Assignment 03

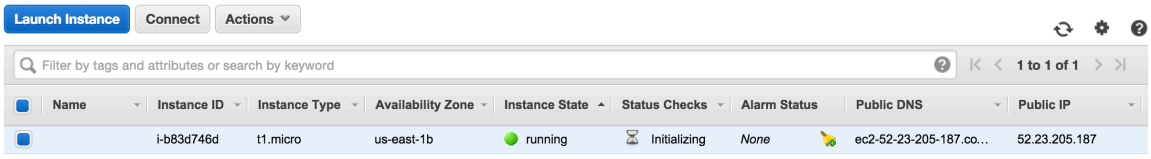
Problem 1

1. Select an EBS-backed Bitnami image. Use ami-2881c240 here. Check the image, it is paravirtual, not HVM virtualization. It has tomcat installed.


```
hqiu@bos-mpdei>> aws ec2 describe-images --image-id ami-2881c240
{
    "Images": [
        {
            "VirtualizationType": "paravirtual",
            "Name": "bitnami-tomcatstack-8.0.9-0-dev-linux-ubuntu-12.04.4-x86_64-ebs-ami-2ca16
444-3-ami-1e1e9b76",
            "Hypervisor": "xen",
            "ImageId": "ami-2881c240",
            "RootDeviceType": "ebs",
            "State": "available",
            "BlockDeviceMappings": [
                {
                    "DeviceName": "/dev/sda1",
                    "Ebs": {
                        "DeleteOnTermination": true,
                        "SnapshotId": "snap-7e33c9f2",
                        "VolumeSize": 10,
                        "VolumeType": "standard",
                        "Encrypted": false
                    }
                },
                {
                    "DeviceName": "/dev/sdb",
                    "VirtualName": "ephemeral0"
                },
                {
                    "DeviceName": "/dev/sdc",
                    "VirtualName": "ephemeral1"
                },
                {
                    "DeviceName": "/dev/sdd",
                    "VirtualName": "ephemeral2"
                },
                {
                    "DeviceName": "/dev/sde",
                    "VirtualName": "ephemeral3"
                }
            ],
            "Architecture": "x86_64",
            "ImageLocation": "979382823631/bitnami-tomcatstack-8.0.9-0-dev-linux-ubuntu-12.04.
4-x86_64-ebs-ami-2ca16444-3-ami-1e1e9b76",
```


2. Create an instance from this selected AMI. Check from the AWS Console to make sure that the instance is launched and running. After launching the instance using ‘aws ec2 run-instances’, we can use ‘aws ec2 describe-instances’ to check the health status, public IP address, public DNS name and all the other configurations of the instance.

```
hqiu@bos-mpdei>> aws ec2 run-instances --image-id ami-2881c240 --count 1 --instance-type t1.mi
cro --key-name ec2hqiu --security-group-ids launch-hqiu
{
    "OwnerId": "217134905396",
    "ReservationId": "r-112faaec",
    "Groups": [],
    "Instances": [
        {
            "Monitoring": {
                "State": "disabled"
            },
            "PublicDnsName": "",
            "KernelId": "aki-919dcaf8",
            "State": {
                "Code": 0,
                "Name": "pending"
            },
            "EbsOptimized": false,
            "LaunchTime": "2015-09-24T12:35:02.000Z",
            "PrivateIpAddress": "172.31.0.58",
            "ProductCodes": [],
            "VpcId": "vpc-dfb48aba",
            "StateTransitionReason": "",
            "InstanceId": "i-b83d746d",
            "ImageId": "ami-2881c240",
            "PrivateDnsName": "ip-172-31-0-58.ec2.internal",
            "KeyName": "ec2hqiu",
            "SecurityGroups": [
                {
                    "GroupName": "launch-hqiu",
                    "GroupId": "sg-adfff3ca"
                }
            ],
            "ClientToken": "",
            "SubnetId": "subnet-66024a11",
            "InstanceType": "t1.micro",
            "NetworkInterfaces": [
                {
                    "Status": "in-use",
                    "MacAddress": "0a:be:f9:0d:0e:d5",
                    "SourceDestCheck": true,
                    "VpcId": "vpc-dfb48aba",
                    "Description": "",
                    "NetworkInterfaceId": "eni-53cff519",
                    "PrivateIpAddresses": [
```

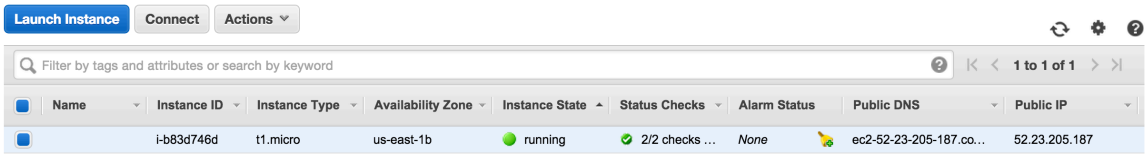


```
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-b83d746d --output table
```

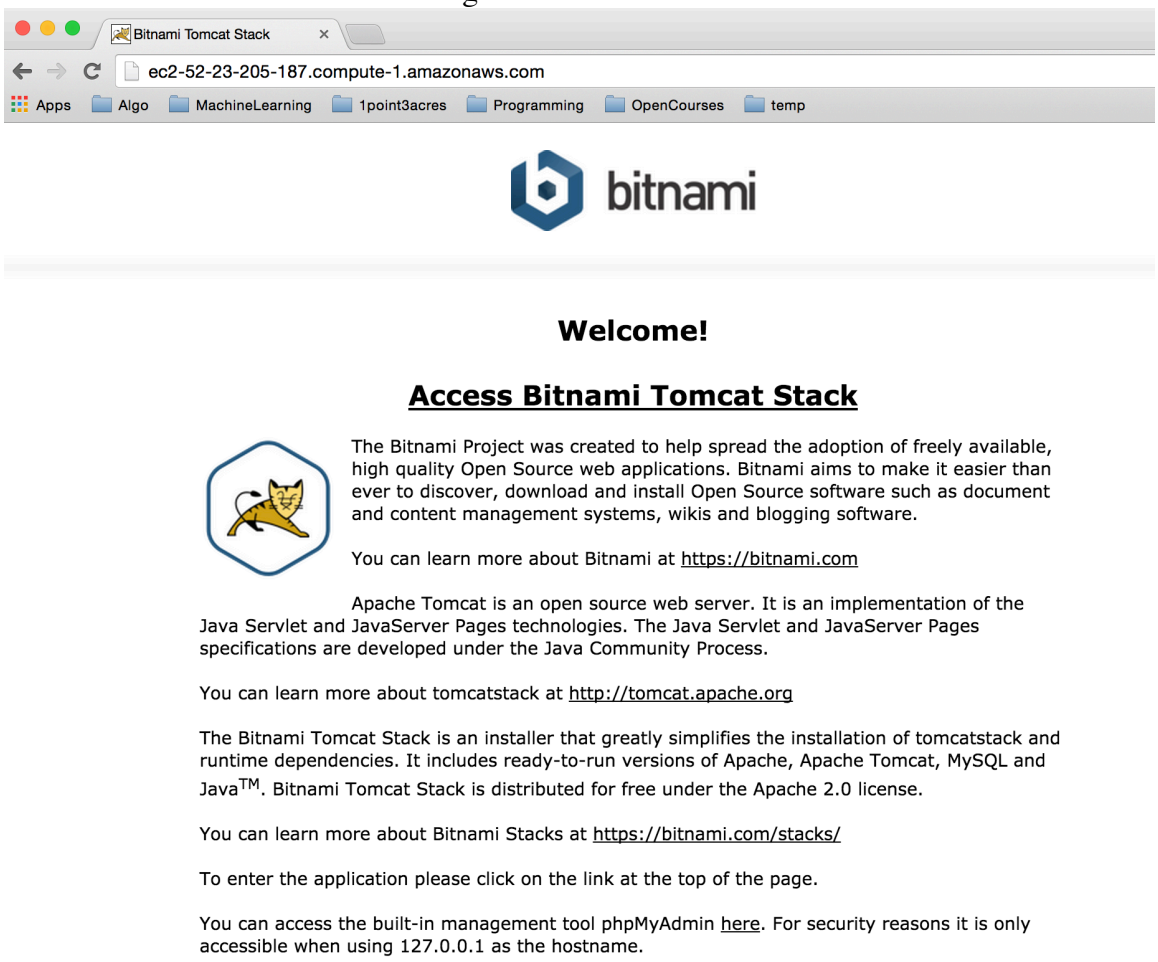
```
|                                DescribeInstances                                |
+---------------------------------------------------------------------------------+
||                                 Reservations                                  ||
|+-----------------------------------+-----------------------------------------+|
||  OwnerId                          |  217134905396                           ||
||  ReservationId                    |  r-112faaec                             ||
|+-----------------------------------+-----------------------------------------+|
|||                                 Instances                                   |||
||+-------------------------+-------------------------------------------------+||
|||  AmiLaunchIndex         |  0                                              |||
|||  Architecture           |  x86_64                                         |||
|||  ClientToken            |                                                 |||
|||  EbsOptimized           |  False                                          |||
|||  Hypervisor             |  xen                                            |||
|||  ImageId                |  ami-2881c240                                   |||
|||  InstanceId             |  i-b83d746d                                     |||
|||  InstanceType           |  t1.micro                                       |||
|||  KernelId               |  aki-919dcaf8                                   |||
|||  KeyName                |  ec2hqiu                                        |||
|||  LaunchTime             |  2015-09-24T12:35:02.000Z                       |||
|||  PrivateDnsName         |  ip-172-31-0-58.ec2.internal                    |||
|||  PrivateIpAddress       |  172.31.0.58                                    |||
|||  PublicDnsName          |  ec2-52-23-205-187.compute-1.amazonaws.com      |||
|||  PublicIpAddress        |  52.23.205.187                                  |||
|||  RootDeviceName         |  /dev/sda1                                      |||
|||  RootDeviceType         |  ebs                                            |||
|||  SourceDestCheck        |  True                                           |||
|||  StateTransitionReason  |                                                 |||
|||  SubnetId               |  subnet-66024a11                                |||
|||  VirtualizationType     |  paravirtual                                    |||
|||  VpcId                  |  vpc-dfb48aba                                   |||
||+-------------------------+-------------------------------------------------+||
|||                          BlockDeviceMappings                                |||
||+-----------------------------------+-----------------------------------------+||
|||  DeviceName                       |  /dev/sda1                              |||
||+-----------------------------------+-----------------------------------------+||
||||                                Ebs                                        ||||
|||+-------------------------------+---------------------------------------+|||
||||  AttachTime                   |  2015-09-24T12:35:06.000Z             ||||
||||  DeleteOnTermination          |  True                                 ||||
||||  Status                       |  attached                             ||||
||||  VolumeId                     |  vol-6d77b48d                         ||||
|||+-------------------------------+---------------------------------------+|||
```

```
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-b83d746d --query 'Reservations[0]
.Instances[0].PublicIpAddress'
"52.23.205.187"
```

```
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-b83d746d --query 'Reservations[0]
.Instances[0].PublicDnsName'
"ec2-52-23-205-187.compute-1.amazonaws.com"
```



3. Add a custom web page to the Tomcat server. Here is the default web page when we access the instance through the browser.



