Hadoop Single Node Setup

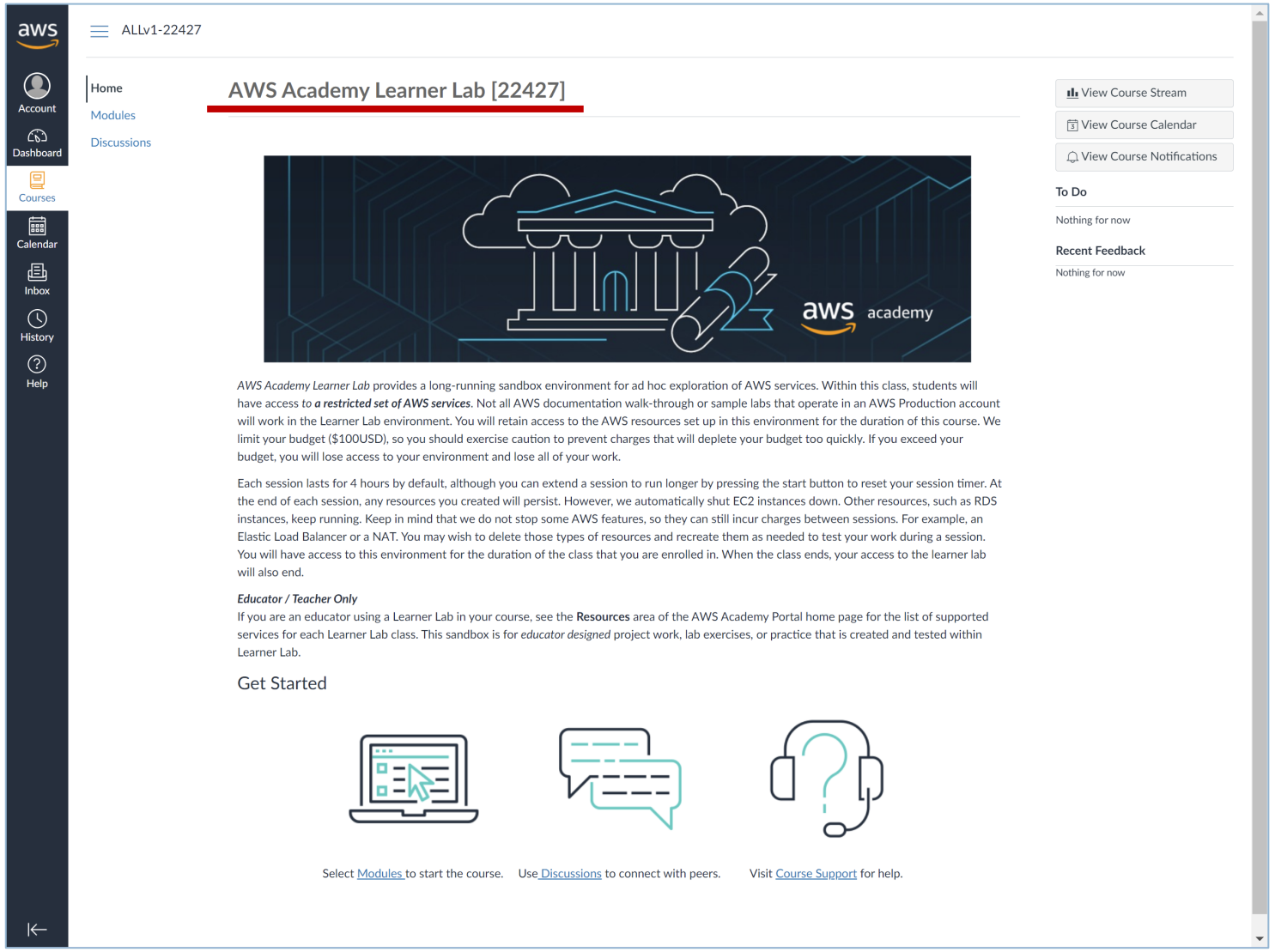
Sorayut Glomglome

Content

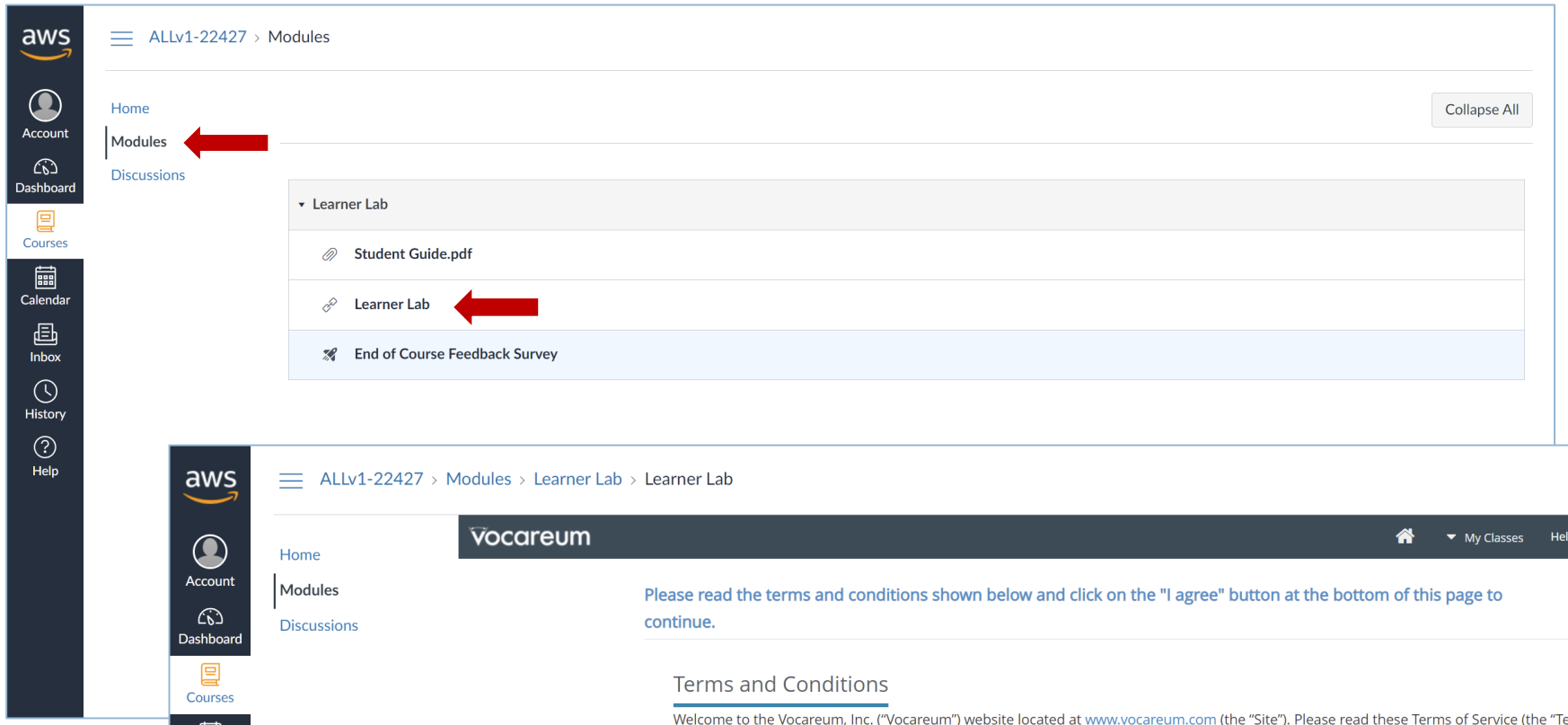
1. Launch a virtual server on EC2 Amazon Web Services
2. Installing Apache Hadoop
3. Create instance image

