

Part A – Create an EBS Volume and mount it to the EC2 Instance

Create an EC2 Volume from AWS CL

aws ec2 create-volume --size 10 --region us-west-2 --availability-zone us-west-2c --volume-type gp2

Created a 10GB SSD in Region: us-west-2 and AvailabilityZone: us-west-2c

```
eric ~ -bash - 129x5
Last login: Mon Feb 15 22:40:03 on ttys000
Tatsumoto:~ eric$ aws ec2 create-volume --size 10 --region us-west-2 --availability-zone us-west-2c --volume-type gp2
us-west-2c      2016-02-16T04:17:13.637Z      False  30      10      creating      vol-7ea54bc7      gp2
Tatsumoto:~ eric$
```

Attach EC2 Volume from AWS CL to Instance

First get EC2 InstanceID and the EBS VolumeID

aws ec2 describe-instances (Instance ID: i-0c8cfed6)

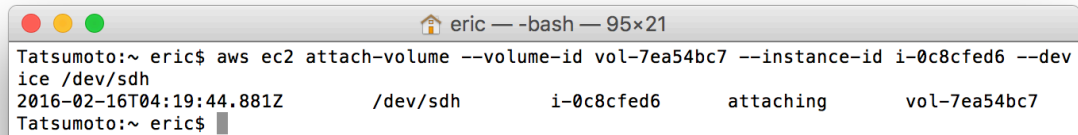
```
eric ~ -bash - 106x60
+-----+-----+
| Description | 0a:54:2f:ee:42:97 |
| MacAddress  | eni-82b8bed8      |
| NetworkInterfaceId | 418674720115 |
| OwnerId     | ip-172-31-6-93.us-west-2.compute.internal |
| PrivateDnsName | 172.31.6.93 |
| SourceDestCheck | True |
| Status      | in-use |
| SubnetId    | subnet-00588159 |
| VpcId       | vpc-f5bf3390 |
+-----+-----+
| Association |
+-----+-----+
| IpOwnerId   | amazon |
| PublicDnsName | ec2-52-36-93-56.us-west-2.compute.amazonaws.com |
| PublicIp    | 52.36.93.56 |
+-----+-----+
| Attachment |
+-----+-----+
| AttachTime  | 2016-02-14T03:05:42.000Z |
| AttachmentId | eni-attach-6460456b |
| DeleteOnTermination | True |
| DeviceIndex | 0 |
| Status      | attached |
+-----+-----+
| Groups |
+-----+-----+
| GroupId     | sg-09875e6e |
| GroupName   | AWS-SecGroup |
+-----+-----+
| PrivateIPAddresses |
+-----+-----+
| Primary     | True |
| PrivateDnsName | ip-172-31-6-93.us-west-2.compute.internal |
| PrivateIpAddress | 172.31.6.93 |
+-----+-----+
```

aws ec2 describe-volumes (Volume ID: vol-7ea54bc7)

```
eric ~ -bash - 138x22
Tatsumoto:~ eric$ aws ec2 describe-volumes
VOLUMES us-west-2c      2016-02-14T03:05:46.214Z      False  24      8      snap-ad8e61f8      in-use      vol-2642a19f      gp2
ATTACHMENTS 2016-02-14T03:05:46.000Z      True   /dev/xvda      i-0c8cfed6      attached      vol-2642a19f
VOLUMES us-west-2c      2016-02-16T04:17:13.637Z      False  30      10      available      vol-7ea54bc7      gp2
VOLUMES us-west-2a      2016-02-16T03:49:51.193Z      False  30      10      available      vol-ab09fe5d      gp2
Tatsumoto:~ eric$
```

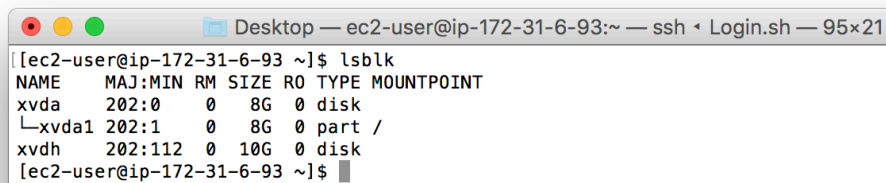
Attach EC2 Volume from AWS CL to Instance

aws ec2 attach-volume --volume-id vol-7ea54bc7 --instance-id i-0c8cfed6 --device /dev/sdh



```
eric — -bash — 95x21
Tatsumoto:~ eric$ aws ec2 attach-volume --volume-id vol-7ea54bc7 --instance-id i-0c8cfed6 --device /dev/sdh
2016-02-16T04:19:44.881Z          /dev/sdh          i-0c8cfed6          attaching          vol-7ea54bc7
Tatsumoto:~ eric$
```