Remote log into the instance and add a custom web page.

```
hqiu@bos-mpdei>> ssh -i "ec2hqiu.pem" ubuntu@ec2-52-23-205-187.compute-1.amazonaws.com
Welcome to Ubuntu 12.04.4 LTS (GNU/Linux 3.2.0-65-virtual x86_64)
       ___ _ _                   _
      | _ |_) |_ _ _  __ _ _ __ (_)
      | _ \ |  _| ' \/ _` | '  \| |
      |___/_|\__|_|_|\__,_|_|_|_|_|

  *** Welcome to the Bitnami Tomcat 8.0.9-0         ***
  *** Bitnami Wiki:   http://wiki.bitnami.com/      ***
  *** Bitnami Forums: http://community.bitnami.com/ ***

bitnami@ip-172-31-0-58:~$ sudo mkdir /opt/bitnami/apache-tomcat/webapps/cscie90
bitnami@ip-172-31-0-58:~$ sudo cp /opt/bitnami/apache-tomcat/webapps/examples/index.html /opt/
bitnami/apache-tomcat/webapps/cscie90
bitnami@ip-172-31-0-58:~$ sudo vi /opt/bitnami/apache-tomcat/webapps/cscie90/index.html
```

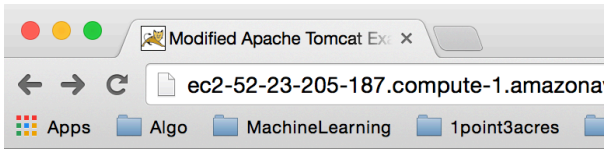
This is the modified index.xml under '/opt/bitnami/apache-tomcat/webapps/cscie90/'.

```
<!--
  Licensed to the Apache Software Foundation (ASF) under one or more
  contributor license agreements.  See the NOTICE file distributed with
  this work for additional information regarding copyright ownership.
  The ASF licenses this file to You under the Apache License, Version 2.0
  (the "License"); you may not use this file except in compliance with
  the License.  You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
-->
<!DOCTYPE HTML><html lang="en"><head>
<meta charset="UTF-8">
<title>Modified Apache Tomcat Examples by hqiu</title>
</head>
<body>
<p>
<h3>Apache Tomcat Examples Modified by hqiu</H3>
<p></p>
<ul>
<li><a href="servlets">Servlets examples</a></li>
<!-- <li><a href="jsp">JSP Examples</a></li> -->
<li><a href="websocket/index.xhtml">WebSocket Examples</a></li>
<li><a href="https://www.google.com">Google Website</a></li>
</ul>
</body></html>
~
~
```

Verify we can access the new page via browser:

http://ec2-52-23-205-187.compute-1.amazonaws.com/cscie90/

(The full path isn't showed in this image below. I will show it in the new instance later.)



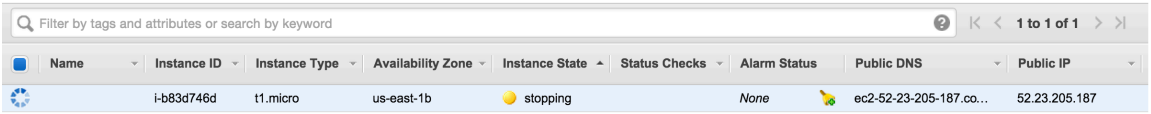
Apache Tomcat Examples Modified by hqiu

- Servlets examples
- WebSocket Examples
- Google Website

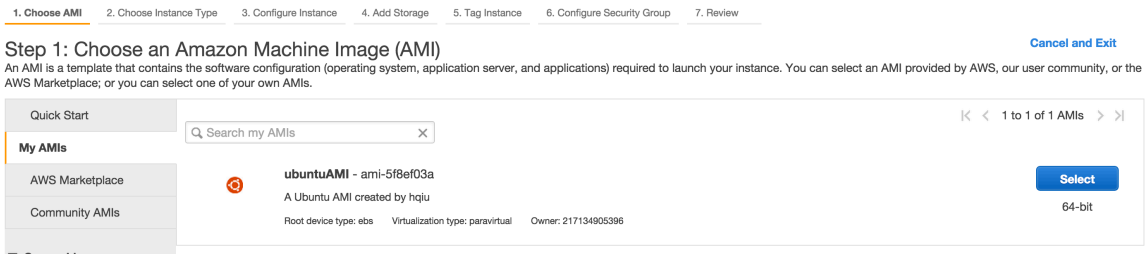
4. Stop the instance and create a new AMI from this instance. First use 'aws ec2 stop-instances' to stop the running instance. Check from the AWS Console. We can create a new image by using 'aws ec2 create-image'. Give the ID of the instance we are going to copy. Once we are done, we can check from the AWS

Console. Go to ‘EC2->Launch Instance->Choose AMI’, we can see the image we just created is in ‘My AMIs”.

```
hqiu@bos-mpdei>> aws ec2 stop-instances --instance-ids i-b83d746d
{
    "StoppingInstances": [
        {
            "InstanceId": "i-b83d746d",
            "CurrentState": {
                "Code": 64,
                "Name": "stopping"
            },
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
}
```




```
hqiu@bos-mpdei>> aws ec2 create-image --instance-id i-b83d746d --name "ubuntuAMI" --descriptio
n "A Ubuntu AMI created by hqiu"
{
    "ImageId": "ami-5f8ef03a"
}
```



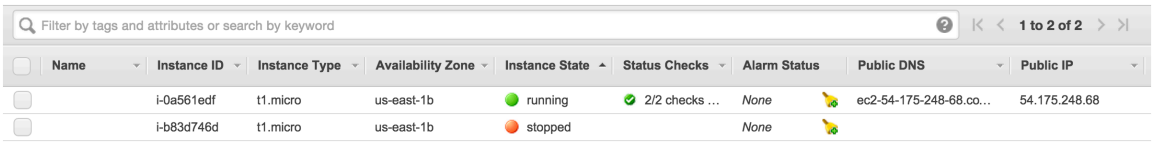
5. Check the configuration of the image we just created. We can see that it has the same virtualization type: paravirtual. The name is what we’ve given: ubuntuAMI.

```
hqiu@bos-mpdei>> aws ec2 describe-images --image-id ami-5f8ef03a
{
    "Images": [
        {
            "VirtualizationType": "paravirtual",
            "Name": "ubuntuAMI",
            "Hypervisor": "xen",
            "ImageId": "ami-5f8ef03a",
            "RootDeviceType": "ebs",
            "State": "pending",
            "BlockDeviceMappings": [
                {
                    "DeviceName": "/dev/sda1",
                    "Ebs": {
                        "DeleteOnTermination": true,
                        "SnapshotId": "snap-61ecd720",
                        "VolumeSize": 10,
                        "VolumeType": "standard",
                        "Encrypted": false
                    }
                },
```

```
hqiu@bos-mpdei>> aws ec2 describe-images --image-id ami-5f8ef03a --output table
-------------------------------------------------------------
|                      DescribeImages                       |
+-----------------------------------------------------------+
||                         Images                          ||
|+---------------------+---------------------------------+|
||  Architecture       |  x86_64                         ||
||  CreationDate       |  2015-09-24T15:44:58.000Z       ||
||  Description        |  A Ubuntu AMI created by hqiu   ||
||  Hypervisor         |  xen                            ||
||  ImageId            |  ami-5f8ef03a                   ||
||  ImageLocation      |  217134905396/ubuntuAMI         ||
||  ImageType          |  machine                        ||
||  KernelId           |  aki-919dcaf8                   ||
||  Name               |  ubuntuAMI                      ||
||  OwnerId            |  217134905396                   ||
||  Public             |  False                          ||
||  RootDeviceName     |  /dev/sda1                      ||
||  RootDeviceType     |  ebs                            ||
||  State              |  available                      ||
||  VirtualizationType |  paravirtual                    ||
|+---------------------+---------------------------------+|
```

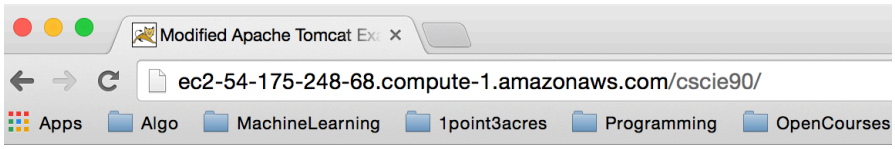
Create an instance from this new AMI.

```
hqiu@bos-mpdei>> aws ec2 run-instances --image-id ami-5f8ef03a --count 1 --instance-type t1.mi
cro --key-name ec2hqiu --security-group-ids launch-hqiu
{
    "OwnerId": "217134905396",
    "ReservationId": "r-9c67e361",
    "Groups": [],
    "Instances": [
        {
            "Monitoring": {
                "State": "disabled"
            },
            "PublicDnsName": "",
            "KernelId": "aki-919dcaf8",
            "State": {
                "Code": 0,
                "Name": "pending"
            },
            "EbsOptimized": false,
            "LaunchTime": "2015-09-24T15:48:21.000Z",
            "PrivateIpAddress": "172.31.10.17",
            "ProductCodes": [],
            "VpcId": "vpc-dfb48aba",
            "StateTransitionReason": "",
            "InstanceId": "i-0a561edf",
            "ImageId": "ami-5f8ef03a",
            "PrivateDnsName": "ip-172-31-10-17.ec2.internal",
            "KeyName": "ec2hqiu",
            "SecurityGroups": [
                {
                    "GroupName": "launch-hqiu",
                    "GroupId": "sg-adfff3ca"
                }
            ],
```



```
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-0a561edf --output table
-------------------------------------------------------------------------------
|                           DescribeInstances                                 |
+-----------------------------------------------------------------------------+
||                             Reservations                                  ||
|+-------------------------------+-------------------------------------------+|
||  OwnerId                      |  217134905396                             ||
||  ReservationId                |  r-9c67e361                               ||
|+-------------------------------+-------------------------------------------+|
|||                              Instances                                  |||
||+-------------------------+-----------------------------------------------+||
|||  AmiLaunchIndex         |  0                                            |||
|||  Architecture           |  x86_64                                       |||
|||  ClientToken            |                                               |||
|||  EbsOptimized           |  False                                        |||
|||  Hypervisor             |  xen                                          |||
|||  ImageId                |  ami-5f8ef03a                                 |||
|||  InstanceId             |  i-0a561edf                                   |||
|||  InstanceType           |  t1.micro                                     |||
|||  KernelId               |  aki-919dcaf8                                 |||
|||  KeyName                |  ec2hqiu                                      |||
|||  LaunchTime             |  2015-09-24T15:48:21.000Z                     |||
|||  PrivateDnsName         |  ip-172-31-10-17.ec2.internal                 |||
|||  PrivateIpAddress       |  172.31.10.17                                 |||
|||  PublicDnsName          |  ec2-54-175-248-68.compute-1.amazonaws.com    |||
|||  PublicIpAddress        |  54.175.248.68                                |||
|||  RootDeviceName         |  /dev/sda1                                    |||
|||  RootDeviceType         |  ebs                                          |||
|||  SourceDestCheck        |  True                                         |||
|||  StateTransitionReason  |                                               |||
|||  SubnetId               |  subnet-66024a11                              |||
|||  VirtualizationType     |  paravirtual                                  |||
|||  VpcId                  |  vpc-dfb48aba                                 |||
||+-------------------------+-----------------------------------------------+||
```

6. Verify we can see the new web page on the new instance: http://ec2-54-175-248-68.compute-1.amazonaws.com/cscie90/



Apache Tomcat Examples Modified by hqiu

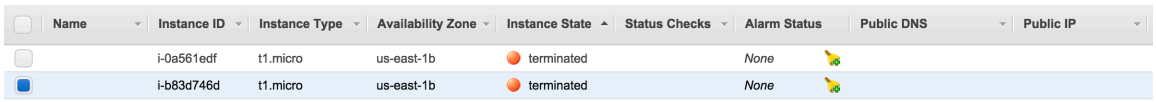
- Servlets examples
- WebSocket Examples
- Google Website

We can also check if the custom web page exists on the new instance:

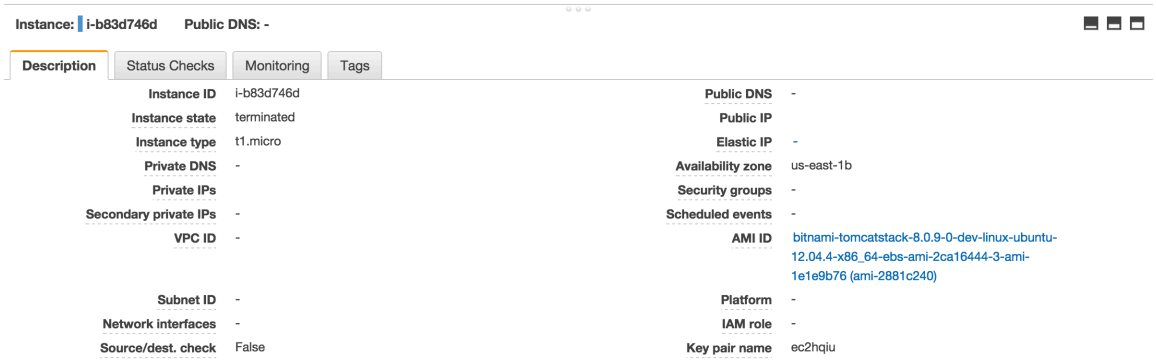
```
hqiu@bos-mpdei>> ssh -i "ec2hqiu.pem" ubuntu@ec2-54-175-248-68.compute-1.amazonaws.com
Welcome to Ubuntu 12.04.4 LTS (GNU/Linux 3.2.0-65-virtual x86_64)
      ___ _ _                   _
     | _ |_) |_ _ _  __ _ _ __ (_)
     | _ \ |  _| ' \/ _` | '  \| |
     |___/_|\__|_|_|\__,_|_|_|_|_|

  *** Welcome to the Bitnami Tomcat 8.0.9-0         ***
  *** Bitnami Wiki:   http://wiki.bitnami.com/      ***
  *** Bitnami Forums: http://community.bitnami.com/ ***
Last login: Thu Sep 24 15:51:42 2015 from 72.246.0.14
bitnami@ip-172-31-10-17:~$ ls /opt/bitnami/apache-tomcat/webapps/cscie90/index.html
/opt/bitnami/apache-tomcat/webapps/cscie90/index.html
```