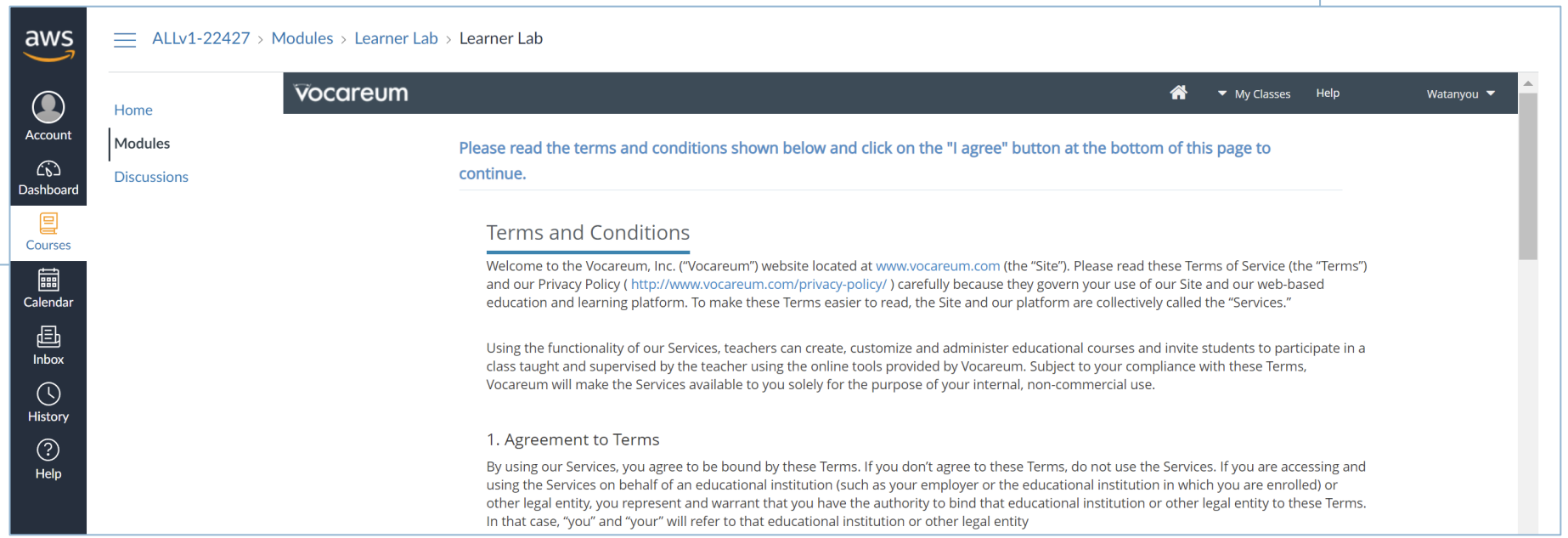
AWS Academy

AWS Academy

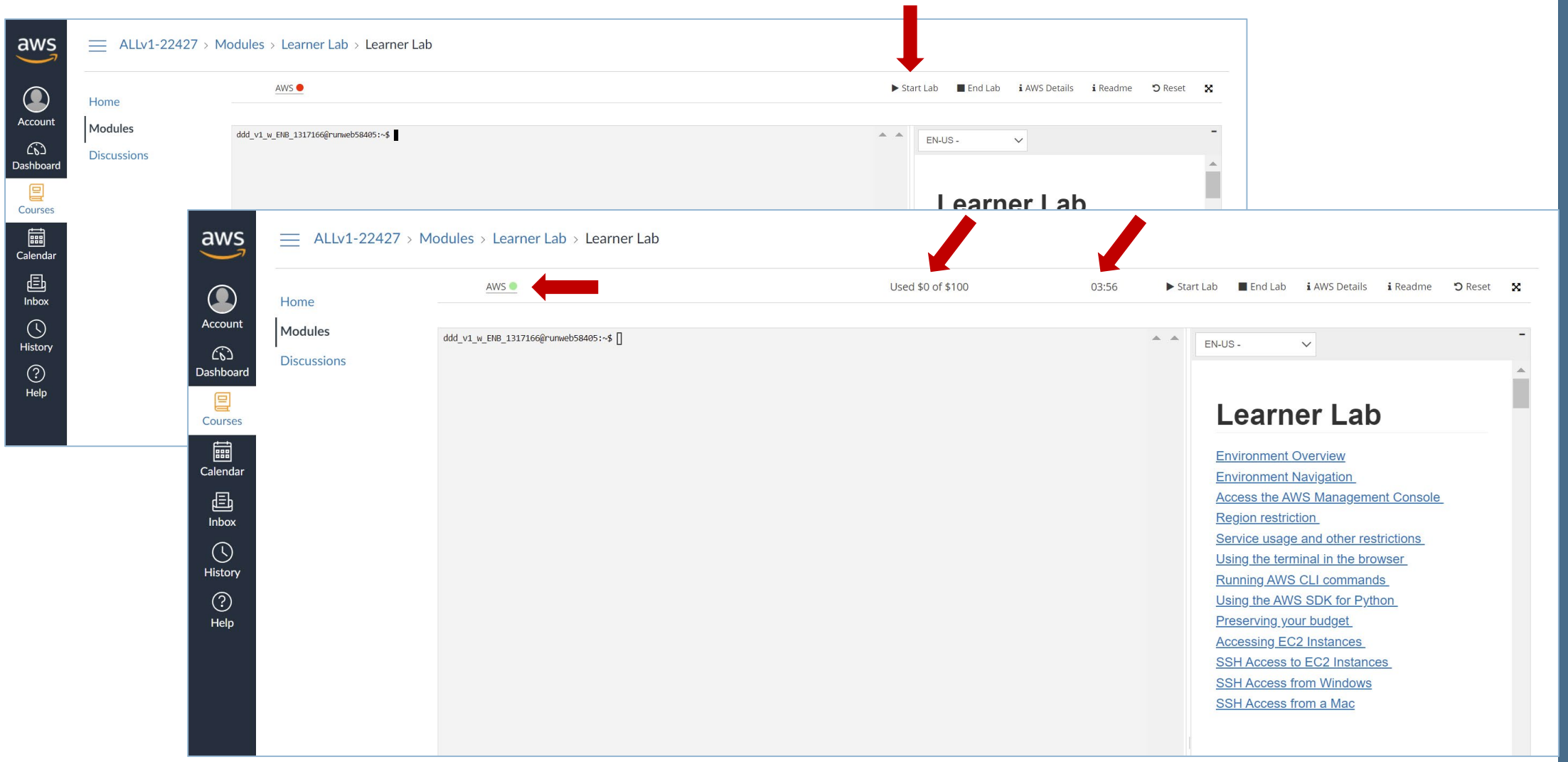

aws
ALLv1-22427
Account
Dashboard
Courses
Calendar
Inbox
History
Help
Home
Modules
Discussions
AWS Academy Learner Lab [22427]
View Course Stream
View Course Calendar
View Course Notifications
To Do
Nothing for now
Recent Feedback
Nothing for now
aws academy
AWS Academy Learner Lab provides a long-running sandbox environment for ad hoc exploration of AWS services. Within this class, students will have access to a restricted set of AWS services. Not all AWS documentation walk-through or sample labs that operate in an AWS Production account will work in the Learner Lab environment. You will retain access to the AWS resources set up in this environment for the duration of this course. We limit your budget ($100USD), so you should exercise caution to prevent charges that will deplete your budget too quickly. If you exceed your budget, you will lose access to your environment and lose all of your work.
Each session lasts for 4 hours by default, although you can extend a session to run longer by pressing the start button to reset your session timer. At the end of each session, any resources you created will persist. However, we automatically shut EC2 instances down. Other resources, such as RDS instances, keep running. Keep in mind that we do not stop some AWS features, so they can still incur charges between sessions. For example, an Elastic Load Balancer or a NAT. You may wish to delete those types of resources and recreate them as needed to test your work during a session. You will have access to this environment for the duration of the class that you are enrolled in. When the class ends, your access to the learner lab will also end.
Educator / Teacher Only
If you are an educator using a Learner Lab in your course, see the Resources area of the AWS Academy Portal home page for the list of supported services for each Learner Lab class. This sandbox is for educator designed project work, lab exercises, or practice that is created and tested within Learner Lab.
Get Started
Select Modules to start the course.
Use Discussions to connect with peers.
Visit Course Support for help.

Learner Lab - Vocareum

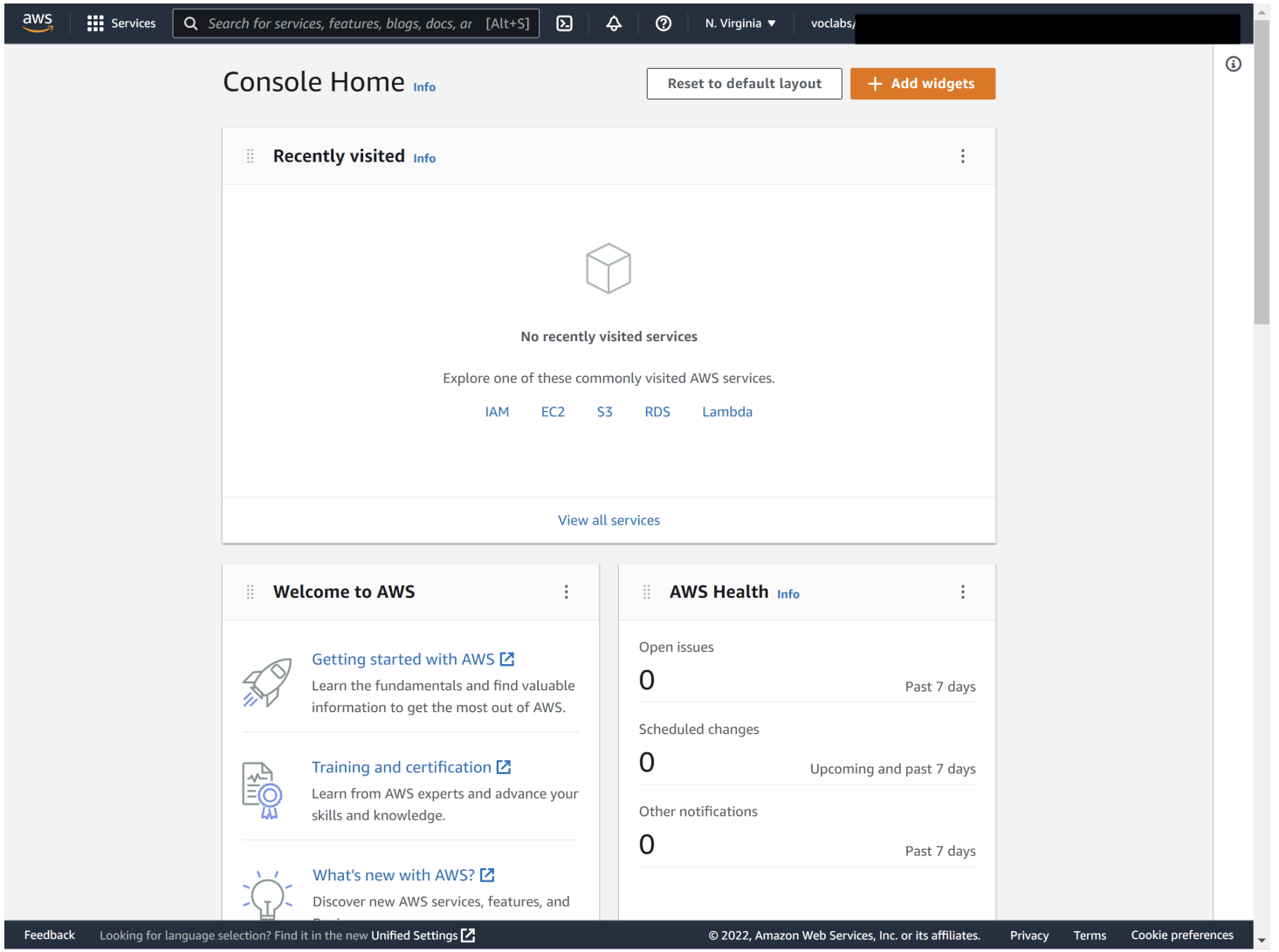




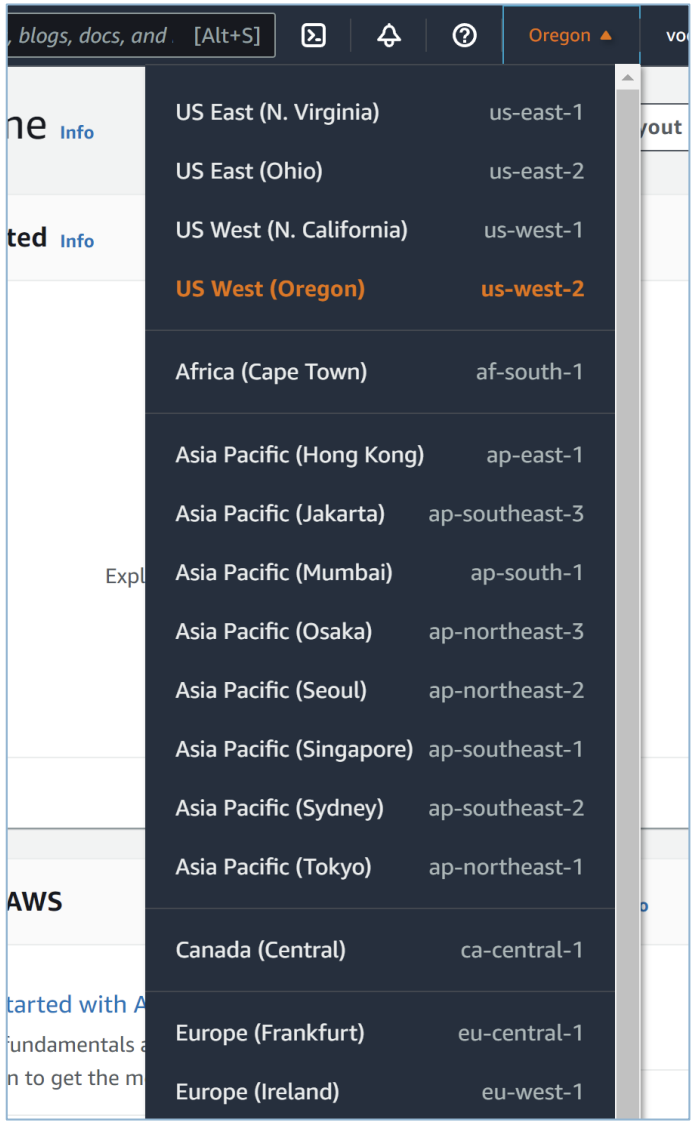
Learner Lab - Vocareum



AWS Console


aws
Services
Search for services, features, blogs, docs, ar [Alt+S]
N. Virginia
voclabs/
Console Home Info
Reset to default layout
+ Add widgets
Recently visited Info
No recently visited services
Explore one of these commonly visited AWS services.
IAM EC2 S3 RDS Lambda
View all services
Welcome to AWS
Getting started with AWS
Learn the fundamentals and find valuable information to get the most out of AWS.
Training and certification
Learn from AWS experts and advance your skills and knowledge.
What's new with AWS?
Discover new AWS services, features, and
AWS Health Info
Open issues
0
Past 7 days
Scheduled changes
0
Upcoming and past 7 days
Other notifications
0
Past 7 days
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

AWS Regions


blogs, docs, and [Alt+S]
Oregon
US East (N. Virginia) us-east-1
US East (Ohio) us-east-2
US West (N. California) us-west-1
US West (Oregon) us-west-2
Africa (Cape Town) af-south-1
Asia Pacific (Hong Kong) ap-east-1
Asia Pacific (Jakarta) ap-southeast-3
Asia Pacific (Mumbai) ap-south-1
Asia Pacific (Osaka) ap-northeast-3
Asia Pacific (Seoul) ap-northeast-2
Asia Pacific (Singapore) ap-southeast-1
Asia Pacific (Sydney) ap-southeast-2
Asia Pacific (Tokyo) ap-northeast-1
Canada (Central) ca-central-1
Europe (Frankfurt) eu-central-1
Europe (Ireland) eu-west-1

1. Launch a virtual server on EC2 Amazon Web Services

Hadoop Installation

Hadoop provides three installation choices:

1. **Local mode:** This is an unzip and run mode to get you started right away where all parts of Hadoop run within the same JVM
2. **Pseudo distributed mode**: This mode will be run on different parts of Hadoop as different Java processors, but within a single machine
3. **Distributed mode:** This is the real setup that spans multiple machines