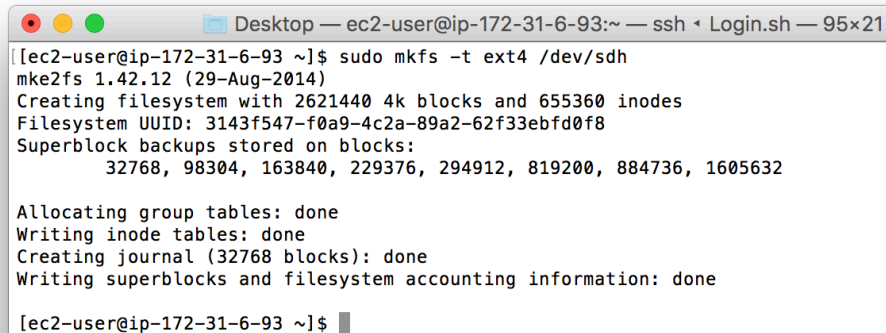
Volume Shows Available has to be Mounted:



```
Desktop — ec2-user@ip-172-31-6-93:~ — ssh • Login.sh — 95x21
[ec2-user@ip-172-31-6-93 ~]$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda         202:0    0   8G  0 disk 
└─xvda1      202:1    0   8G  0 part /
xvdh         202:112  0  10G  0 disk 
[ec2-user@ip-172-31-6-93 ~]$
```

Create a file system on the Volume:

sudo mkfs -t ext4 /dev/sdh



```
Desktop — ec2-user@ip-172-31-6-93:~ — ssh • Login.sh — 95x21
[ec2-user@ip-172-31-6-93 ~]$ sudo mkfs -t ext4 /dev/sdh
mke2fs 1.42.12 (29-Aug-2014)
Creating filesystem with 2621440 4k blocks and 655360 inodes
Filesystem UUID: 3143f547-f0a9-4c2a-89a2-62f33ebfd0f8
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

[ec2-user@ip-172-31-6-93 ~]$
```

Make a Directory/Mount-Point for the Volume:



```
Desktop — ec2-user@ip-172-31-6-93:~ — ssh • Login.sh — 95x21
[ec2-user@ip-172-31-6-93 ~]$ sudo mkdir /SSD_Drive
```

Mount the Volume to the MountPoint:

sudo mount /dev/sdh /SSD_Drive

```
Desktop — ec2-user@ip-172-31-6-93:~ — ssh • Login.sh — 95×21
[ec2-user@ip-172-31-6-93 ~]$ sudo mount /dev/sdh /SSD_Drive
[ec2-user@ip-172-31-6-93 ~]$
```

Test Copying a File (CHANGES.txt) to the Mounted Drive

```
Desktop — ec2-user@ip-172-31-6-93:/SSD_Drive — ssh • Login.sh — 95×21
[[ec2-user@ip-172-31-6-93 spark-1.6.0-bin-hadoop2.6]$ sudo cp CHANGES.txt /SSD_Drive
[[ec2-user@ip-172-31-6-93 spark-1.6.0-bin-hadoop2.6]$ cd ..
[[ec2-user@ip-172-31-6-93 ~]$ cd ..
[[ec2-user@ip-172-31-6-93 home]$ cd ..
[[ec2-user@ip-172-31-6-93 /]$ cd ..
[[ec2-user@ip-172-31-6-93 /]$ cd ./SSD_Drive/
[[ec2-user@ip-172-31-6-93 SSD_Drive]$ ls
CHANGES.txt  lost+found
[ec2-user@ip-172-31-6-93 SSD_Drive]$
```

Drive is Mounted Properly

```
Desktop — ec2-user@ip-172-31-6-93:/SSD_Drive — ssh • Login.sh — 95×21
[[ec2-user@ip-172-31-6-93 SSD_Drive]$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda        202:0    0   8G  0 disk
└─xvda1     202:1    0   8G  0 part /
xvdh        202:112  0  10G  0 disk /SSD_Drive
[ec2-user@ip-172-31-6-93 SSD_Drive]$
```

Part B – Create an AMI Image from a Running Instance (CLI)