7. Terminate all the instances.

```
hqiu@bos-mpdei>> aws ec2 terminate-instances --instance-ids i-0a561edf
{
    "TerminatingInstances": [
        {
            "InstanceId": "i-0a561edf",
            "CurrentState": {
                "Code": 32,
                "Name": "shutting-down"
            },
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
}
hqiu@bos-mpdei>> aws ec2 terminate-instances --instance-ids i-b83d746d
{
    "TerminatingInstances": [
        {
            "InstanceId": "i-b83d746d",
            "CurrentState": {
                "Code": 48,
                "Name": "terminated"
            },
            "PreviousState": {
                "Code": 80,
                "Name": "stopped"
            }
        }
    ]
}
```



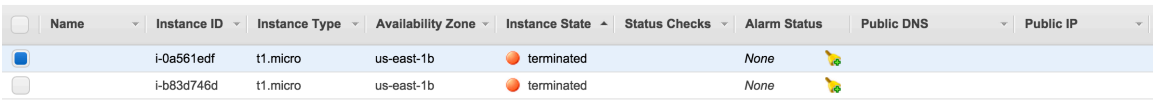
Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS	Public IP
	i-0a561edf	t1.micro	us-east-1b	terminated	None			
	i-b83d746d	t1.micro	us-east-1b	terminated	None			



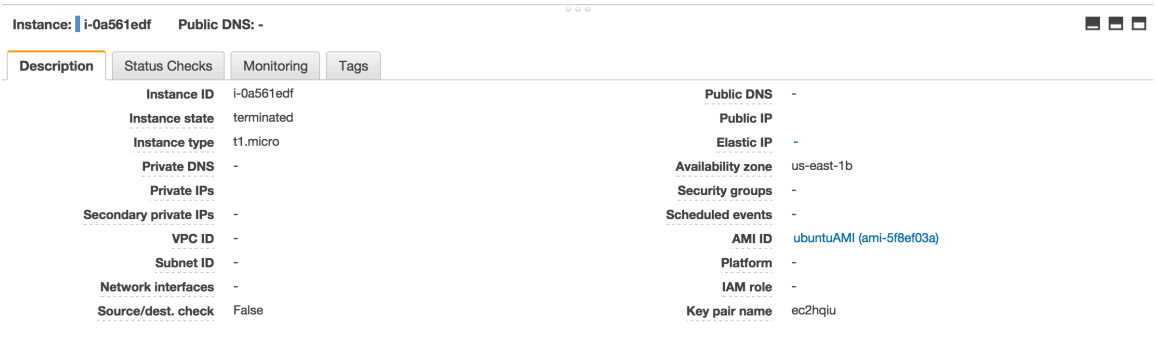
Instance: i-b83d746d Public DNS: -

Description | Status Checks | Monitoring | Tags

Instance ID	i-b83d746d	Public DNS	-
Instance state	terminated	Public IP	
Instance type	t1.micro	Elastic IP	-
Private DNS	-	Availability zone	us-east-1b
Private IPs		Security groups	-
Secondary private IPs	-	Scheduled events	-
VPC ID	-	AMI ID	bitnami-tomcatstack-8.0.9-0-dev-linux-ubuntu-12.04.4-x86_64-ebs-ami-2ca16444-3-ami-1e1e9b76 (ami-2881c240)
Subnet ID	-	Platform	-
Network interfaces	-	IAM role	-
Source/dest. check	False	Key pair name	ec2hqiu



Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS	Public IP
	i-0a561edf	t1.micro	us-east-1b	terminated	None			
	i-b83d746d	t1.micro	us-east-1b	terminated	None			



Instance: i-0a561edf Public DNS: -

Description | Status Checks | Monitoring | Tags

Instance ID	i-0a561edf	Public DNS	-
Instance state	terminated	Public IP	
Instance type	t1.micro	Elastic IP	-
Private DNS	-	Availability zone	us-east-1b
Private IPs		Security groups	-
Secondary private IPs	-	Scheduled events	-
VPC ID	-	AMI ID	ubuntuAMI (ami-5f8ef03a)
Subnet ID	-	Platform	-
Network interfaces	-	IAM role	-
Source/dest. check	False	Key pair name	ec2hqiu

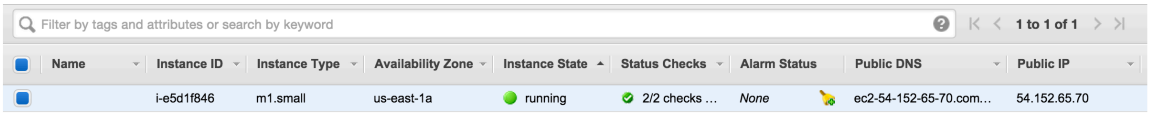
Problem 2

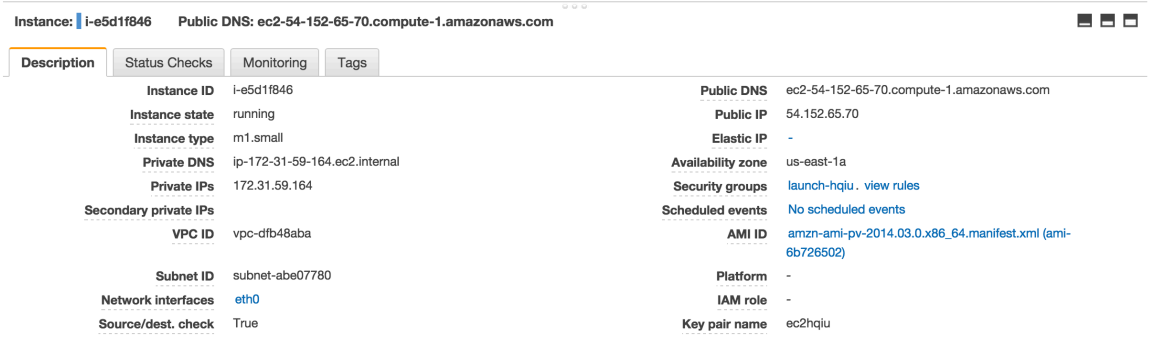
1. Select a small Amazon owned instance-store Linux AMI (ami-6b726502). The AMI has the paravirtual and not HVM virtualization type. Create an instance from this AMI. Run ‘sudo yum update’ once it’s launched.

```
hqiu@bos-mpdei>> aws ec2 describe-images --image-id ami-6b726502
{
    "Images": [
        {
            "VirtualizationType": "paravirtual",
            "Name": "amzn-ami-pv-2014.03.0.x86_64-s3",
            "Hypervisor": "xen",
            "ImageOwnerAlias": "amazon",
            "ImageId": "ami-6b726502",
            "RootDeviceType": "instance-store",
            "State": "available",
            "BlockDeviceMappings": [],
            "Architecture": "x86_64",
            "ImageLocation": "amzn-ami-us-east-1/amzn-ami-pv-2014.03.0.x86_64.manifest.xml",
            "KernelId": "aki-919dcaf8",
            "OwnerId": "137112412989",
            "CreationDate": "2014-03-25T06:35:32.000Z",
            "Public": true,
            "ImageType": "machine",
            "Description": "Amazon Linux AMI x86_64 PV S3"
        }
    ]
}
hqiu@bos-mpdei>> aws ec2 run-instances --image-id ami-6b726502 --count 1 --instance-type m1.sm
all --key-name ec2hqiu --security-group-ids launch-hqiu
{
    "OwnerId": "217134905396",
    "ReservationId": "r-e8ca663e",
    "Groups": [],
    "Instances": [
        {
            "Monitoring": {
                "State": "disabled"
            },
            "PublicDnsName": "",
            "KernelId": "aki-919dcaf8",
            "State": {
                "Code": 0,
                "Name": "pending"
            },
            "EbsOptimized": false,
            "LaunchTime": "2015-09-25T03:27:41.000Z",
            "PrivateIpAddress": "172.31.59.164",
            "ProductCodes": [],
            "VpcId": "vpc-dfb48aba",
            "StateTransitionReason": "",
            "InstanceId": "i-e5d1f846",
            "ImageId": "ami-6b726502",
            "PrivateDnsName": "ip-172-31-59-164.ec2.internal",
            "KeyName": "ec2hqiu",
            "SecurityGroups": [
                {
                    "GroupName": "launch-hqiu",
                    "GroupId": "sg-adfff3ca"
                }
            ],
```

```
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-e5d1f846 --output table
-------------------------------------------------------------------------------
|                           DescribeInstances                                 |
+-----------------------------------------------------------------------------+
||                            Reservations                                   ||
|+-----------------------------------+-------------------------------------+|
||  OwnerId                          |  217134905396                       ||
||  ReservationId                    |  r-e8ca663e                         ||
|+-----------------------------------+-------------------------------------+|
|||                             Instances                                   |||
||+-------------------------+-----------------------------------------------+||
|||  AmiLaunchIndex         |  0                                            |||
|||  Architecture           |  x86_64                                       |||
|||  ClientToken            |                                               |||
|||  EbsOptimized           |  False                                        |||
|||  Hypervisor             |  xen                                          |||
|||  ImageId                |  ami-6b726502                                 |||
|||  InstanceId             |  i-e5d1f846                                   |||
|||  InstanceType           |  m1.small                                     |||
|||  KernelId               |  aki-919dcaf8                                 |||
|||  KeyName                |  ec2hqiu                                      |||
|||  LaunchTime             |  2015-09-25T03:27:41.000Z                     |||
|||  PrivateDnsName         |  ip-172-31-59-164.ec2.internal                |||
|||  PrivateIpAddress       |  172.31.59.164                                |||
|||  PublicDnsName          |  ec2-54-152-65-70.compute-1.amazonaws.com     |||
|||  PublicIpAddress        |  54.152.65.70                                 |||
|||  RootDeviceType         |  instance-store                               |||
|||  SourceDestCheck        |  True                                         |||
|||  StateTransitionReason  |                                               |||
|||  SubnetId               |  subnet-abe07780                              |||
|||  VirtualizationType     |  paravirtual                                  |||
|||  VpcId                  |  vpc-dfb48aba                                 |||
||+-------------------------+-----------------------------------------------+||
```

```
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-e5d1f846 --query 'Reservations[0]
.Instances[0].PublicIpAddress'
"54.152.65.70"
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-e5d1f846 --query 'Reservations[0]
.Instances[0].PublicDnsName'
"ec2-54-152-65-70.compute-1.amazonaws.com"
```