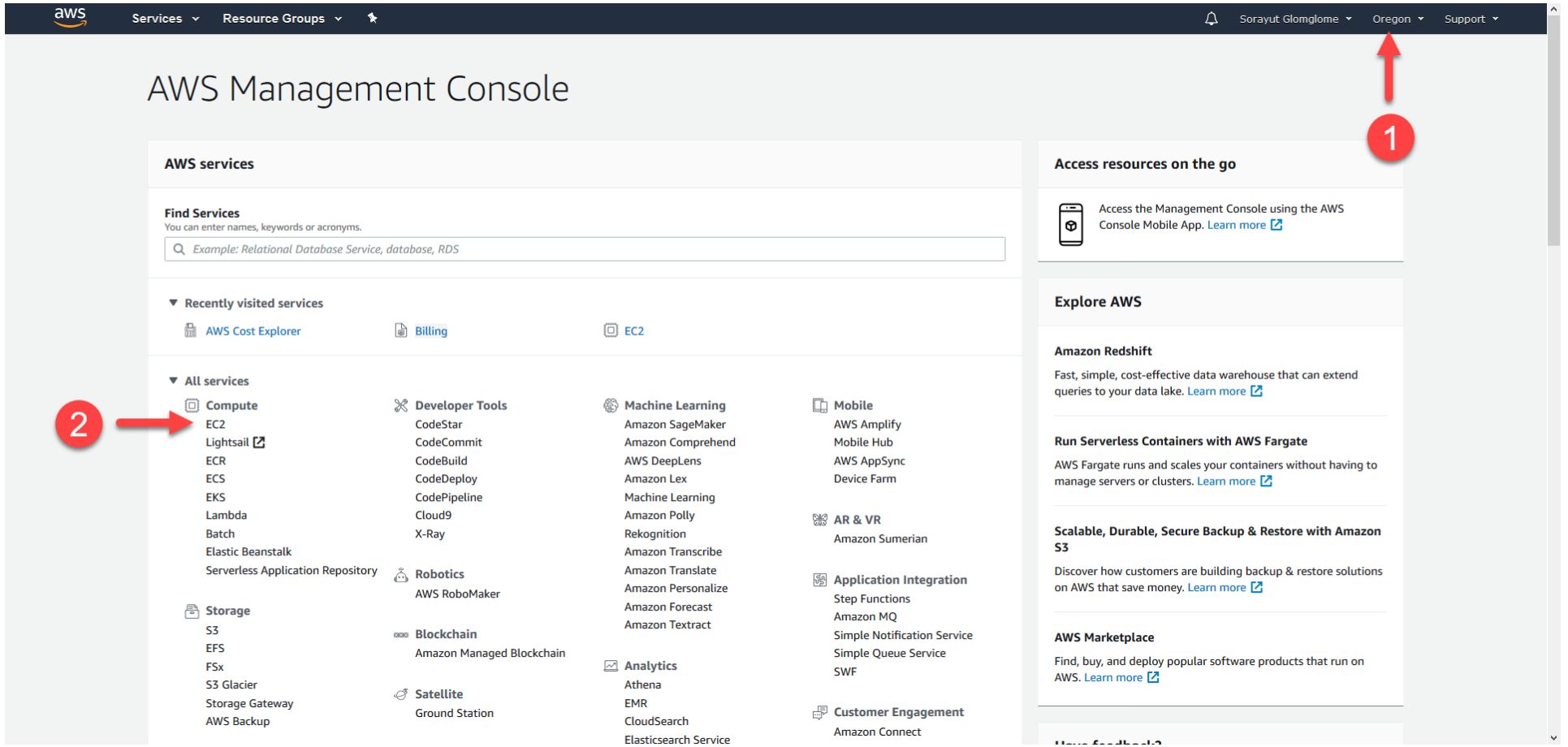
Virtual Server

This lab will use a EC2 virtual server to install a Hadoop server using the following features:

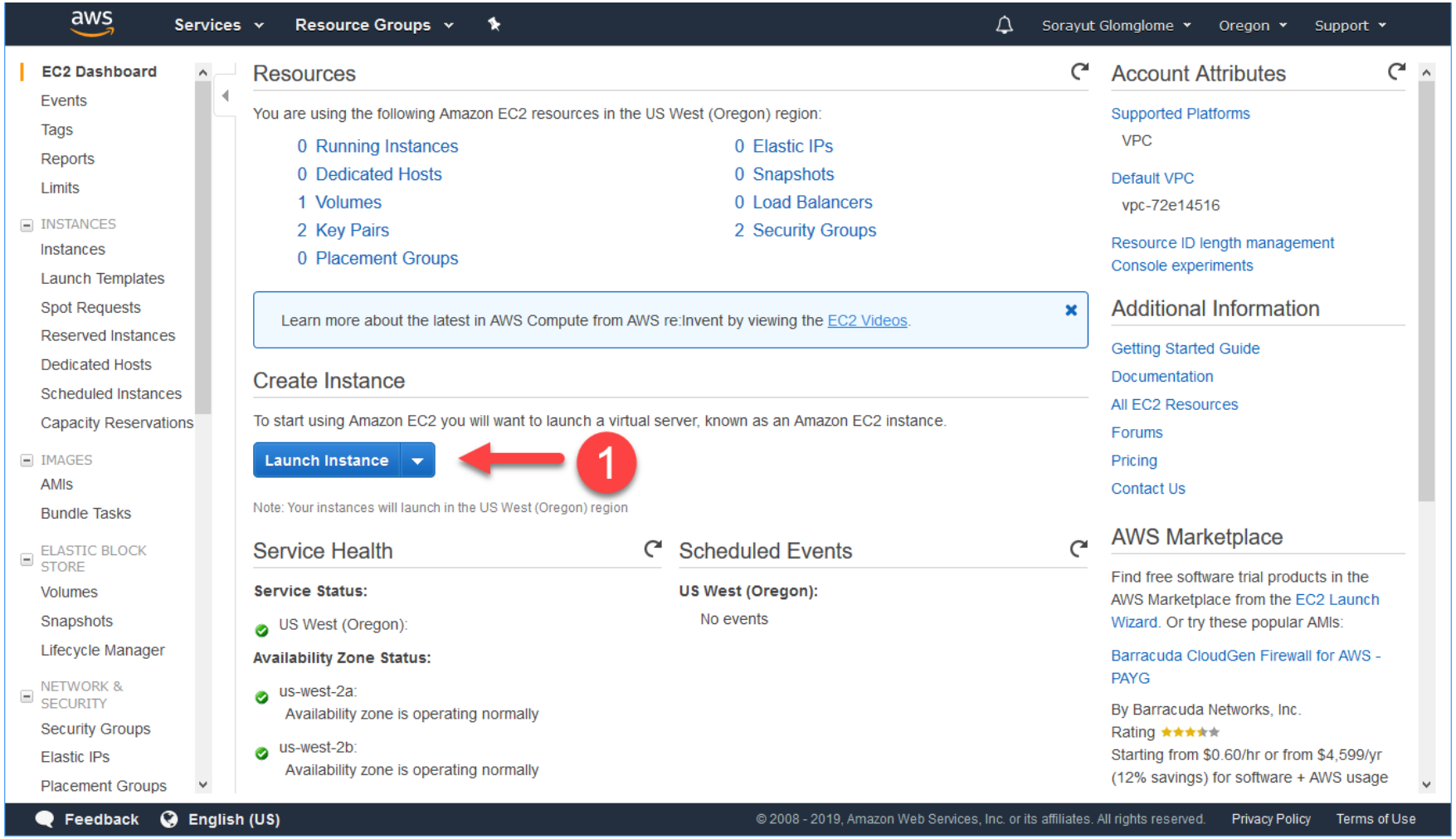
~~Ubuntu Server 16.04 LTS (HVM)~~
~~m5.large 2vCPU, 8GB memory~~

Ubuntu Server 18.04 LTS (HVM)
t2.large 2vCPU, 4GB memory

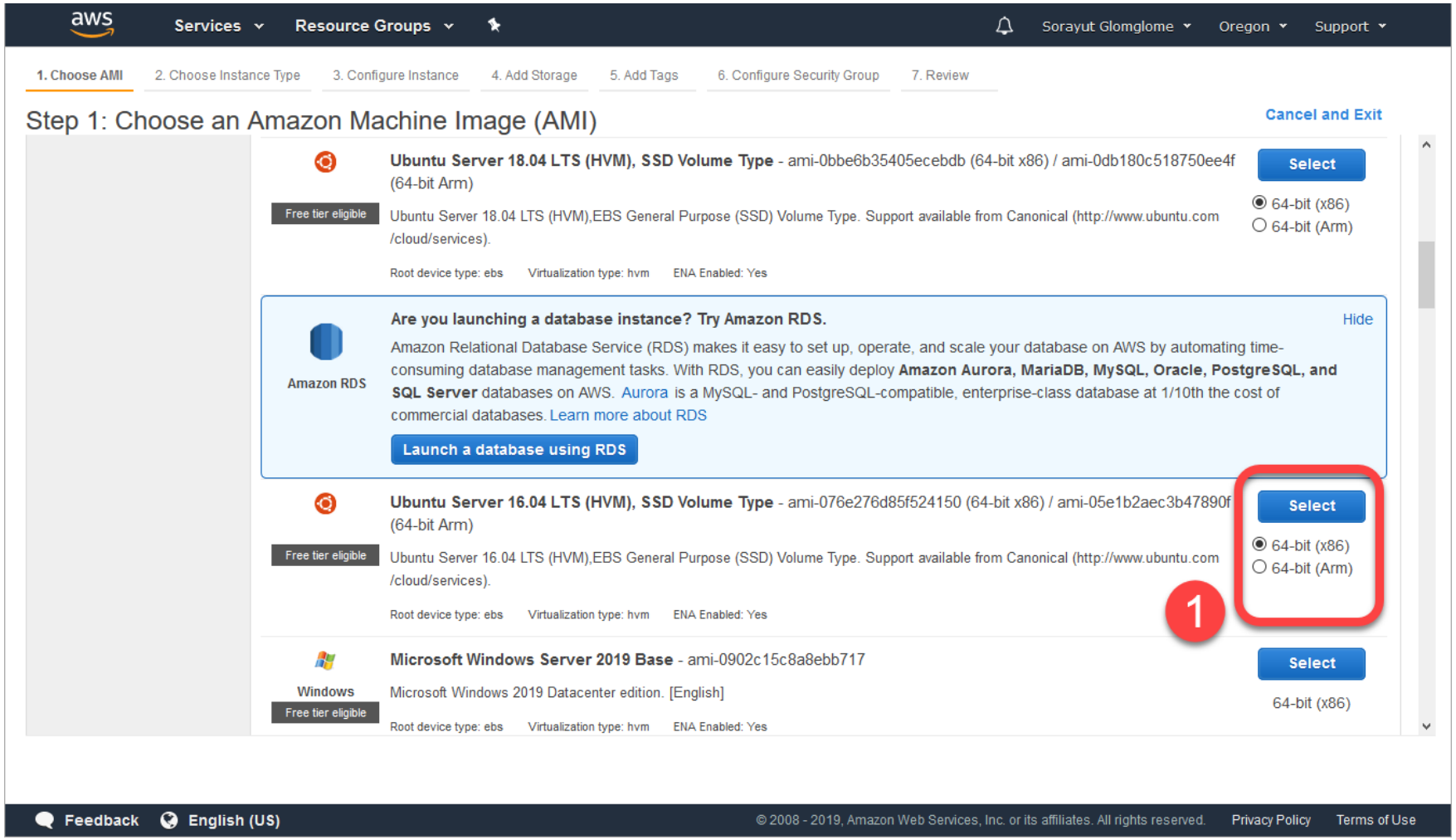
Log in to Amazon Web Services
Select a datacenter location
Select EC2 service



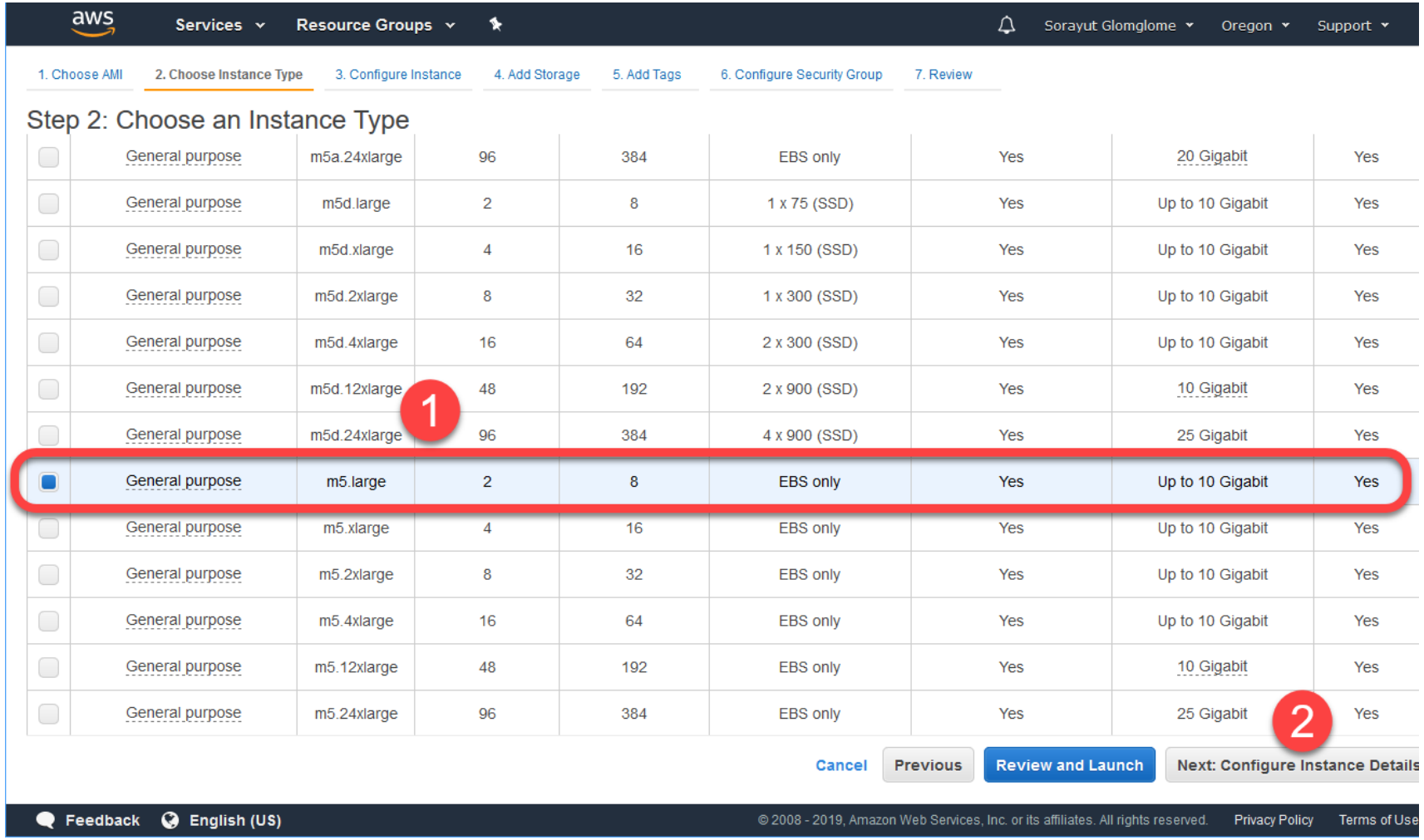
Select Launch Instance



Select Ubuntu Server 16.04 LTS (HVM)



