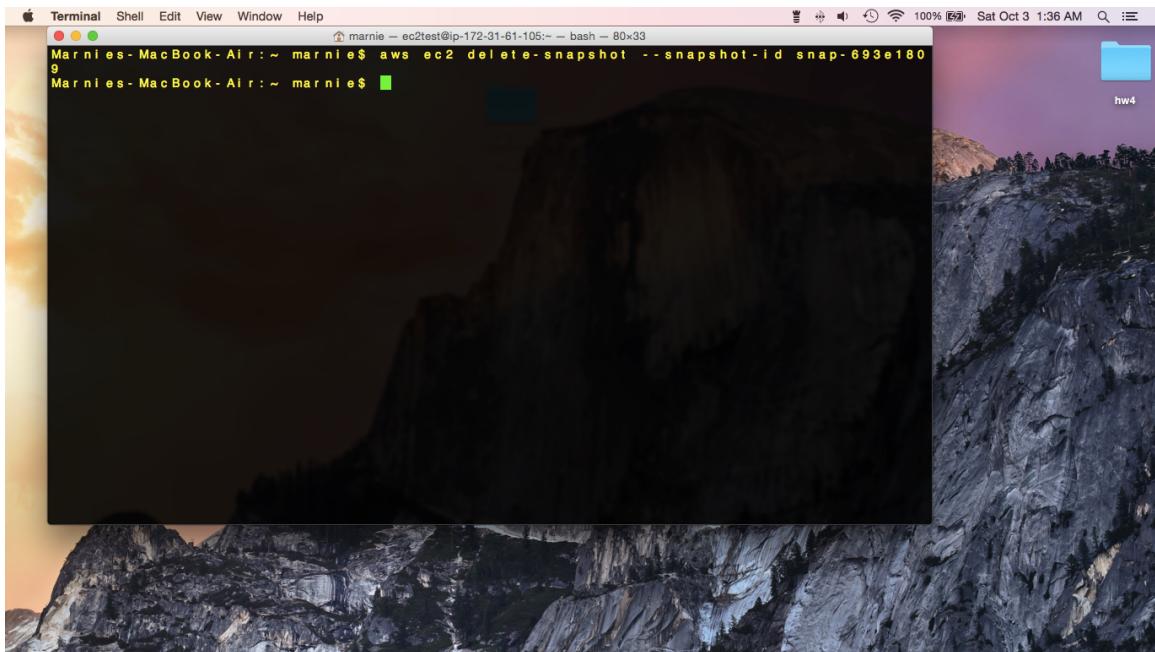


A screenshot of a Mac OS X desktop. In the foreground, a terminal window is open with the following commands and outputs:

```
marnie$ aws ec2 detach-volume --volume-id vol-fa41f51a
[Terminal output redacted]
marnie$ aws ec2 delete-volume --volume-id vol-fa41f51a
marnie$ aws ec2 detach-volume --volume-id vol-0247f3e2
[Terminal output redacted]
marnie$ aws ec2 delete-volume --volume-id vol-0247f3e2
marnie$
```

The desktop background is a scenic view of a mountain range at sunset. A file browser window titled "hw4" is visible in the top right corner, containing a folder icon.

Delete Snapshot



[Total Points: 10]