Do ‘sudo yum update’ on the instance to update all the software.

```
hqiu@bos-mpdei>> ssh -i "ec2hqiu.pem" ec2-user@ec2-54-152-65-70.compute-1.amazonaws.com
The authenticity of host 'ec2-54-152-65-70.compute-1.amazonaws.com (54.152.65.70)' can't be es
tablished.
RSA key fingerprint is 98:13:c2:47:67:0c:0a:6b:aa:9b:cd:02:30:40:f8:21.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-54-152-65-70.compute-1.amazonaws.com,54.152.65.70' (RSA) to th
e list of known hosts.

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

https://aws.amazon.com/amazon-linux-ami/2014.03-release-notes/
39 package(s) needed for security, out of 204 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2015.09 is available.
[ec2-user@ip-172-31-59-164 ~]$ sudo yum update
Loaded plugins: priorities, update-motd, upgrade-helper
Resolving Dependencies
--> Running transaction check
---> Package PyYAML.x86_64 0:3.10-3.6.amzn1 will be obsoleted
---> Package acl.x86_64 0:2.2.49-6.9.amzn1 will be updated
---> Package acl.x86_64 0:2.2.49-6.11.amzn1 will be an update
---> Package at.x86_64 0:3.1.10-43.8.amzn1 will be updated
---> Package at.x86_64 0:3.1.10-44.13.amzn1 will be an update
---> Package attr.x86_64 0:2.4.44-7.9.amzn1 will be updated


Replaced:
  PyYAML.x86_64 0:3.10-3.6.amzn1                cloud-disk-utils.x86_64 0:0.27-1.3.amzn1
  m2crypto.x86_64 0:0.20.2-9.10.amzn1           man.x86_64 0:1.6f-32.11.amzn1
  python-argparse.noarch 0:1.2.1-2.2.amzn1      python-boto.noarch 0:2.27.0-1.0.amzn1
  python-botocore.noarch 0:0.36.0-1.0.amzn1     python-chardet.noarch 0:2.0.1-1.2.amzn1
  python-cheetah.x86_64 0:2.4.1-1.7.amzn1       python-colorama.noarch 0:0.2.5-1.4.amzn1
  python-configobj.noarch 0:4.6.0-3.9.amzn1     python-crypto.x86_64 0:2.6.1-1.7.amzn1
  python-daemon.noarch 0:1.5.2-1.2.amzn1        python-dateutil.noarch 0:2.1-1.0.amzn1
  python-docutils.noarch 0:0.11-1.11.amzn1      python-imaging.x86_64 0:1.1.6-19.6.amzn1
  python-jmespath.noarch 0:0.3.1-1.0.amzn1      python-kitchen.noarch 0:1.1.1-4.3.amzn1
  python-lockfile.noarch 0:0.8-3.2.amzn1        python-markdown.noarch 0:2.0.1-3.1.6.amzn1
  python-ordereddict.noarch 0:1.1-2.2.amzn1     python-paramiko.noarch 0:1.7.5-2.1.4.amzn1
  python-ply.noarch 0:3.4-3.3.amzn1             python-pygments.noarch 0:1.4-4.8.amzn1
  python-requests.noarch 0:1.2.3-5.7.amzn1      python-rsa.noarch 0:3.1.2-4.3.amzn1
  python-simplejson.x86_64 0:3.3.0-1.5.amzn1    python-six.noarch 0:1.2.0-1.3.amzn1
  python-urllib3.noarch 0:1.7-4.6.amzn1

Complete!
```

2. Verify the tools to create the bundle are there.

```
[ec2-user@ip-172-31-59-164 ~]$ which ec2-bundle-vol
/opt/aws/bin/ec2-bundle-vol
[ec2-user@ip-172-31-59-164 ~]$ ec2-ami-tools-version
1.5.7 20071010

Copyright 2008-2014 Amazon.com, Inc. or its affiliates.  All Rights Reserved.
Licensed under the Amazon Software License (the "License").  You may not use
this file except in compliance with the License. A copy of the License is
located at http://aws.amazon.com/asl or in the "license" file accompanying this
file.  This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied. See the License for the
specific language governing permissions and limitations under the License.
```

3. Make the directory '/mnt' to be writable. Copy the private key and certificate files to the /mnt directory of the remote machine. These keys label this instance and the future AMI as my own.

```
[ec2-user@ip-172-31-59-164 ~]$ ls -lart /mnt
total 8
drwxr-xr-x  2 root root 4096 Jan  6  2012 .
dr-xr-xr-x 25 root root 4096 Sep 25 03:29 ..
[ec2-user@ip-172-31-59-164 ~]$ sudo chmod 777 /mnt
[ec2-user@ip-172-31-59-164 ~]$ ls -lart /mnt
total 8
drwxrwxrwx  2 root root 4096 Jan  6  2012 .
dr-xr-xr-x 25 root root 4096 Sep 25 03:29 ..
```

Modify something on my instance at the same time. So I can prove that my modification got carried over to the new AMI and instance later. I added a file 'myNewFile.txt' to 'ec2-user' home directory.

```
hqiu@bos-mpdei>> scp -i ec2hqiu.pem pk-*.pem cert-*.pem ec2-user@ec2-54-152-65-70.compute-1.am
azonaws.com:/mnt
pk-53WR3UN6I2U4EPMCGV7GO6APAOXKERJP.pem                      100% 1736     1.7KB/s   00:00
cert-53WR3UN6I2U4EPMCGV7GO6APAOXKERJP.pem                    100% 1302     1.3KB/s   00:00
hqiu@bos-mpdei>> scp -i ec2hqiu.pem ../Homework/HW3/myNewFile.txt ec2-user@ec2-54-152-65-70.co
mpute-1.amazonaws.com:~/
myNewFile.txt                                                100%   76     0.1KB/s   00:00
```

Check the new added file is on the instance and show its content.

```
hqiu@bos-mpdei>> ssh -i "ec2hqiu.pem" ec2-user@ec2-54-152-65-70.compute-1.amazonaws.com
Last login: Fri Sep 25 03:44:04 2015 from c-71-233-46-151.hsd1.ma.comcast.net

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

https://aws.amazon.com/amazon-linux-ami/2015.09-release-notes/
[ec2-user@ip-172-31-59-164 ~]$ ls -lart ~/myNewFile.txt
-rw-r--r-- 1 ec2-user ec2-user 76 Sep 25 03:45 /home/ec2-user/myNewFile.txt
[ec2-user@ip-172-31-59-164 ~]$ cat ~/myNewFile.txt
This is the file that will be carried over to the new AMI.
Created by hqiu.
```

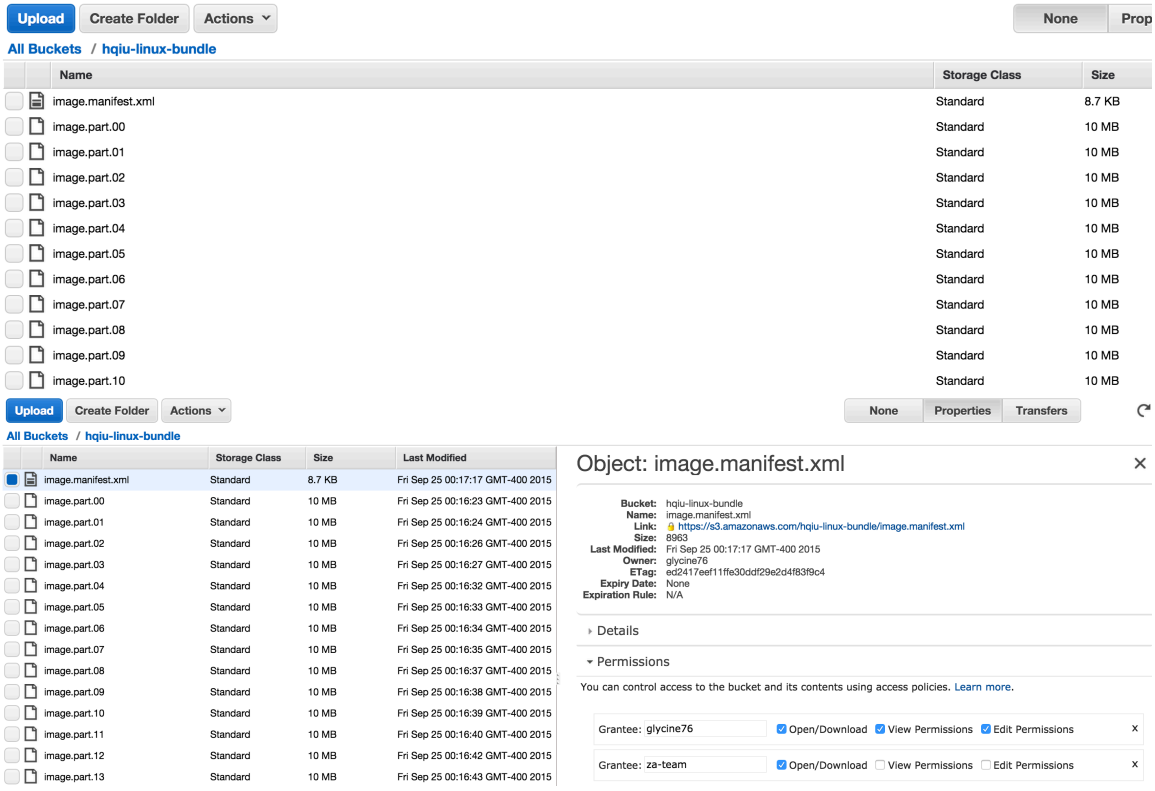
4. Bundle my instance up and create my own S3 backed replica image. Connect to my instance through my laptop terminal. Change myself to 'root' in the directory '/mnt' of my remote instance. Do 'ec2-bundle-vol'. Give your private key and certificate. The last argument is my AWS account ID. Specify the architecture 'x86_64'. 'ec2-bundle-vol' has bundled all files of my root file system except the excluded directories. Check the content of the '/tmp/image.manifest.xml'.

```
[ec2-user@ip-172-31-59-164 ~]$ cd /mnt
[ec2-user@ip-172-31-59-164 mnt]$ sudo -E su
[root@ip-172-31-59-164 mnt]# ls
cert-53WR3UN6I2U4EPMCGV7GO6APAOXKERJP.pem  pk-53WR3UN6I2U4EPMCGV7GO6APAOXKERJP.pem
[root@ip-172-31-59-164 mnt]# ec2-bundle-vol -k pk-53WR3UN6I2U4EPMCGV7GO6APAOXKERJP.pem -c cert
-53WR3UN6I2U4EPMCGV7GO6APAOXKERJP.pem -u 217134905396
Please specify a value for arch [x86_64]: x86_64
Setting partition type to bundle "/" with...
Auto-detecting partition type for "/"
Partition label detected using parted: "loop"
Using partition type "none"
Copying / into the image file /tmp/image...
Excluding:
         /proc
         /sys
         /dev
         /dev/pts
         /proc/sys/fs/binfmt_misc
         /dev
         /media
         /mnt
         /proc
         /sys
         /tmp/image
         /mnt/img-mnt

Image file created: /tmp/image
Volume cloning done.
Bundling image file...
Splitting /tmp/image.tar.gz.enc...
Created image.part.00
Created image.part.01
Created image.part.02
Created image.part.03
Created image.part.04
Created image.part.05
Created image.part.06
Created image.part.07
Created image.part.08
Created image.part.09
Created image.part.10
Created image.part.11

Created image.part.32
Created image.part.33
Created image.part.34
Created image.part.35
Created image.part.36
Created image.part.37
Created image.part.38
Created image.part.39
Created image.part.40
Created image.part.41
Created image.part.42
Created image.part.43
Generating digests for each part...
Digests generated.
Unable to read instance meta-data for ancestor-ami-ids
Unable to read instance meta-data for ramdisk-id
Unable to read instance meta-data for product-codes
Creating bundle manifest...
Bundle manifest is /tmp/image.manifest.xml
ec2-bundle-vol complete.
```

5. Check the content of ‘image.manifest.xml’. Use ‘ec2-upload-bundle’ to upload the bundled AMI to an Amazon S3 bucket before it could be accessed by EC2. Use my access key and secret access key as the command line arguments. Specify the region.