Select m5.large



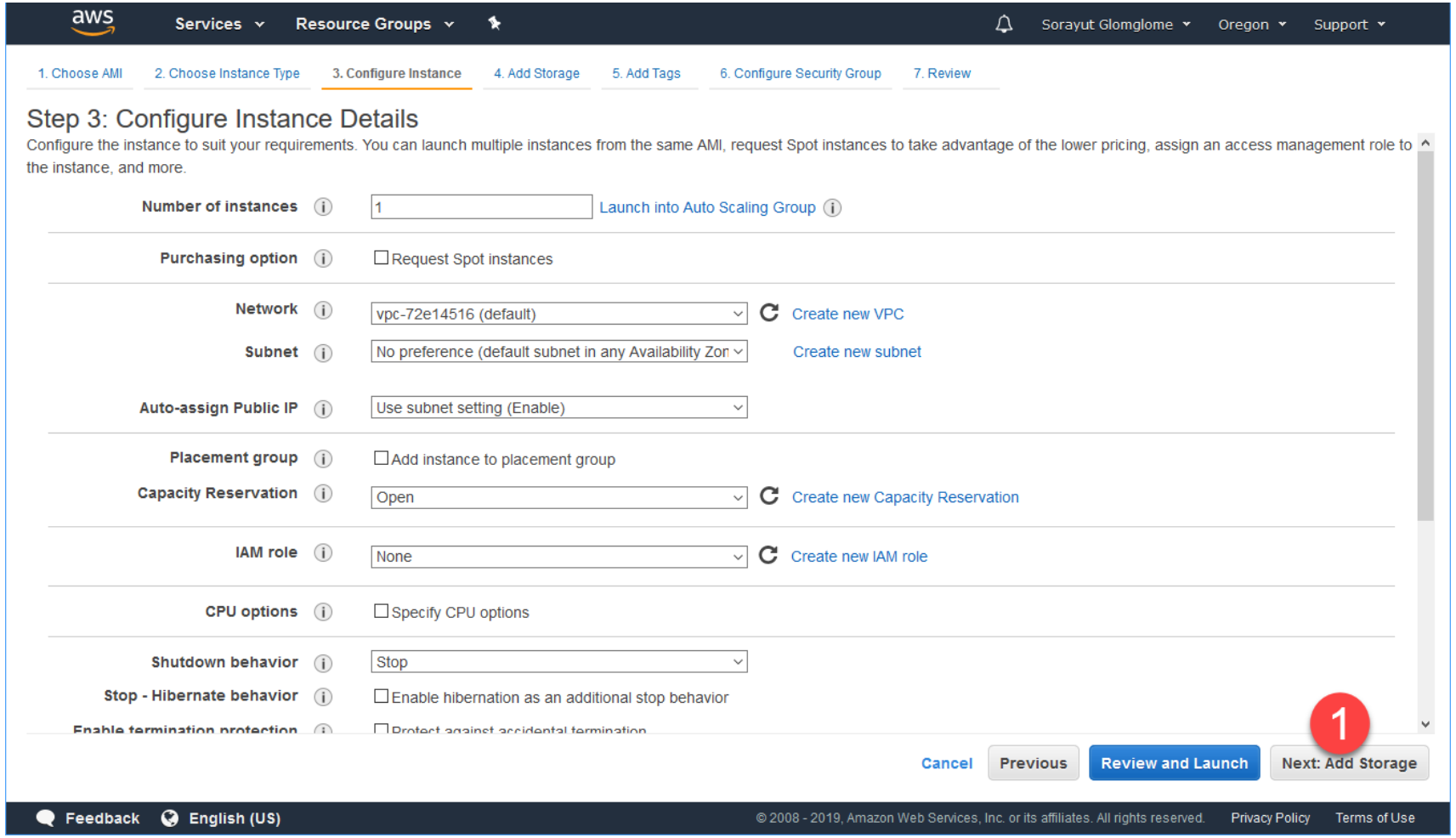
EC2 pricing example

Linux | RHEL | SLES | Windows | Windows with SQL Standard | Windows with SQL Web | Windows with SQL Enterprise | Linux with SQL Standard | Linux with SQL Web | Linux with SQL Enterprise

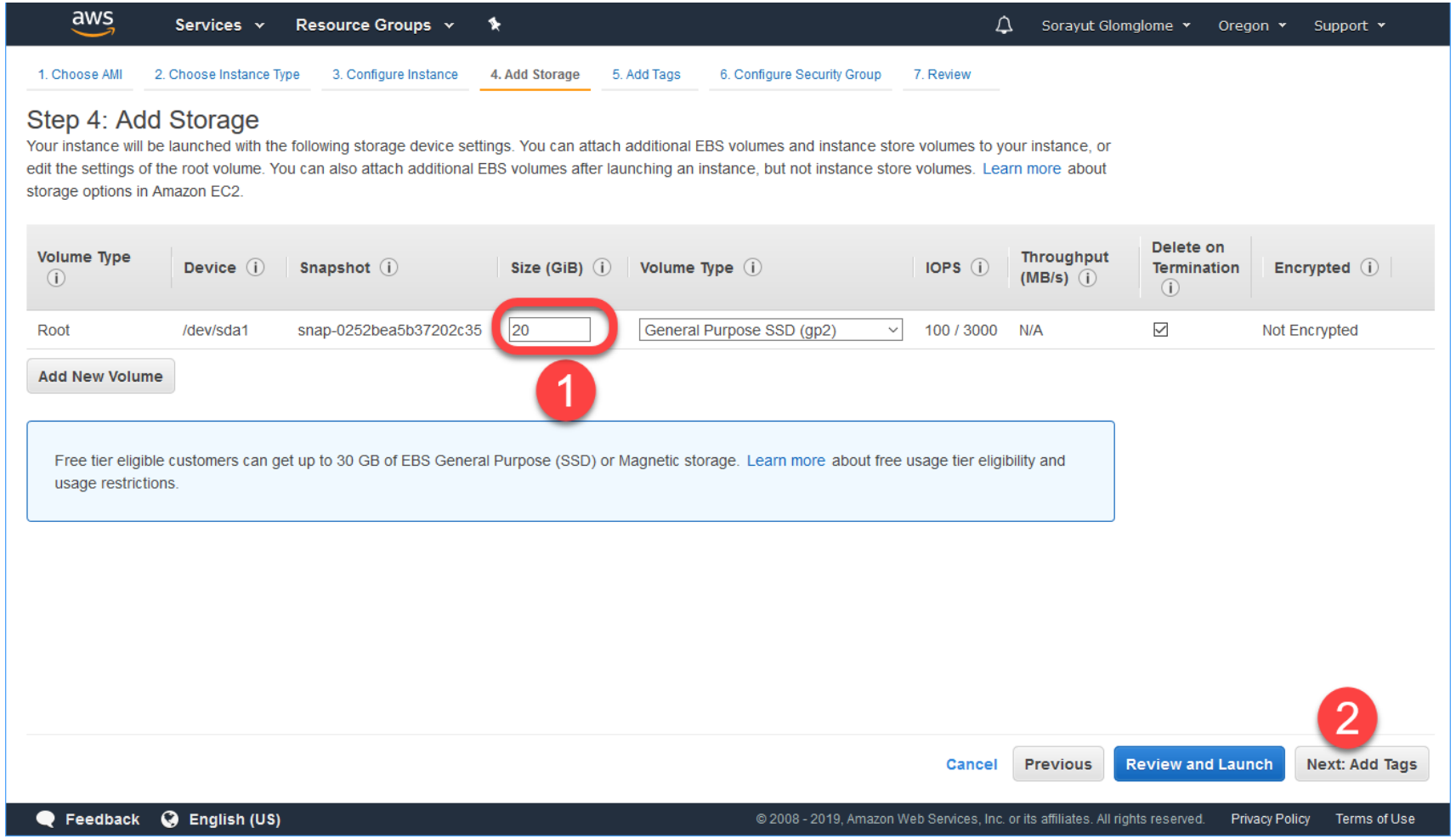
Region: US West (Oregon)

	vCPU	ECU	Memory (GiB)	Instance Storage (GB)	Linux/UNIX Usage
General Purpose - Current Generation					
a1.medium	1	N/A	2 GiB	EBS Only	$0.0255 per Hour
a1.large	2	N/A	4 GiB	EBS Only	$0.051 per Hour
a1.xlarge	4	N/A	8 GiB	EBS Only	$0.102 per Hour
a1.2xlarge	8	N/A	16 GiB	EBS Only	$0.204 per Hour
a1.4xlarge	16	N/A	32 GiB	EBS Only	$0.408 per Hour
t3.nano	2	Variable	0.5 GiB	EBS Only	$0.0052 per Hour
t3.micro	2	Variable	1 GiB	EBS Only	$0.0104 per Hour
t3.small	2	Variable	2 GiB	EBS Only	$0.0208 per Hour
t3.medium	2	Variable	4 GiB	EBS Only	$0.0416 per Hour
t3.large	2	Variable	8 GiB	EBS Only	$0.0832 per Hour
t3.large	2	Variable	8 GiB	EBS Only	$0.0832 per Hour
t3.xlarge	4	Variable	16 GiB	EBS Only	$0.1664 per Hour
t3.2xlarge	8	Variable	32 GiB	EBS Only	$0.3328 per Hour
t2.nano	1	Variable	0.5 GiB	EBS Only	$0.0058 per Hour
t2.micro	1	Variable	1 GiB	EBS Only	$0.0116 per Hour
t2.small	1	Variable	2 GiB	EBS Only	$0.023 per Hour
t2.medium	2	Variable	4 GiB	EBS Only	$0.0464 per Hour
t2.large	2	Variable	8 GiB	EBS Only	$0.0928 per Hour
t2.xlarge	4	Variable	16 GiB	EBS Only	$0.1856 per Hour
t2.2xlarge	8	Variable	32 GiB	EBS Only	$0.3712 per Hour
m5.large	2	8	8 GiB	EBS Only	$0.096 per Hour
m5.xlarge	4	16	16 GiB	EBS Only	$0.192 per Hour
m5.2xlarge	8	31	32 GiB	EBS Only	$0.384 per Hour
m5.4xlarge	16	60	64 GiB	EBS Only	$0.768 per Hour
m5.12xlarge	48	173	192 GiB	EBS Only	$2.304 per Hour
m5.24xlarge	96	345	384 GiB	EBS Only	$4.608 per Hour
m5a.large	2	N/A	8 GiB	EBS Only	$0.086 per Hour
m5a.xlarge	4	N/A	16 GiB	EBS Only	$0.172 per Hour