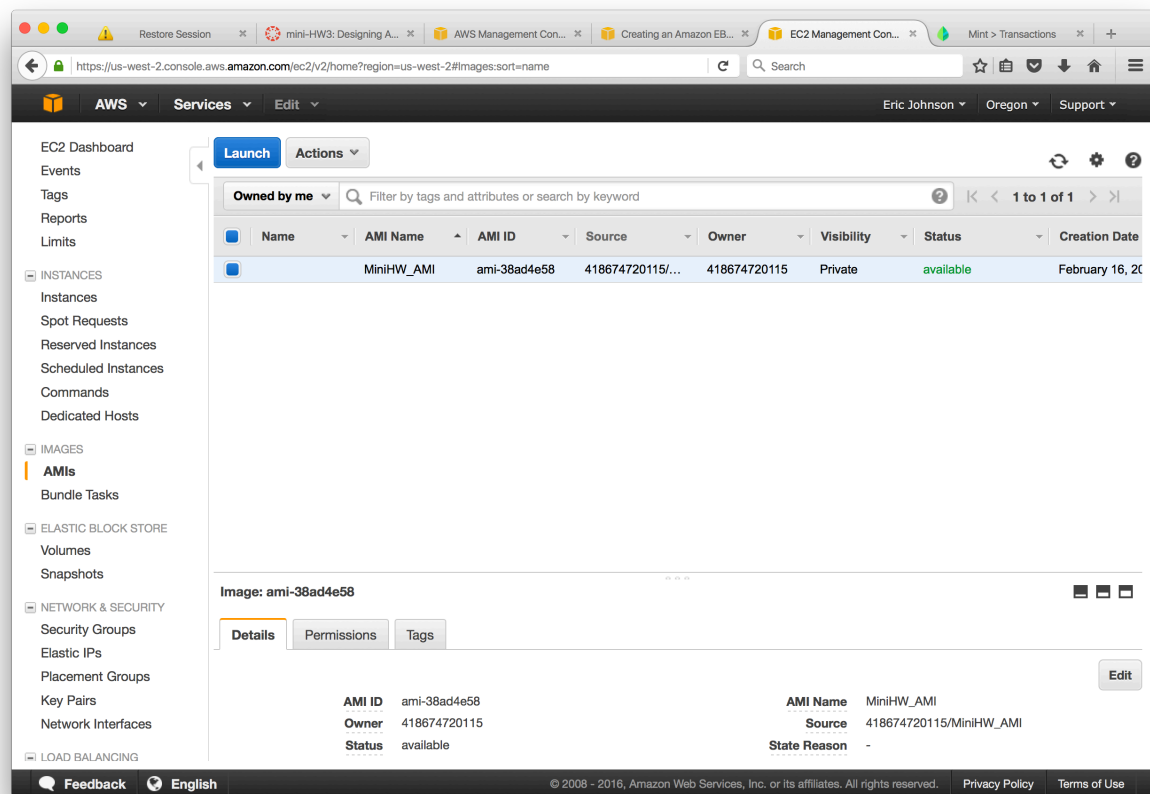
Create an EC2 Volume from AWS CL

```
aws ec2 create-image --instance-id i-0c8cfed6 --name MiniHW_AMI --description "Copy of Linux box w/ Mounted SSD"
```



```
Desktop — ec2-user@ip-172-31-6-93:/SSD_Drive — -bash — 91x30
Tatsumoto:Desktop eric$ aws ec2 create-image --instance-id i-0c8cfed6 --name MiniHW_AMI --description "Copy of Linux box w/ Mounted SSD"
ami-38ad4e58
Tatsumoto:Desktop eric$
```

AMI ID: ami-38ad4e58



The screenshot shows the AWS Management Console interface. The left sidebar contains navigation links for EC2 Dashboard, INSTANCES, IMAGES, ELASTIC BLOCK STORE, NETWORK & SECURITY, and LOAD BALANCING. The 'IMAGES' section is expanded, showing a list of AMIs. A table lists the AMI 'MiniHW_AMI' with ID 'ami-38ad4e58', owned by '418674720115', with a status of 'available'. Below the table, the 'Details' tab for 'Image: ami-38ad4e58' is selected, showing fields for AMI ID, Owner, AMI Name, Source, and State Reason.

Name	AMI Name	AMI ID	Source	Owner	Visibility	Status	Creation Date
MiniHW_AMI	ami-38ad4e58	418674720115/...	418674720115	Private	available	February 16, 2016	

Field	Value
AMI ID	ami-38ad4e58
Owner	418674720115
Status	available
AMI Name	MiniHW_AMI
Source	418674720115/MiniHW_AMI
State Reason	-