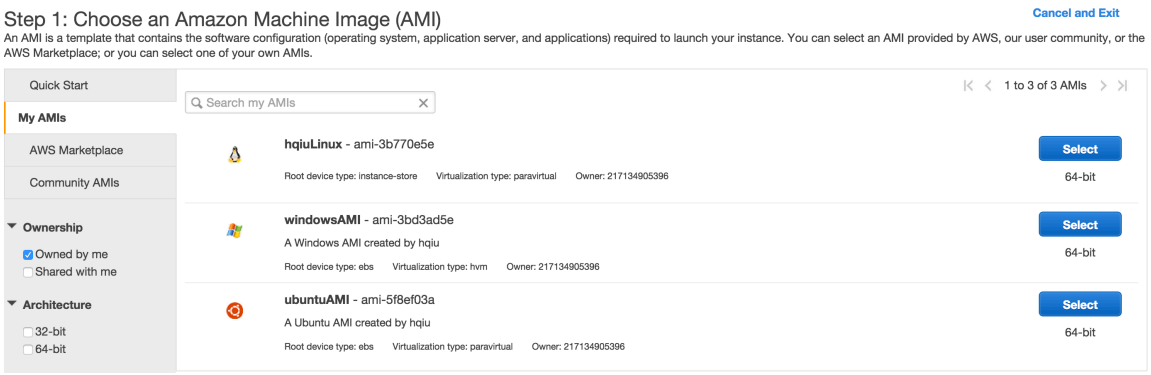
```
[root@ip-172-31-59-164 mnt]# cd /tmp
[root@ip-172-31-59-164 tmp]# vi image.manifest.xml
[root@ip-172-31-59-164 tmp]# ec2-upload-bundle -b hqiu-linux-bundle -m /tmp/image.manifest.xml
 -a AKIAIABZEJE32N2IXMZA -s U78DzRsy9WzDuHzPRx3mz6+B0sGQdHzGD38b+mLB --region us-east-1
Creating bucket...
Uploading bundled image parts to the S3 bucket hqiu-linux-bundle ...
Uploaded image.part.00
Uploaded image.part.01
Uploaded image.part.02
Uploaded image.part.03
Uploaded image.part.04
Uploaded image.part.05
Uploaded image.part.06

Uploaded image.part.36
Uploaded image.part.37
Uploaded image.part.38
Uploaded image.part.39
Uploaded image.part.40
Uploaded image.part.41
Uploaded image.part.42
Uploaded image.part.43
Uploading manifest ...
Uploaded manifest.
Manifest uploaded to: hqiu-linux-bundle/image.manifest.xml
Bundle upload completed.
```

Check AWS Console. View my bucket. Download and examine ‘image.manifest.xml’.



6. Register the image with EC2. Issue ‘aws ec2 register-image’ from my local laptop. ‘hqiu-linux-bundle’ is the name of my S3 bucket. ‘hqiuLinux’ is my new image name. The newly created image will be listed in ‘My AMIs’ too.

```
hqiu@bos-mpdei>> aws ec2 register-image --image-location hqiu-linux-bundle/image.manifest.xml
--name hqiuLinux
{
    "ImageId": "ami-3b770e5e"
}
```



ami-3b770e5e is the image id of the newly created AMI.

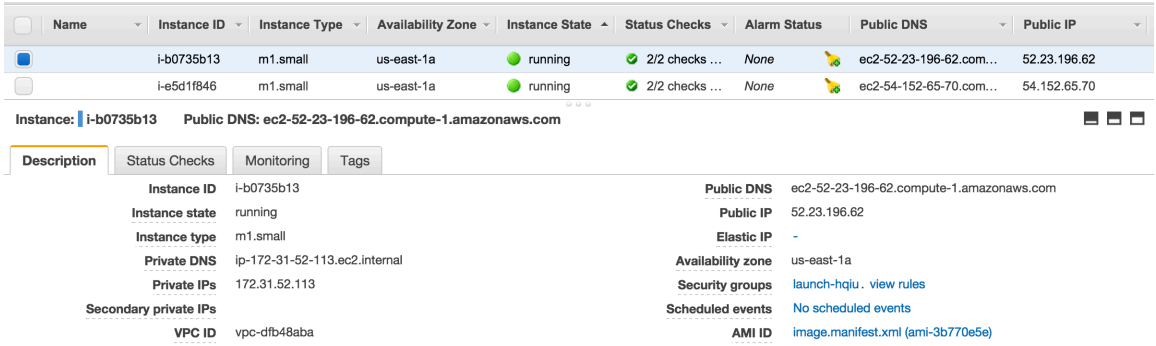
```
hqiu@bos-mpdei>> aws ec2 describe-images --image-id ami-3b770e5e
{
    "Images": [
        {
            "VirtualizationType": "paravirtual",
            "Name": "hqiuLinux",
            "Hypervisor": "xen",
            "ImageId": "ami-3b770e5e",
            "RootDeviceType": "instance-store",
            "State": "available",
            "BlockDeviceMappings": [
                {
                    "DeviceName": "sda2",
                    "VirtualName": "ephemeral0"
                }
            ],
            "Architecture": "x86_64",
            "ImageLocation": "hqiu-linux-bundle/image.manifest.xml",
            "KernelId": "aki-919dcaf8",
            "OwnerId": "217134905396",
            "RootDeviceName": "/dev/sda1",
            "CreationDate": "2015-09-25T04:42:18.000Z",
            "Public": false,
            "ImageType": "machine"
        }
    ]
}
```

7. Create an instance based on the new AMI.

```
hqiu@bos-mpdei>> aws ec2 run-instances --image-id ami-3b770e5e --count 1 --instance-type m1.sm
all --key-name ec2hqiu --security-groups launch-hqiu
{
    "OwnerId": "217134905396",
    "ReservationId": "r-629e32b4",
    "Groups": [],
    "Instances": [
        {
            "Monitoring": {
                "State": "disabled"
            },
            "PublicDnsName": "",
            "KernelId": "aki-919dcaf8",
            "State": {
                "Code": 0,
                "Name": "pending"
            },
            "EbsOptimized": false,
            "LaunchTime": "2015-09-25T04:47:16.000Z",
            "PrivateIpAddress": "172.31.52.113",
            "ProductCodes": [],
            "VpcId": "vpc-dfb48aba",
            "StateTransitionReason": "",
            "InstanceId": "i-b0735b13",
            "ImageId": "ami-3b770e5e",
            "PrivateDnsName": "ip-172-31-52-113.ec2.internal",
            "KeyName": "ec2hqiu",
            "SecurityGroups": [
                {
                    "GroupName": "launch-hqiu",
                    "GroupId": "sg-adfff3ca"
                }
```

```
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-b0735b13 --output table
-------------------------------------------------------------------------
|                           DescribeInstances                           |
+-----------------------------------------------------------------------+
||                            Reservations                             ||
|+---------------------------------+-------------------------------------+|
||  OwnerId                        |  217134905396                       ||
||  ReservationId                  |  r-629e32b4                         ||
|+---------------------------------+-------------------------------------+|
|||                             Instances                               |||
||+-------------------------+---------------------------------------------+||
|||  AmiLaunchIndex         |  0                                          |||
|||  Architecture           |  x86_64                                     |||
|||  ClientToken            |                                             |||
|||  EbsOptimized           |  False                                      |||
|||  Hypervisor             |  xen                                        |||
|||  ImageId                |  ami-3b770e5e                               |||
|||  InstanceId             |  i-b0735b13                                 |||
|||  InstanceType           |  m1.small                                   |||
|||  KernelId               |  aki-919dcaf8                               |||
|||  KeyName                |  ec2hqiu                                    |||
|||  LaunchTime             |  2015-09-25T04:47:16.000Z                   |||
|||  PrivateDnsName         |  ip-172-31-52-113.ec2.internal              |||
|||  PrivateIpAddress       |  172.31.52.113                              |||
|||  PublicDnsName          |  ec2-52-23-196-62.compute-1.amazonaws.com   |||
|||  PublicIpAddress        |  52.23.196.62                               |||
|||  RootDeviceType         |  instance-store                             |||
|||  SourceDestCheck        |  True                                       |||
|||  StateTransitionReason  |                                             |||
|||  SubnetId               |  subnet-abe07780                            |||
|||  VirtualizationType     |  paravirtual                                |||
|||  VpcId                  |  vpc-dfb48aba                               |||
||+-------------------------+---------------------------------------------+||
```

```
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-b0735b13 --query 'Reservations[0]
.Instances[0].PublicIpAddress'
"52.23.196.62"
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-b0735b13 --query 'Reservations[0]
.Instances[0].PublicDnsName'
"ec2-52-23-196-62.compute-1.amazonaws.com"
```

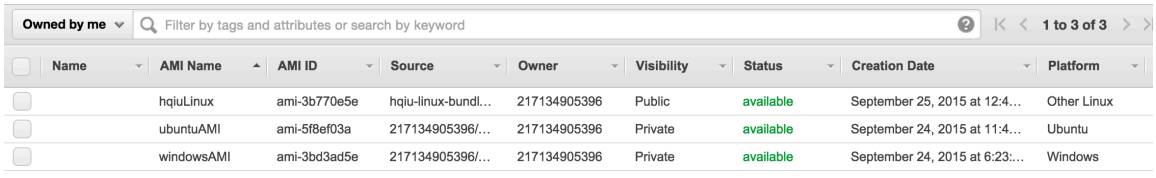


8. Check my new added file ‘myNewFile.txt’ gets survived during the process. It has the same content and is the same file.

```
hqiu@bos-mpdei>> ssh -i "ec2hqiu.pem" ec2-user@52.23.196.62
Last login: Fri Sep 25 03:45:54 2015 from c-71-233-46-151.hsd1.ma.comcast.net

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

https://aws.amazon.com/amazon-linux-ami/2015.09-release-notes/
[ec2-user@ip-172-31-59-164 ~]$ ls -lart ~/myNewFile.txt
-rw-r--r-- 1 ec2-user ec2-user 76 Sep 25 03:45 /home/ec2-user/myNewFile.txt
[ec2-user@ip-172-31-59-164 ~]$ cat ~/myNewFile.txt
This is the file that will be carried over to the new AMI.
Created by hqiu.
```

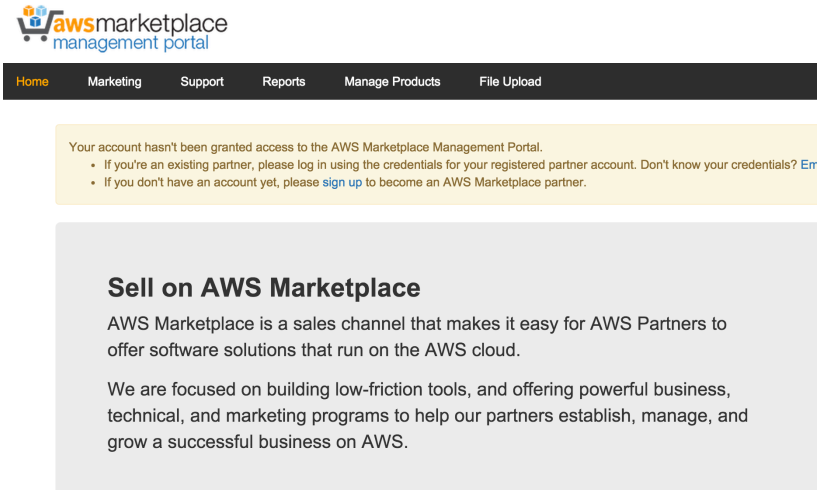
9. Grant access to my AMI (ami-3b770e5e) to the general public. Check from AWS EC2 Console, in the navigation pane, click ‘AMIs’.

```
hqiu@bos-mpdei>> aws ec2 describe-image-attribute --image-id ami-3b770e5e --attribute launchPe
rmission
{
    "LaunchPermissions": [],
    "ImageId": "ami-3b770e5e"
}
hqiu@bos-mpdei>> aws ec2 modify-image-attribute --image-id ami-3b770e5e --launch-permission "{
\"Add\": [{\"Group\":\"all\"}]}"
hqiu@bos-mpdei>> aws ec2 describe-image-attribute --image-id ami-3b770e5e --attribute launchPe
rmission
{
    "LaunchPermissions": [
        {
            "Group": "all"
        }
    ],
    "ImageId": "ami-3b770e5e"
}
```



10. Register my AMI as a paid AMI. Go to the AWS market place:
    https://aws.amazon.com/marketplace/management/tour/

We can sell our AMI using AWS Marketplace. Since I'm not a partner or AWS Market Seller, I could only stop at this step. But we should be able to sell our AMI through this.