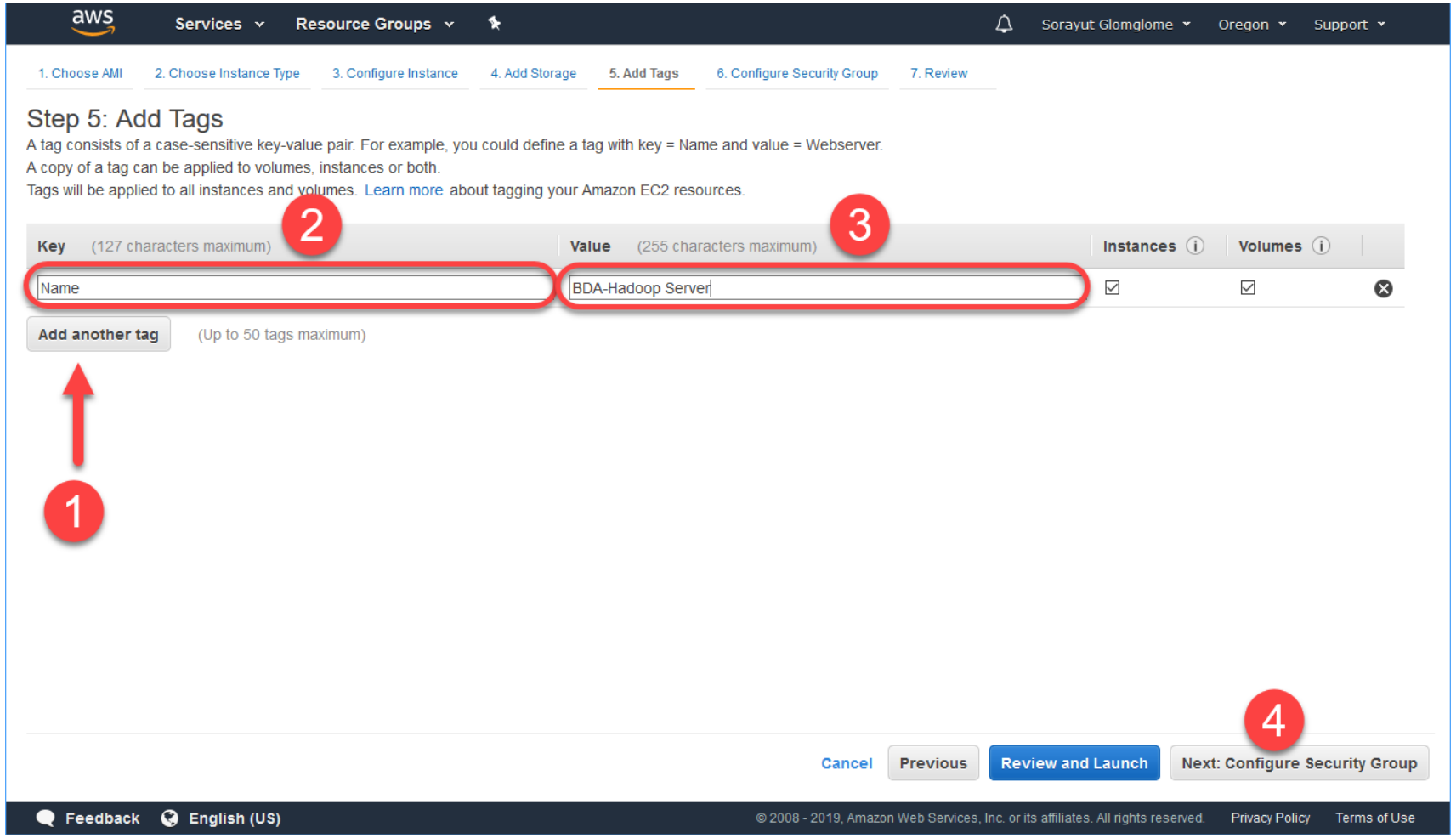
Leave configuration as default



Add Storage : 20 GiB

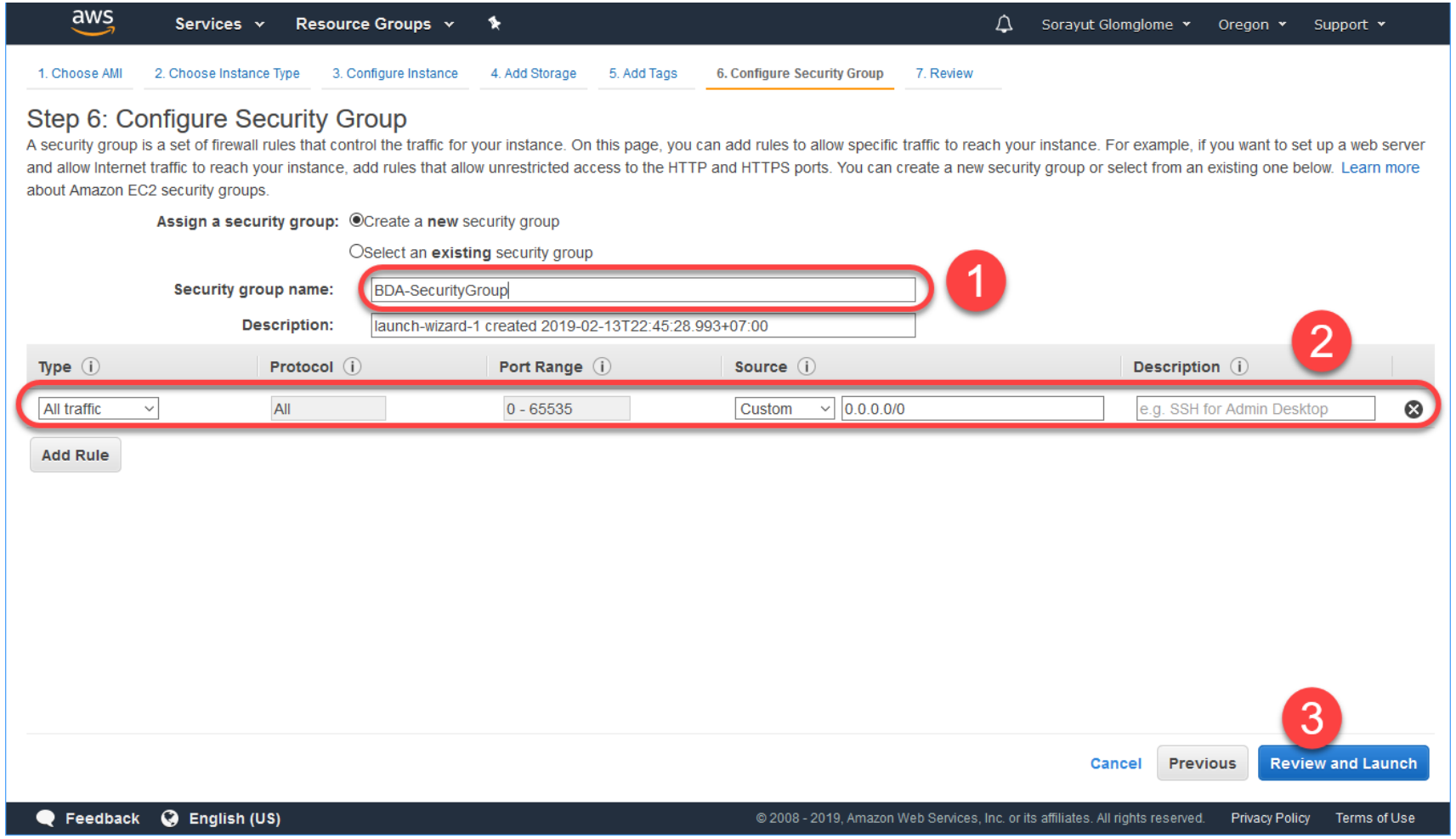


Name your server

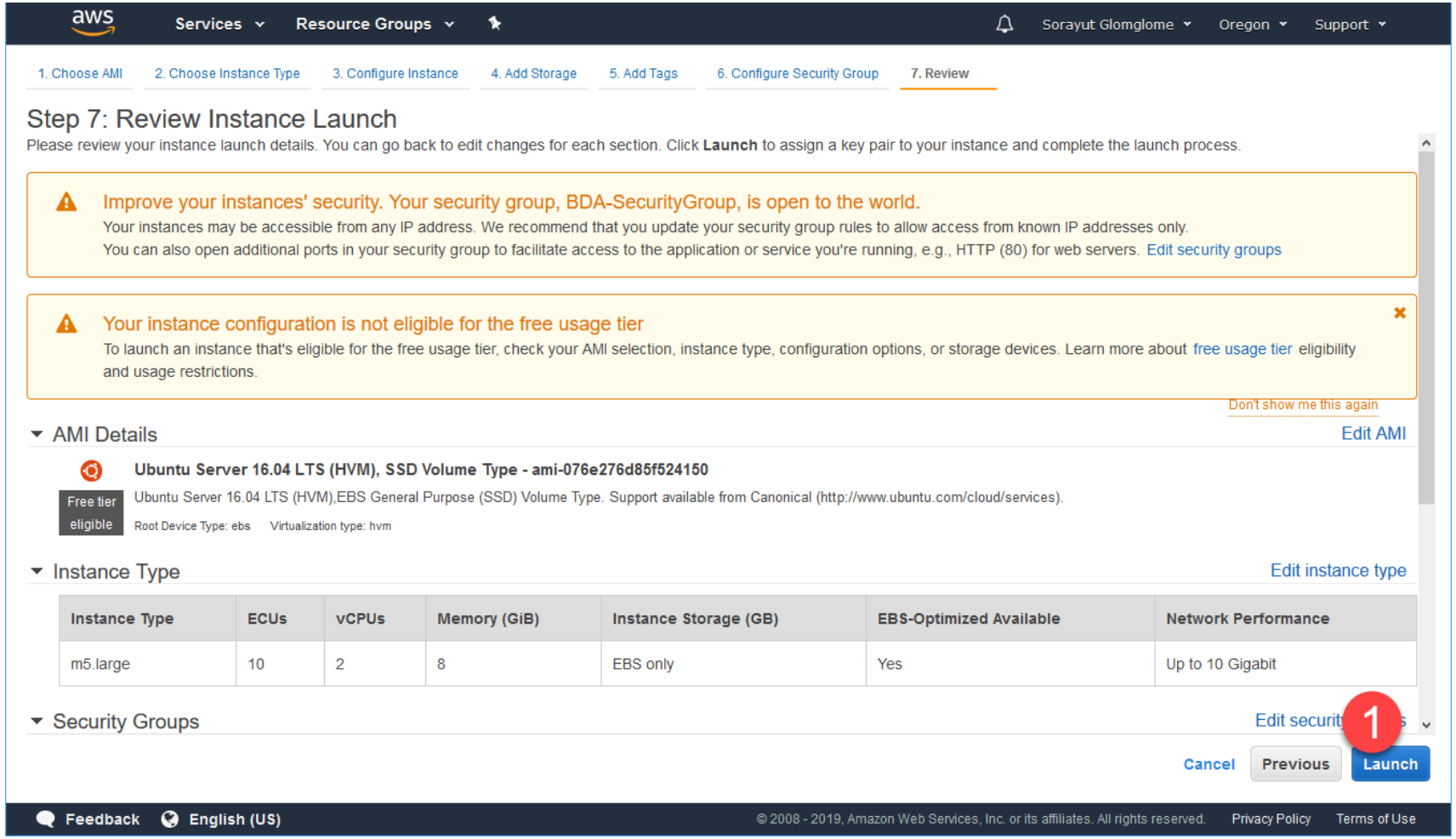


Create and name a new security group

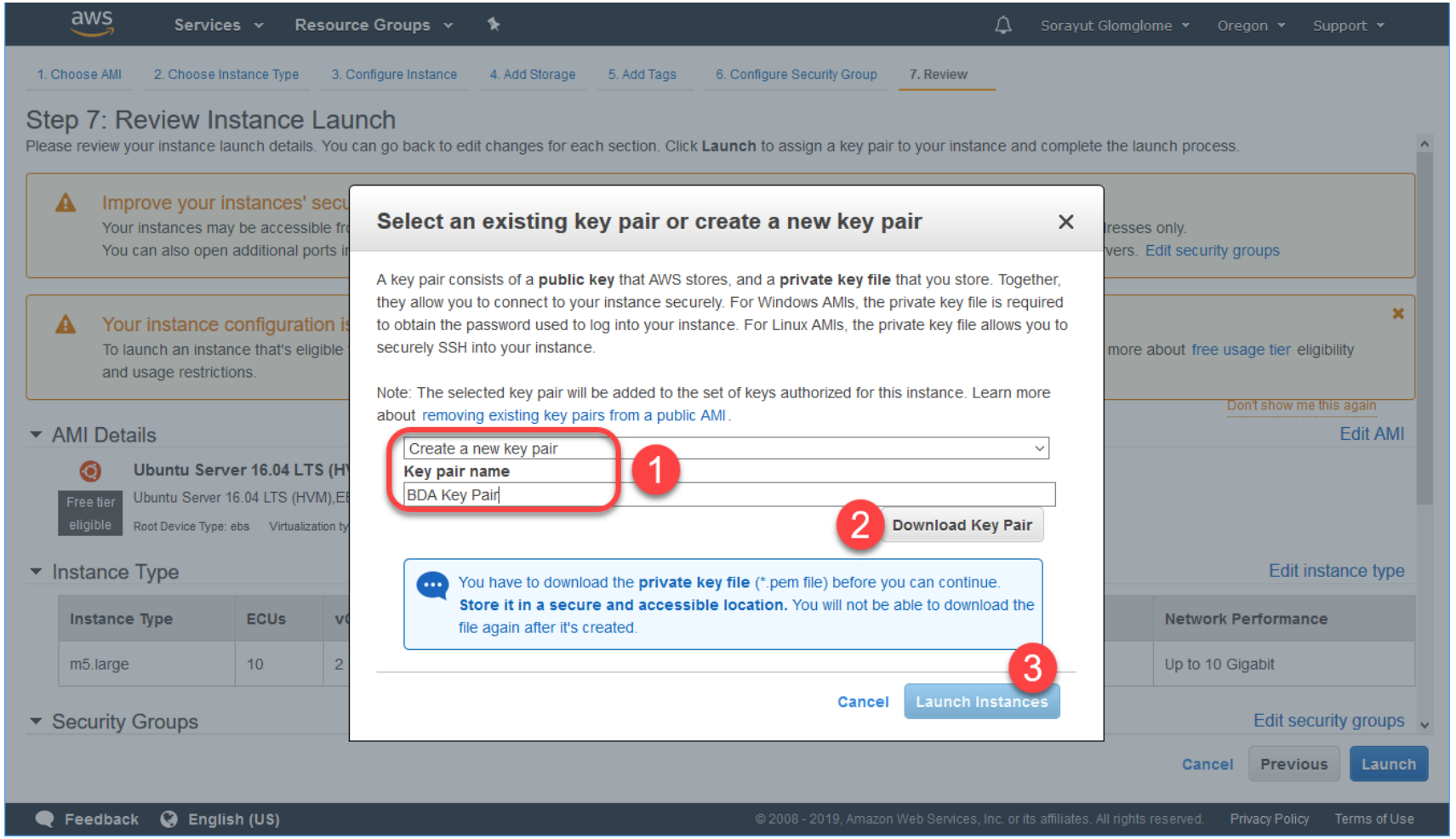
Allow all traffic because of the convenience