11. Terminate the instances.

```
hqiu@bos-mpdei>> aws ec2 terminate-instances --instance-ids i-e5d1f846
{
    "TerminatingInstances": [
        {
            "InstanceId": "i-e5d1f846",
            "CurrentState": {
                "Code": 32,
                "Name": "shutting-down"
            },
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
}
hqiu@bos-mpdei>> aws ec2 terminate-instances --instance-ids i-b0735b13
{
    "TerminatingInstances": [
        {
            "InstanceId": "i-b0735b13",
            "CurrentState": {
                "Code": 32,
                "Name": "shutting-down"
            },
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
}
```

Problem 3

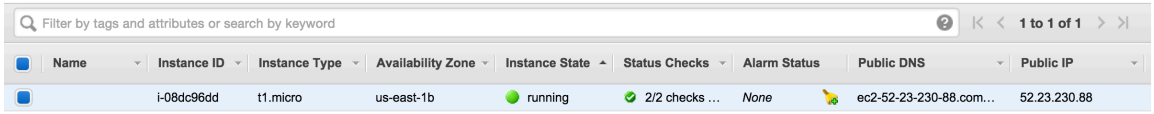
1. Create an instance based on an EBS Windows backed AMI: ami-cd9339a6.

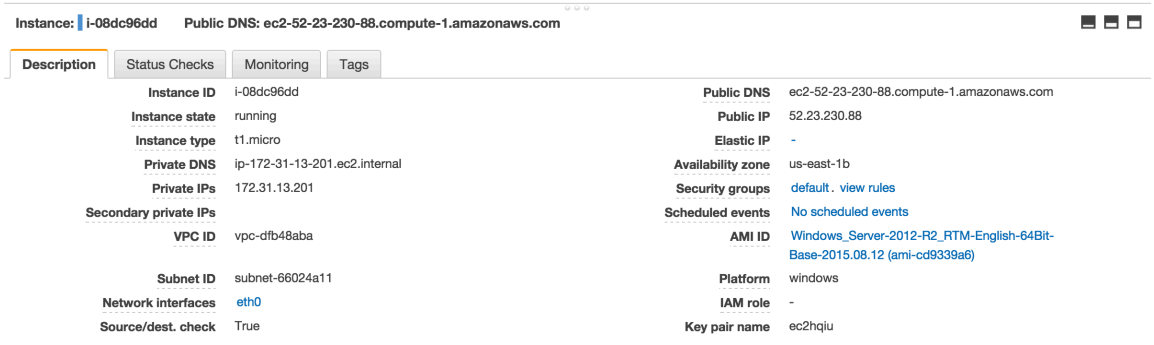
```
hqiu@bos-mpdei>> aws ec2 describe-images --image-id ami-cd9339a6
{
    "Images": [
        {
            "VirtualizationType": "hvm",
            "Name": "Windows_Server-2012-R2_RTM-English-64Bit-Base-2015.08.12",
            "Hypervisor": "xen",
            "ImageOwnerAlias": "amazon",
            "SriovNetSupport": "simple",
            "ImageId": "ami-cd9339a6",
            "Platform": "windows",
            "State": "available",
            "BlockDeviceMappings": [
                {
                    "DeviceName": "/dev/sda1",
                    "Ebs": {
                        "DeleteOnTermination": true,
                        "SnapshotId": "snap-6449f02a",
                        "VolumeSize": 30,
                        "VolumeType": "gp2",
                        "Encrypted": false
                    }
                },
```

```
hqiu@bos-mpdei>> aws ec2 run-instances --image-id ami-cd9339a6 --count 1 --instance-type t1.mi
cro --key-name ec2hqiu --security-group-ids default
{
    "OwnerId": "217134905396",
    "ReservationId": "r-96d95f6b",
    "Groups": [],
    "Instances": [
        {
            "Monitoring": {
                "State": "disabled"
            },
            "PublicDnsName": "",
            "Platform": "windows",
            "State": {
                "Code": 0,
                "Name": "pending"
            },
            "EbsOptimized": false,
            "LaunchTime": "2015-09-24T21:56:47.000Z",
            "PrivateIpAddress": "172.31.13.201",
            "ProductCodes": [],
            "VpcId": "vpc-dfb48aba",
            "StateTransitionReason": "",
            "InstanceId": "i-08dc96dd",
            "ImageId": "ami-cd9339a6",
            "PrivateDnsName": "ip-172-31-13-201.ec2.internal",
            "KeyName": "ec2hqiu",
            "SecurityGroups": [
                {
                    "GroupName": "default",
                    "GroupId": "sg-817073e5"
                }
            ],
            "ClientToken": "",
            "SubnetId": "subnet-66024a11",
            "InstanceType": "t1.micro",
```

```
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-08dc96dd --output table
```

```
|                                DescribeInstances                                |
+---------------------------------------------------------------------------------+
||                                Reservations                                   ||
|+-----------------------------------+-------------------------------------------+|
||  OwnerId                          |  217134905396                             ||
||  ReservationId                    |  r-96d95f6b                               ||
|+-----------------------------------+-------------------------------------------+|
|||                                 Instances                                     |||
||+-------------------------+-----------------------------------------------------+||
|||  AmiLaunchIndex         |  0                                                  |||
|||  Architecture           |  x86_64                                             |||
|||  ClientToken            |                                                     |||
|||  EbsOptimized           |  False                                              |||
|||  Hypervisor             |  xen                                                |||
|||  ImageId                |  ami-cd9339a6                                       |||
|||  InstanceId             |  i-08dc96dd                                         |||
|||  InstanceType           |  t1.micro                                           |||
|||  KeyName                |  ec2hqiu                                            |||
|||  LaunchTime             |  2015-09-24T21:56:47.000Z                           |||
|||  Platform               |  windows                                            |||
|||  PrivateDnsName         |  ip-172-31-13-201.ec2.internal                      |||
|||  PrivateIpAddress       |  172.31.13.201                                      |||
|||  PublicDnsName          |  ec2-52-23-230-88.compute-1.amazonaws.com           |||
|||  PublicIpAddress        |  52.23.230.88                                       |||
|||  RootDeviceName         |  /dev/sda1                                          |||
|||  RootDeviceType         |  ebs                                                |||
|||  SourceDestCheck        |  True                                               |||
|||  StateTransitionReason  |                                                     |||
|||  SubnetId               |  subnet-66024a11                                    |||
|||  VirtualizationType     |  hvm                                                |||
|||  VpcId                  |  vpc-dfb48aba                                       |||
||+-------------------------+-----------------------------------------------------+||
|||                          BlockDeviceMappings                                  |||
||+-----------------------------------+-------------------------------------------+||
|||  DeviceName                       |  /dev/sda1                                |||
||+-----------------------------------+-------------------------------------------+||
||||                                  Ebs                                        ||||
|||+-----------------------------+-----------------------------------------------+|||
||||  AttachTime                 |  2015-09-24T21:56:51.000Z                     ||||
||||  DeleteOnTermination        |  True                                         ||||
||||  Status                     |  attached                                     ||||
||||  VolumeId                   |  vol-94a66a74                                 ||||
```

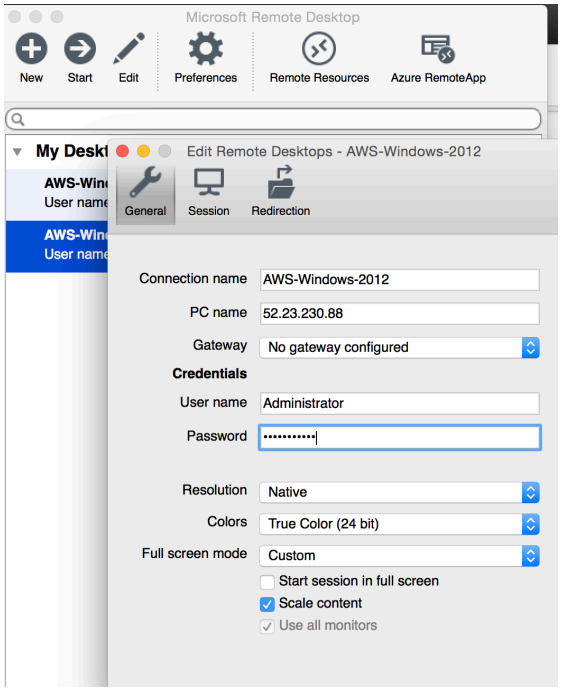


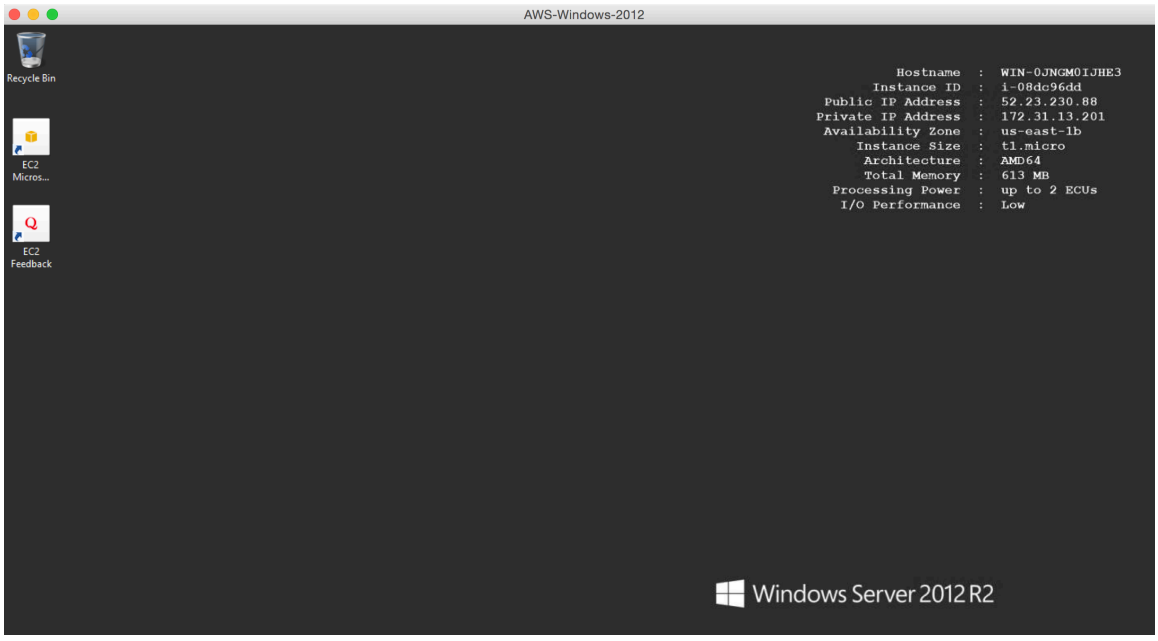



```
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-08dc96dd --query 'Reservations[0]
.Instances[0].PublicIpAddress'
"52.23.230.88"
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-08dc96dd --query 'Reservations[0]
.Instances[0].PublicDnsName'
"ec2-52-23-230-88.compute-1.amazonaws.com"
```

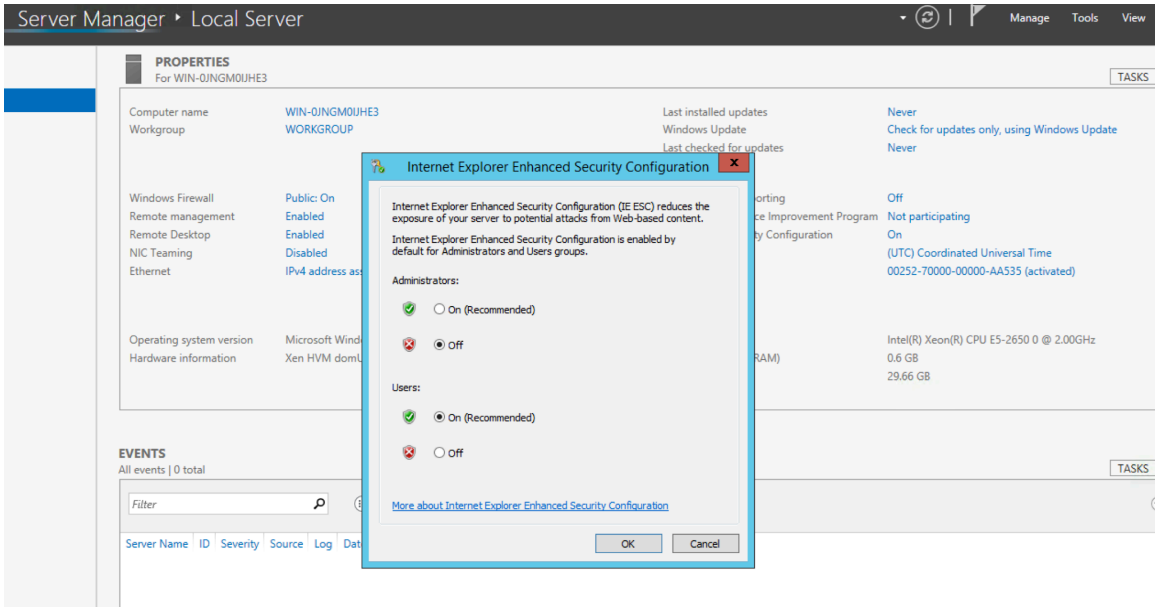
2. Get the password for the instance. Connect to the instance through ‘Microsoft Remote Desktop’.