Review and then launch the instance



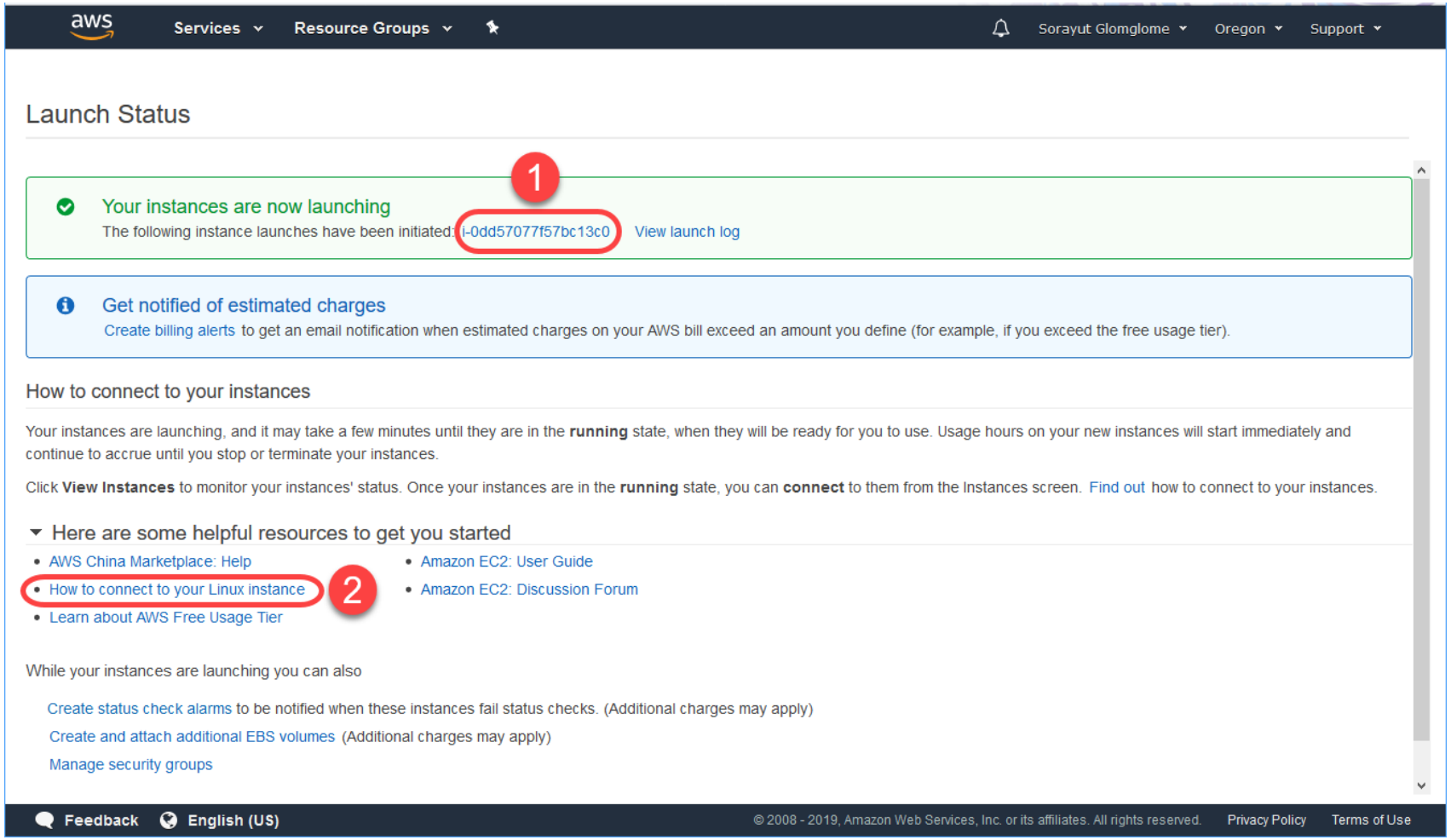
Name a new key pair for authentication
Download to a save location
IMPORTANT Do not lose the key file



Launch Status

Click 1 to check the instance

Click 2 to read how to connect to Linux instance

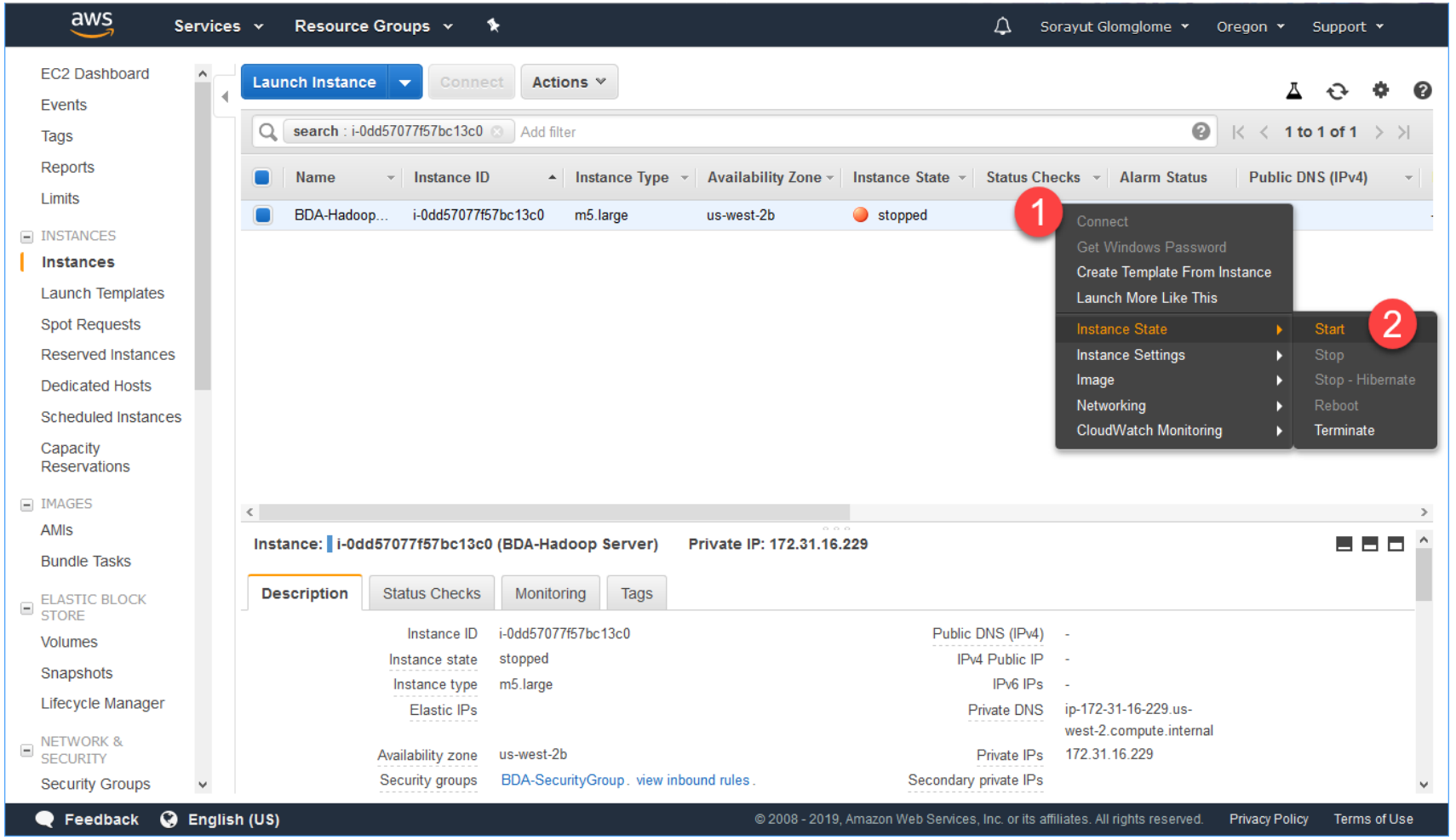


Connect to Your Linux instance

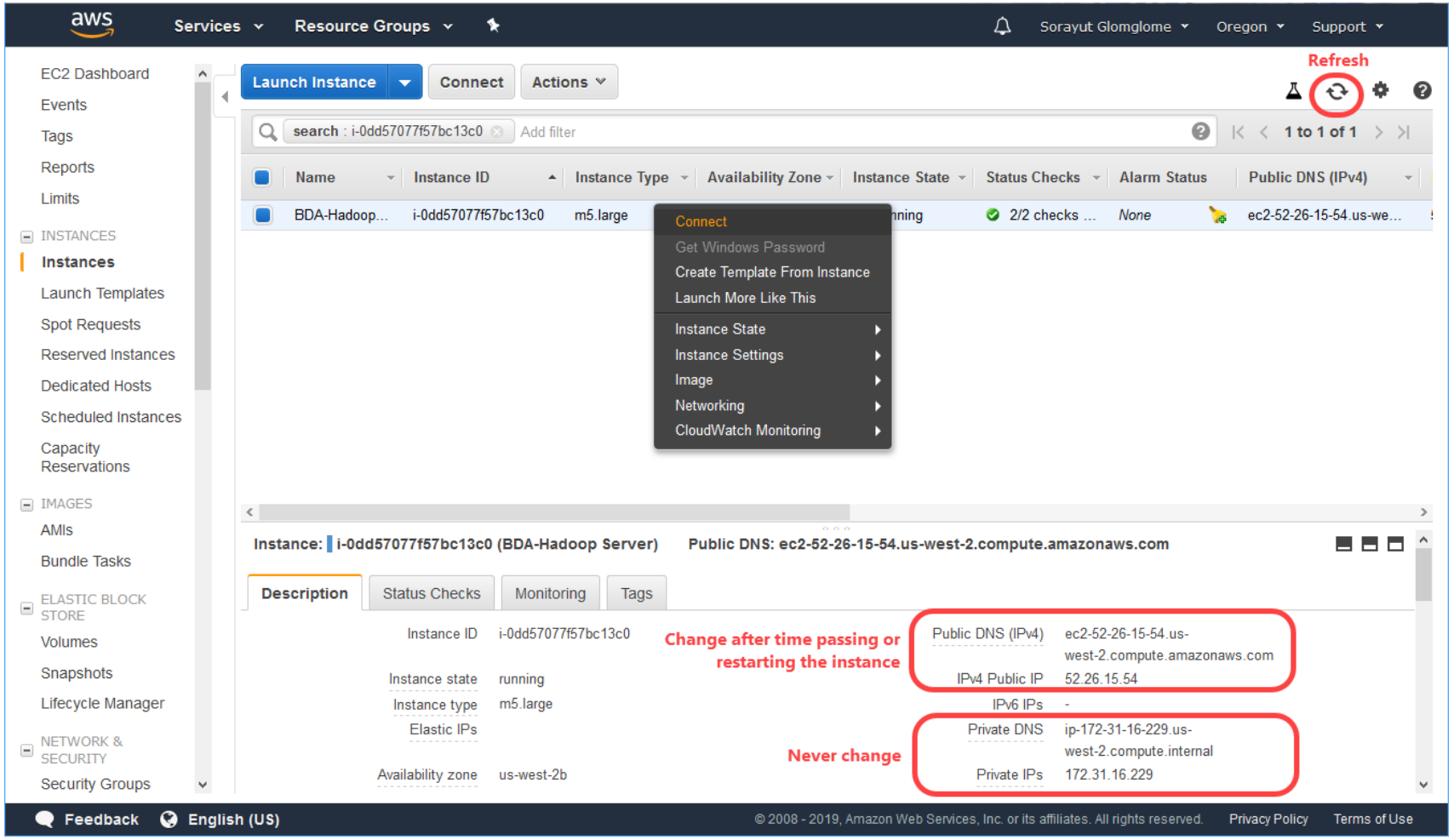
- https://goo.gl/wdSfSr

Your Computer	Topic
Linux or Mac OS X	Connecting to Your Linux Instance Using SSH
Windows	Connecting to Your Linux Instance from Windows Using PuTTY Connecting to Your Linux Instance from Windows Using Windows Subsystem for Linux Connecting to Your Linux Instance Using SSH
All	Connecting to Your Linux Instance Using MindTerm