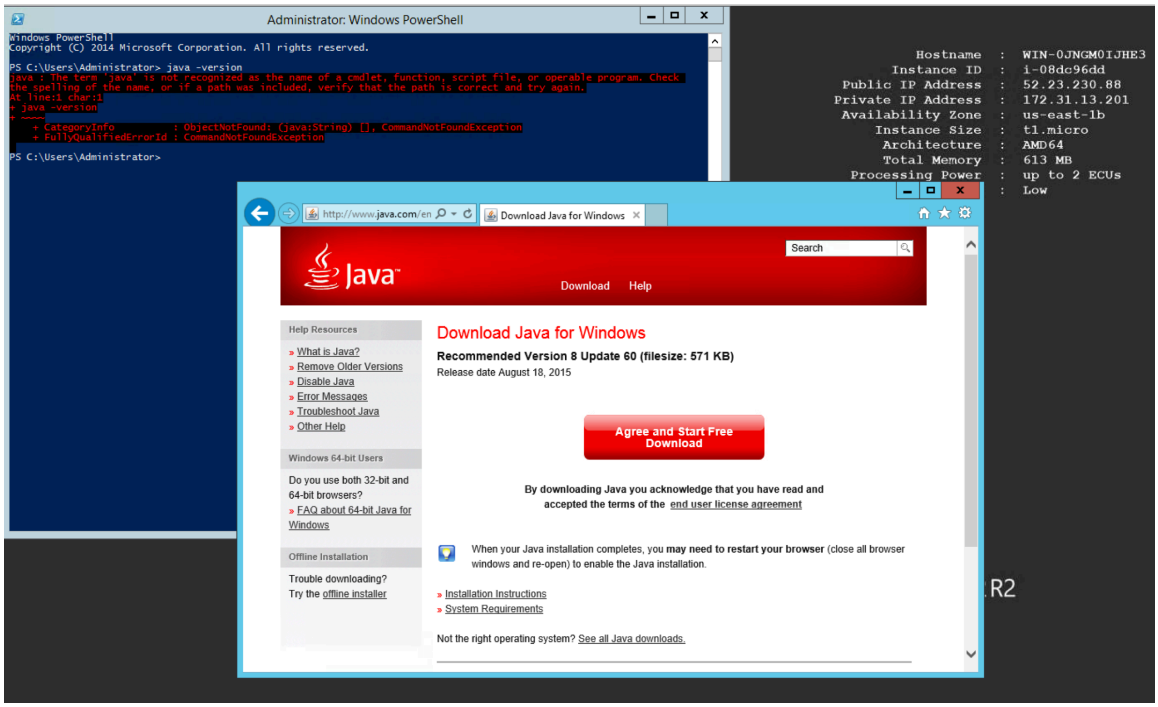
```
hqiu@bos-mpdei>> aws ec2 get-password-data --priv-launch-key ec2hqiu.pem --instance-id i-08dc9
6dd
{
    "InstanceId": "i-08dc96dd",
    "Timestamp": "2015-09-24T21:59:55.000Z",
    "PasswordData": "sGRehp.eR97"
}
```

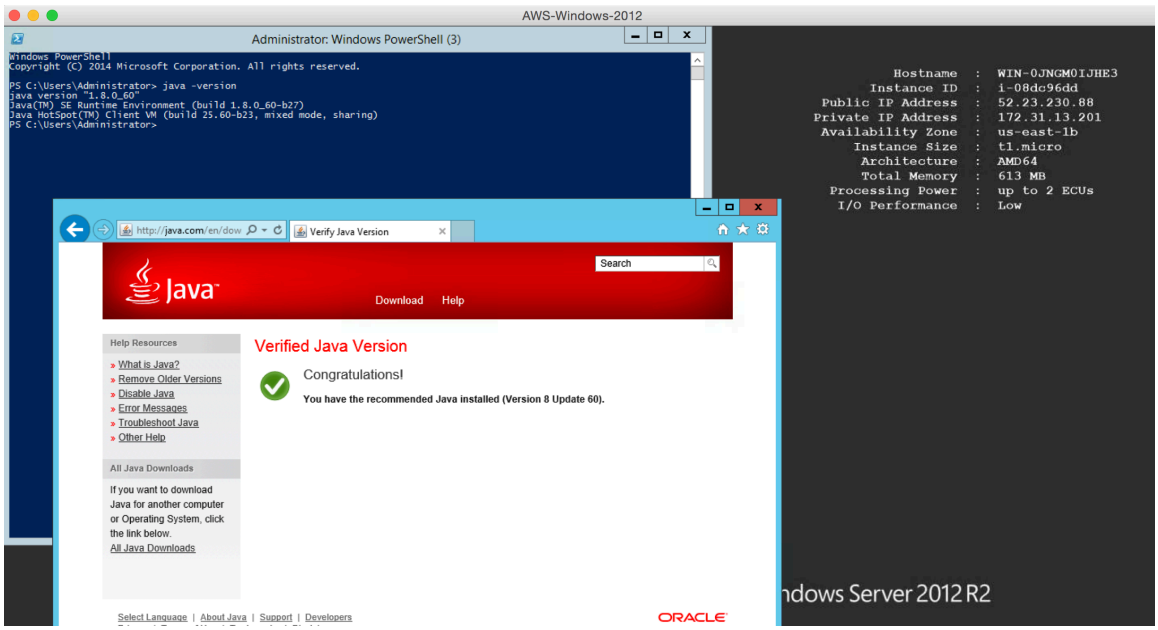


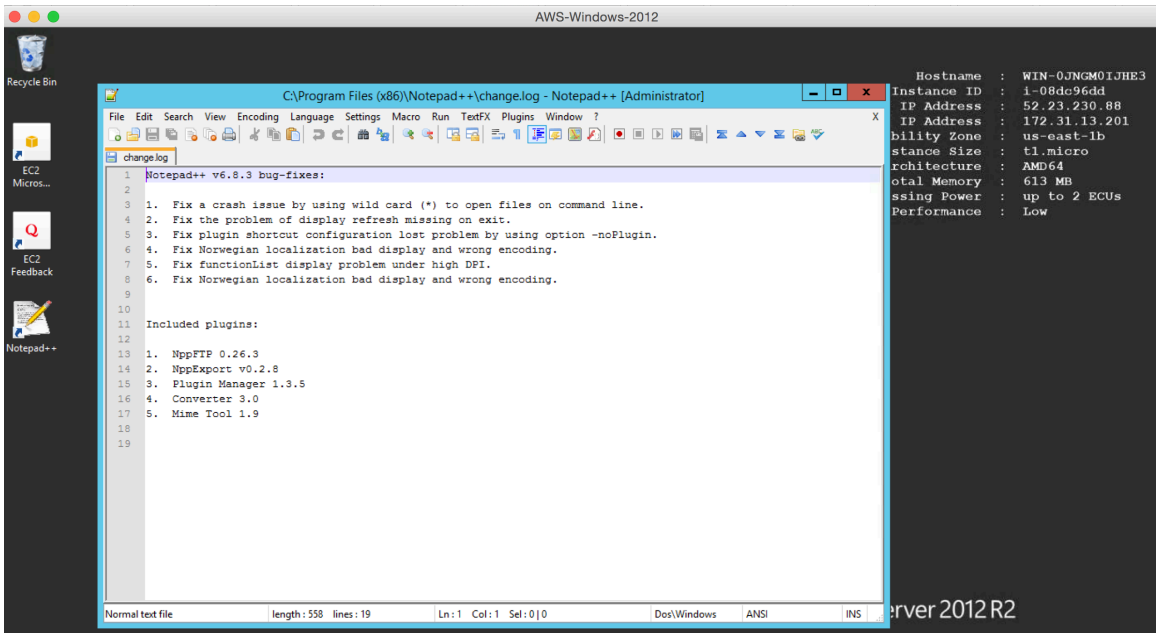


3. Do some changes to the instance. I installed Java and Notepad++ onto it.



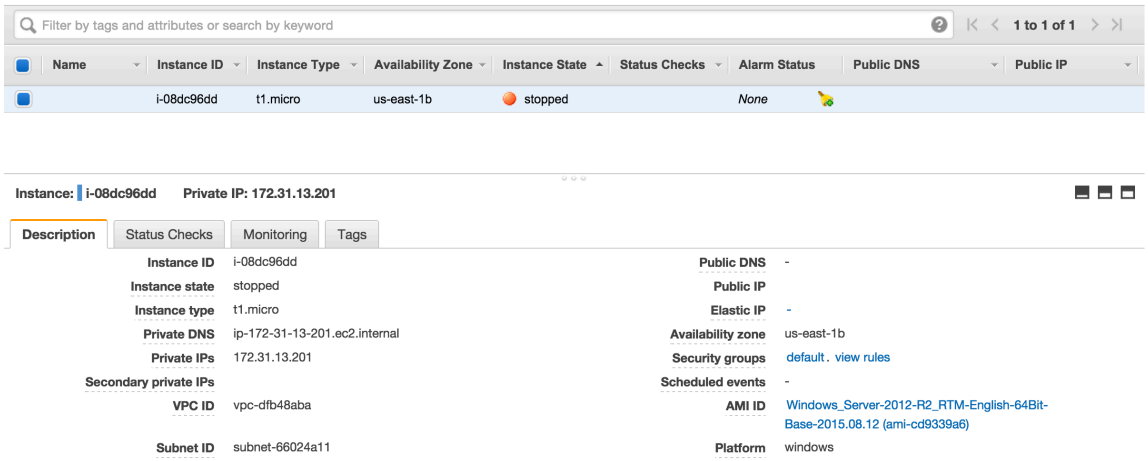






4. Stop the instance and create an image from the instance.

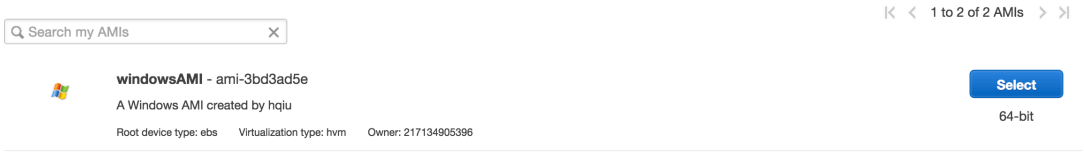
```
hqiu@bos-mpdei>> aws ec2 stop-instances --instance-ids i-08dc96dd
{
    "StoppingInstances": [
        {
            "InstanceId": "i-08dc96dd",
            "CurrentState": {
                "Code": 64,
                "Name": "stopping"
            },
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
}
```



5. Create the image and check the newly create image.

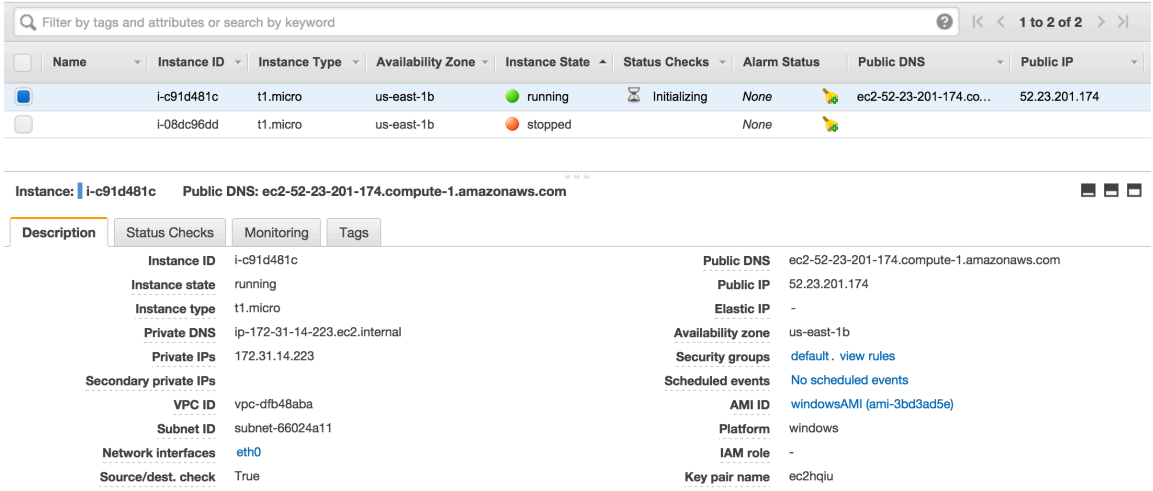
```
hqiu@bos-mpdei>> aws ec2 create-image --instance-id i-08dc96dd --name "windowsAMI" --descripti
on "A Windows AMI created by hqiu"
{
    "ImageId": "ami-3bd3ad5e"
}
    hqiu@bos-mpdei>> aws ec2 describe-images --image-id ami-3bd3ad5e
    {
        "Images": [
            {
                "VirtualizationType": "hvm",
                "Name": "windowsAMI",
                "Hypervisor": "xen",
                "SriovNetSupport": "simple",
                "ImageId": "ami-3bd3ad5e",
                "Platform": "windows",
                "State": "pending",
                "BlockDeviceMappings": [
                    {
                        "DeviceName": "/dev/sda1",
                        "Ebs": {
                            "DeleteOnTermination": true,
                            "SnapshotId": "snap-b9d1e7cb",
                            "VolumeSize": 30,
                            "VolumeType": "gp2",
                            "Encrypted": false
                        }
                    },
    hqiu@bos-mpdei>> aws ec2 describe-images --image-id ami-3bd3ad5e --output table
    ------------------------------------------------------------
    |                     DescribeImages                       |
    +----------------------------------------------------------+
    ||                        Images                          ||
    |+----------------------+---------------------------------+|
    ||  Architecture        |  x86_64                         ||
    ||  CreationDate        |  2015-09-24T22:23:48.000Z       ||
    ||  Description         |  A Windows AMI created by hqiu  ||
    ||  Hypervisor          |  xen                            ||
    ||  ImageId             |  ami-3bd3ad5e                   ||
    ||  ImageLocation       |  217134905396/windowsAMI        ||
    ||  ImageType           |  machine                        ||
    ||  Name                |  windowsAMI                     ||
    ||  OwnerId             |  217134905396                   ||
    ||  Platform            |  windows                        ||
    ||  Public              |  False                          ||
    ||  RootDeviceName      |  /dev/sda1                      ||
    ||  RootDeviceType      |  ebs                            ||
    ||  SriovNetSupport     |  simple                         ||
    ||  State               |  pending                        ||
    ||  VirtualizationType  |  hvm                            ||
    |+----------------------+---------------------------------+|
```

Check from the AWS Console.



6. Create an instance based on my newly created AMI ami-3bd3ad5e.

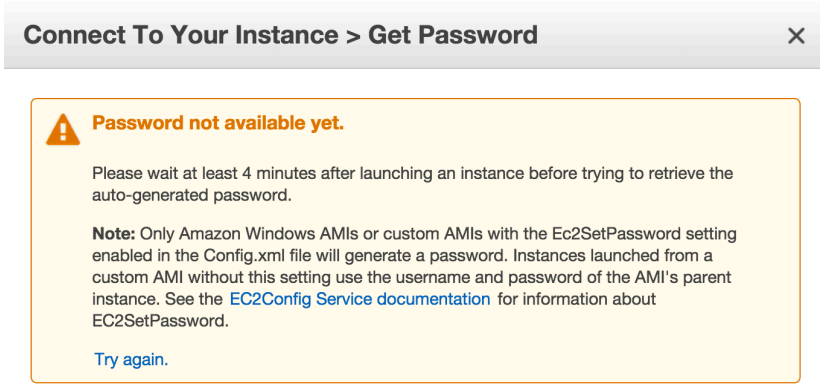
```
hqiu@bos-mpdei>> aws ec2 run-instances --image-id ami-3bd3ad5e --count 1 --instance-type t1.mi
cro --key-name ec2hqiu --security-group-ids default
{
    "OwnerId": "217134905396",
    "ReservationId": "r-8bfb7d76",
    "Groups": [],
    "Instances": [
        {
            "Monitoring": {
                "State": "disabled"
            },
            "PublicDnsName": "",
            "Platform": "windows",
            "State": {
                "Code": 0,
                "Name": "pending"
            },
            "EbsOptimized": false,
            "LaunchTime": "2015-09-24T22:26:00.000Z",
            "PrivateIpAddress": "172.31.14.223",
            "ProductCodes": [],
            "VpcId": "vpc-dfb48aba",
            "StateTransitionReason": "",
            "InstanceId": "i-c91d481c",
            "ImageId": "ami-3bd3ad5e",
            "PrivateDnsName": "ip-172-31-14-223.ec2.internal",
            "KeyName": "ec2hqiu",
            "SecurityGroups": [
                {
                    "GroupName": "default",
                    "GroupId": "sg-817073e5"
                }
            ],
```



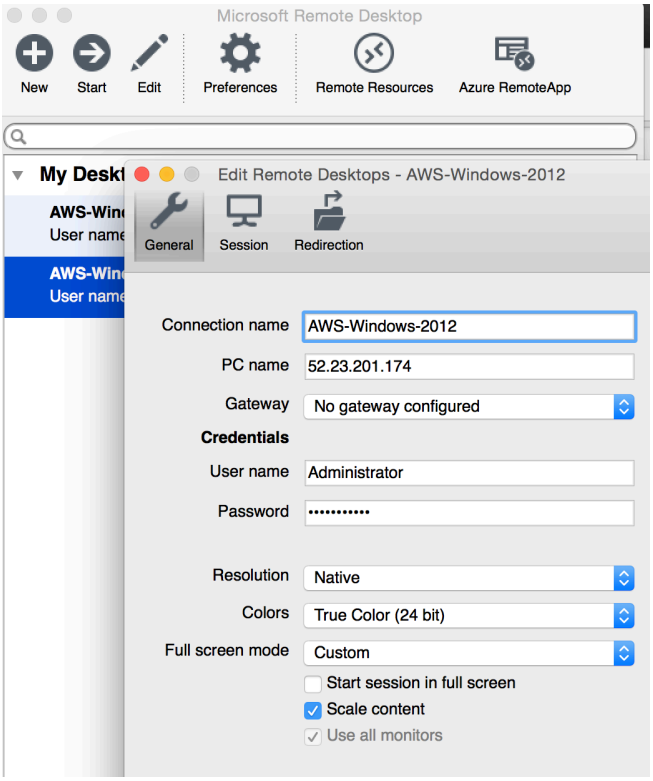
```
hqiu@bos-mpdei>> aws ec2 describe-instances --instance-ids i-c91d481c --output table
-------------------------------------------------------------------------------
|                           DescribeInstances                                 |
+-----------------------------------------------------------------------------+
||                            Reservations                                   ||
|+-----------------------------------+-------------------------------------+|
||  OwnerId                          |  217134905396                       ||
||  ReservationId                    |  r-8bfb7d76                         ||
|+-----------------------------------+-------------------------------------+|
|||                              Instances                                  |||
||+-------------------------+---------------------------------------------+||
|||  AmiLaunchIndex         |  0                                          |||
|||  Architecture           |  x86_64                                     |||
|||  ClientToken            |                                             |||
|||  EbsOptimized           |  False                                      |||
|||  Hypervisor             |  xen                                        |||
|||  ImageId                |  ami-3bd3ad5e                               |||
|||  InstanceId             |  i-c91d481c                                 |||
|||  InstanceType           |  t1.micro                                   |||
|||  KeyName                |  ec2hqiu                                    |||
|||  LaunchTime             |  2015-09-24T22:26:00.000Z                   |||
|||  Platform               |  windows                                    |||
|||  PrivateDnsName         |  ip-172-31-14-223.ec2.internal              |||
|||  PrivateIpAddress       |  172.31.14.223                              |||
|||  PublicDnsName          |  ec2-52-23-201-174.compute-1.amazonaws.com  |||
|||  PublicIpAddress        |  52.23.201.174                              |||
|||  RootDeviceName         |  /dev/sda1                                  |||
|||  RootDeviceType         |  ebs                                        |||
|||  SourceDestCheck        |  True                                       |||
|||  StateTransitionReason  |                                             |||
|||  SubnetId               |  subnet-66024a11                            |||
|||  VirtualizationType     |  hvm                                        |||
|||  VpcId                  |  vpc-dfb48aba                               |||
||+-------------------------+---------------------------------------------+||
```

7. Since it's my custom AMI, it won't automatically get a new password. The password is the one from my original instance.

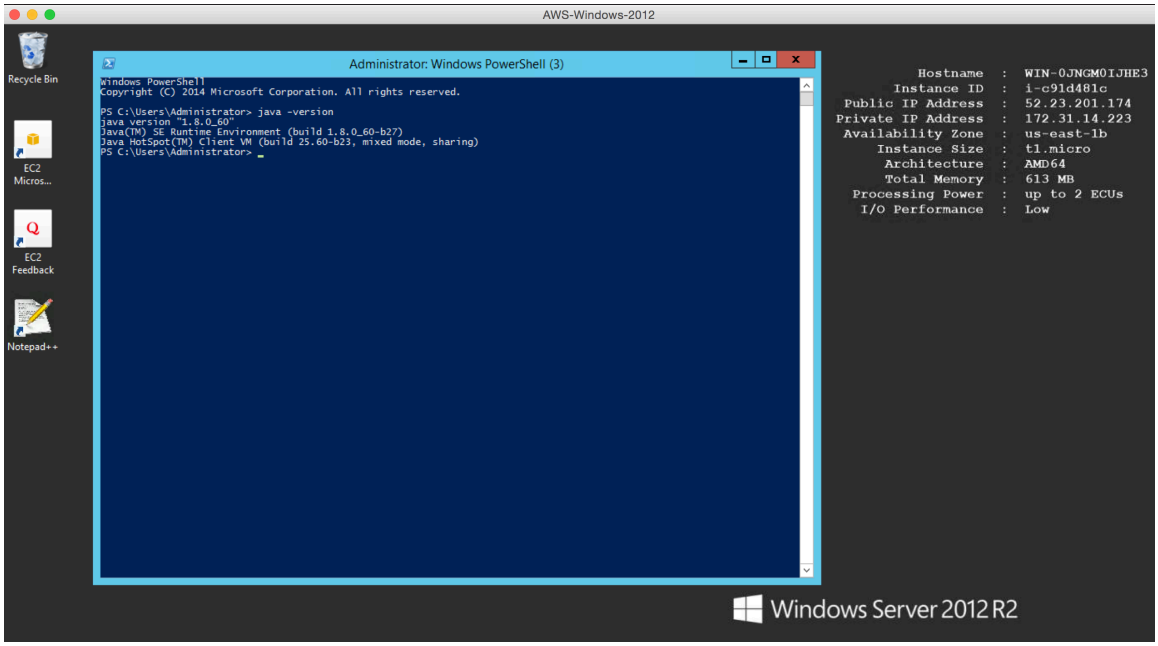
```
hqiu@bos-mpdei>> aws ec2 get-password-data --priv-launch-key ec2hqiu.pem --instance-id i-c91d4
81c
{
    "InstanceId": "i-c91d481c",
    "Timestamp": "2015-09-24T22:29:01.000Z",
    "PasswordData": ""
}
```



8. Connect to the new instance using the original password. Check if Java and Notepad++ has already been installed.



Verify the software is installed.



9. Terminate the instances.

```
hqiu@bos-mpdei>> aws ec2 terminate-instances --instance-ids i-08dc96dd
{
    "TerminatingInstances": [
        {
            "InstanceId": "i-08dc96dd",
            "CurrentState": {
                "Code": 48,
                "Name": "terminated"
            },
            "PreviousState": {
                "Code": 80,
                "Name": "stopped"
            }
        }
    ]
}
hqiu@bos-mpdei>> aws ec2 terminate-instances --instance-ids i-c91d481c
{
    "TerminatingInstances": [
        {
            "InstanceId": "i-c91d481c",
            "CurrentState": {
                "Code": 32,
                "Name": "shutting-down"
            },
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
}
```