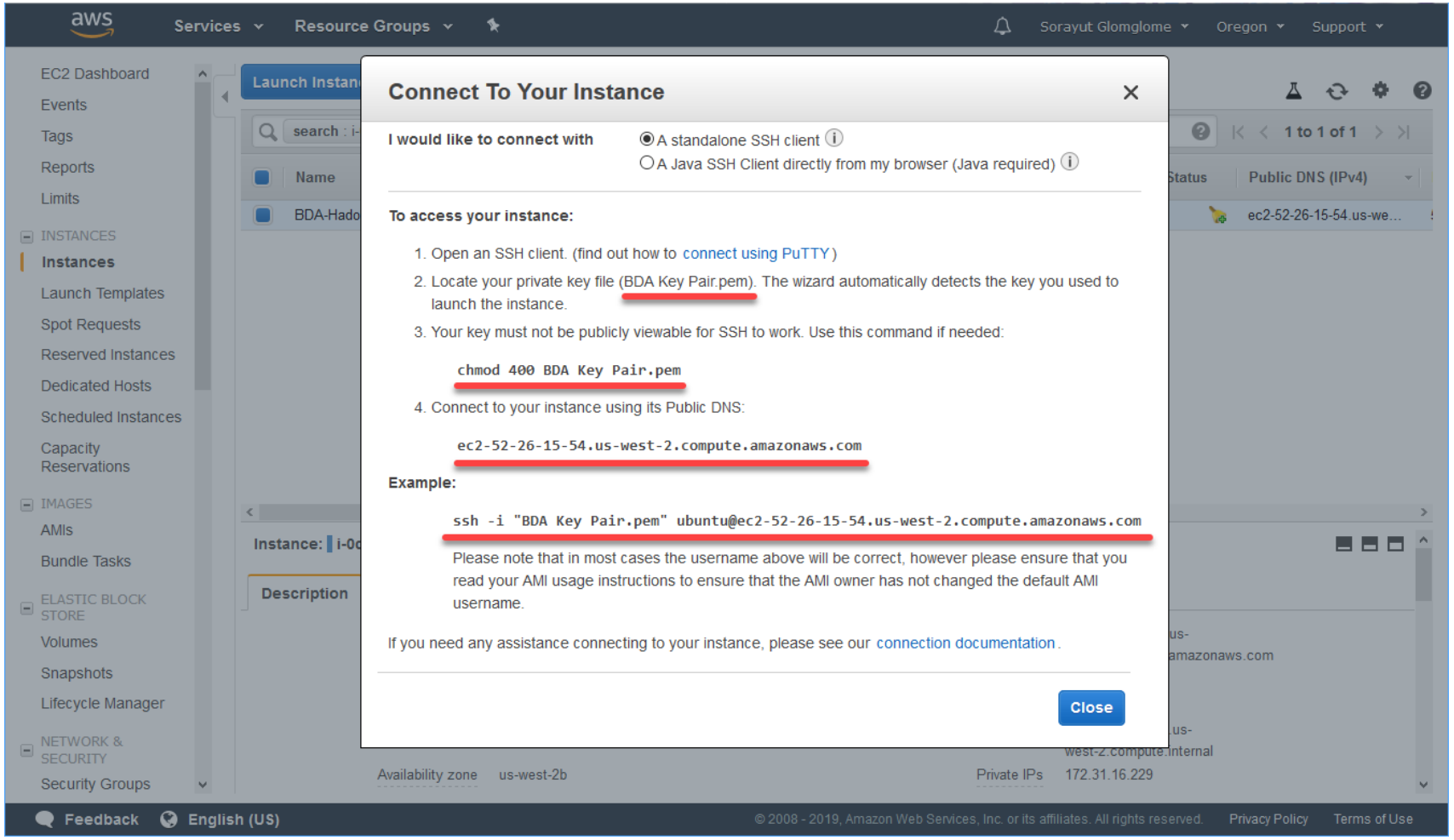
Start the instance:
Right click on the row of instance
Select Instance State -> Start



Right click and select connect to display How to connect to instance

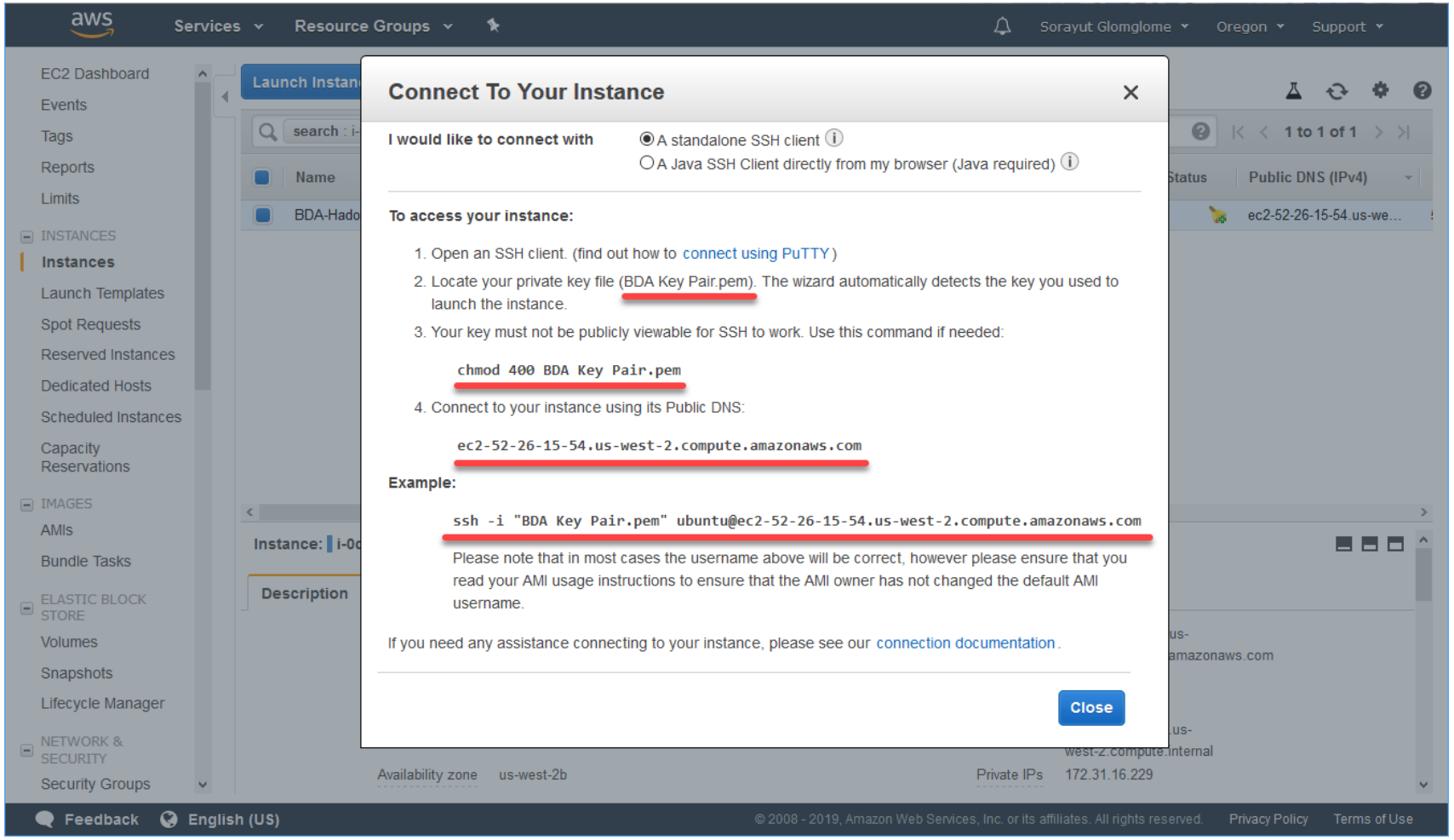


Connect to instance using SSH

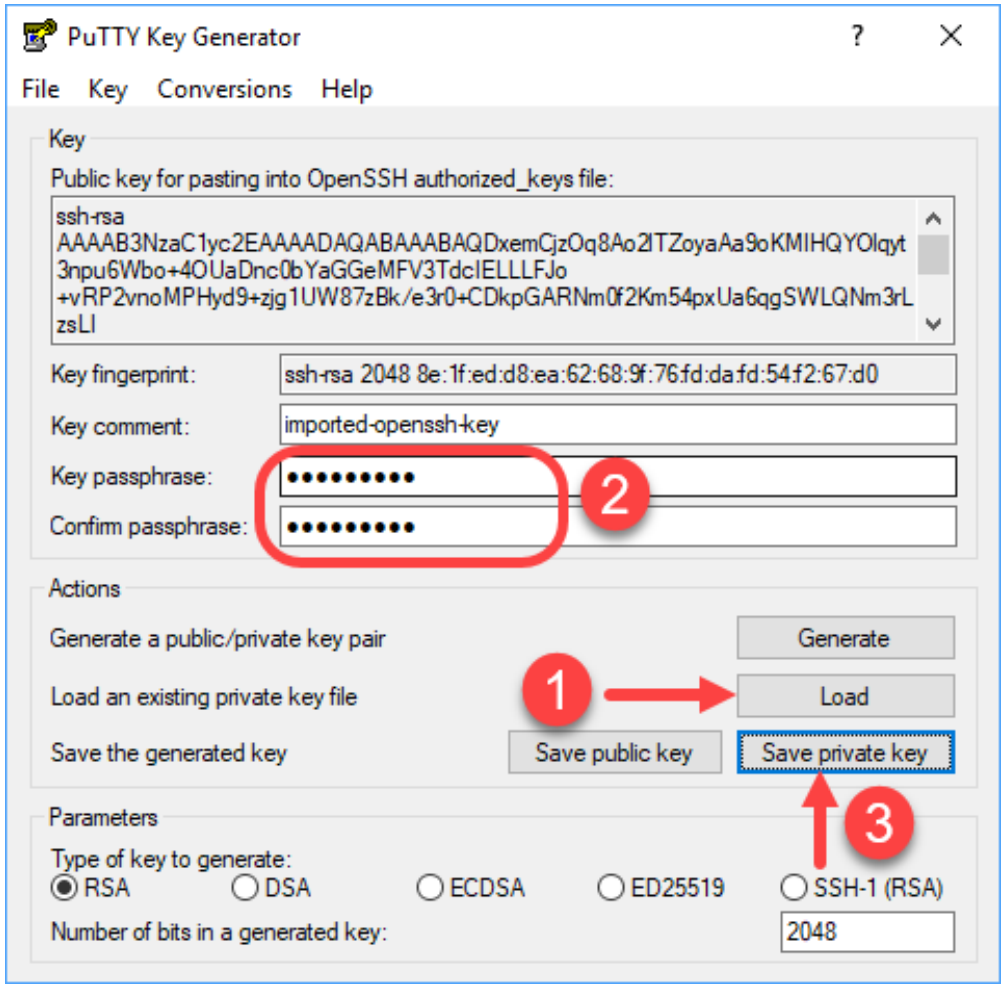


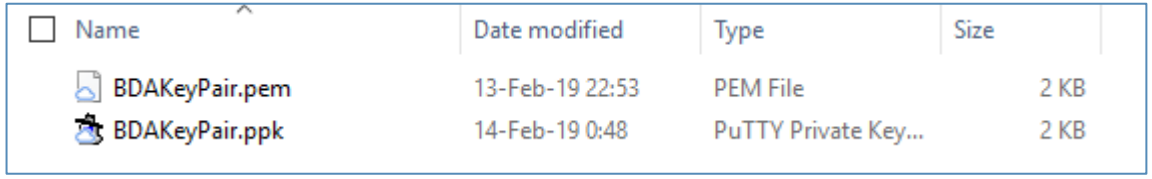
Install PuTTY and PuTTYGen

https://www.ssh.com/ssh/putty/windows/puttygen



Generate .ppk file from .pem file using PuTTYGen

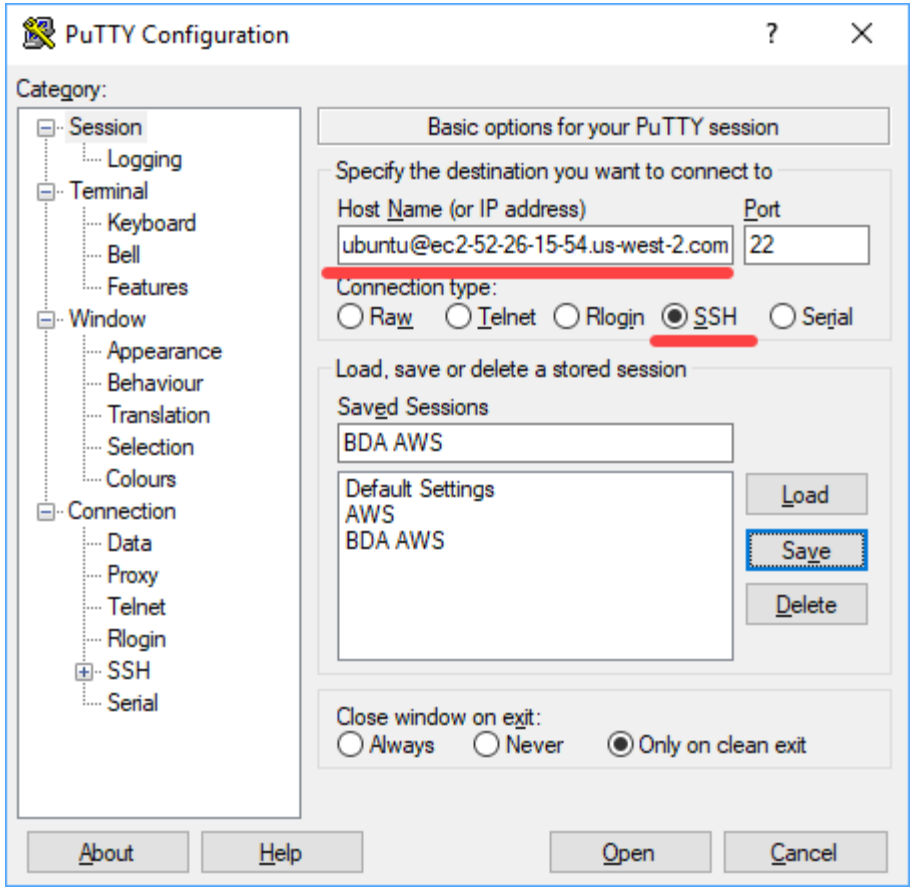


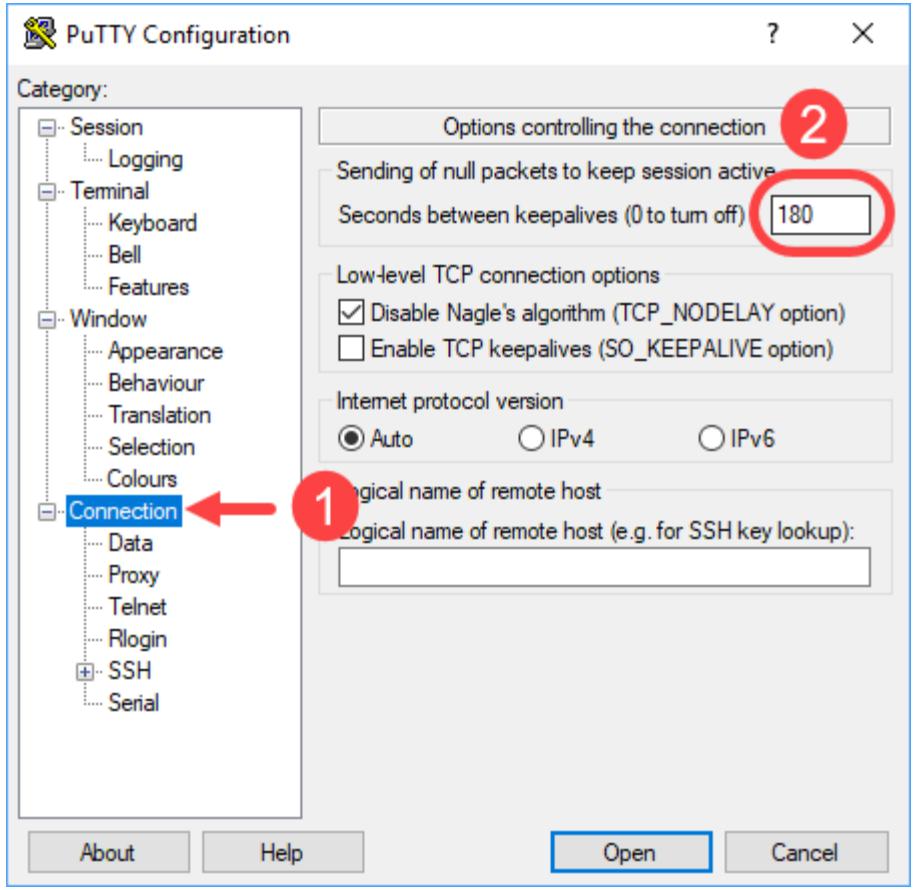


Configure PuTTY

Host name : ubuntu@[PublicDNS]

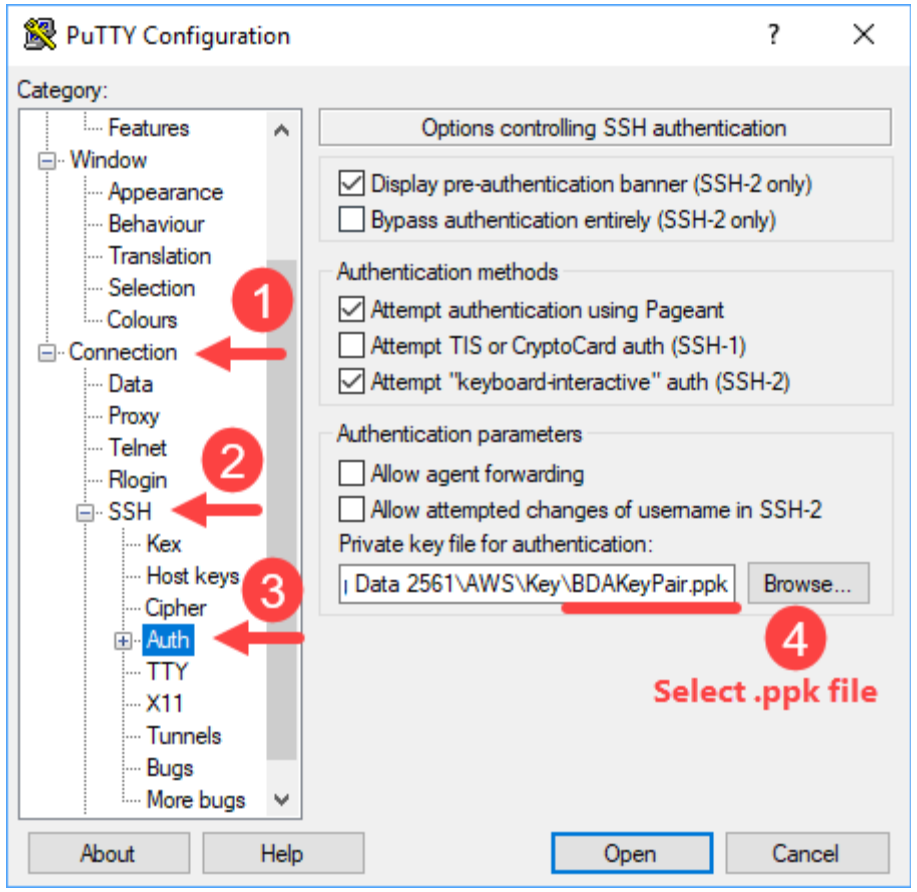
Generate keepalive signal

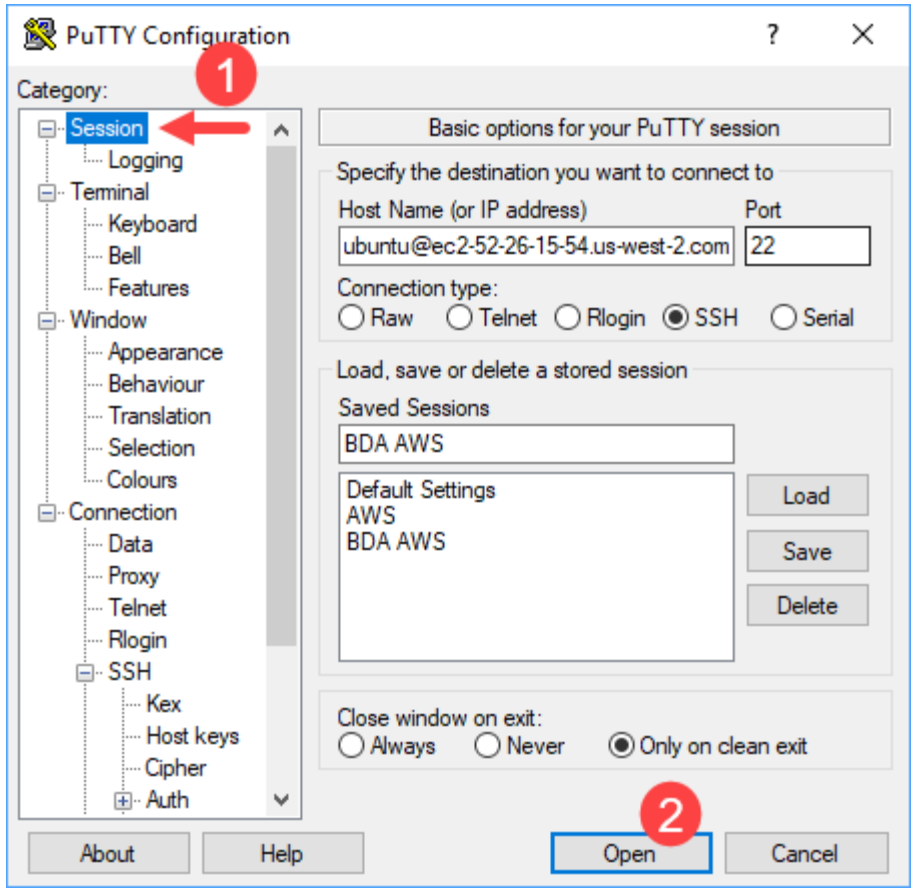




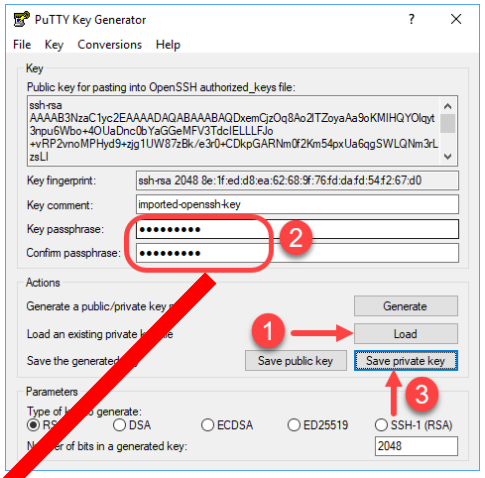
Select .ppk file for authentication

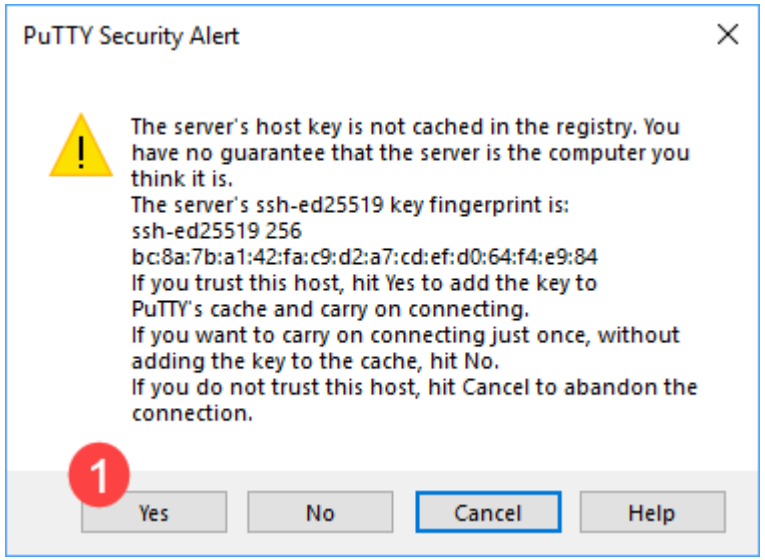
Use private key instead of typing password

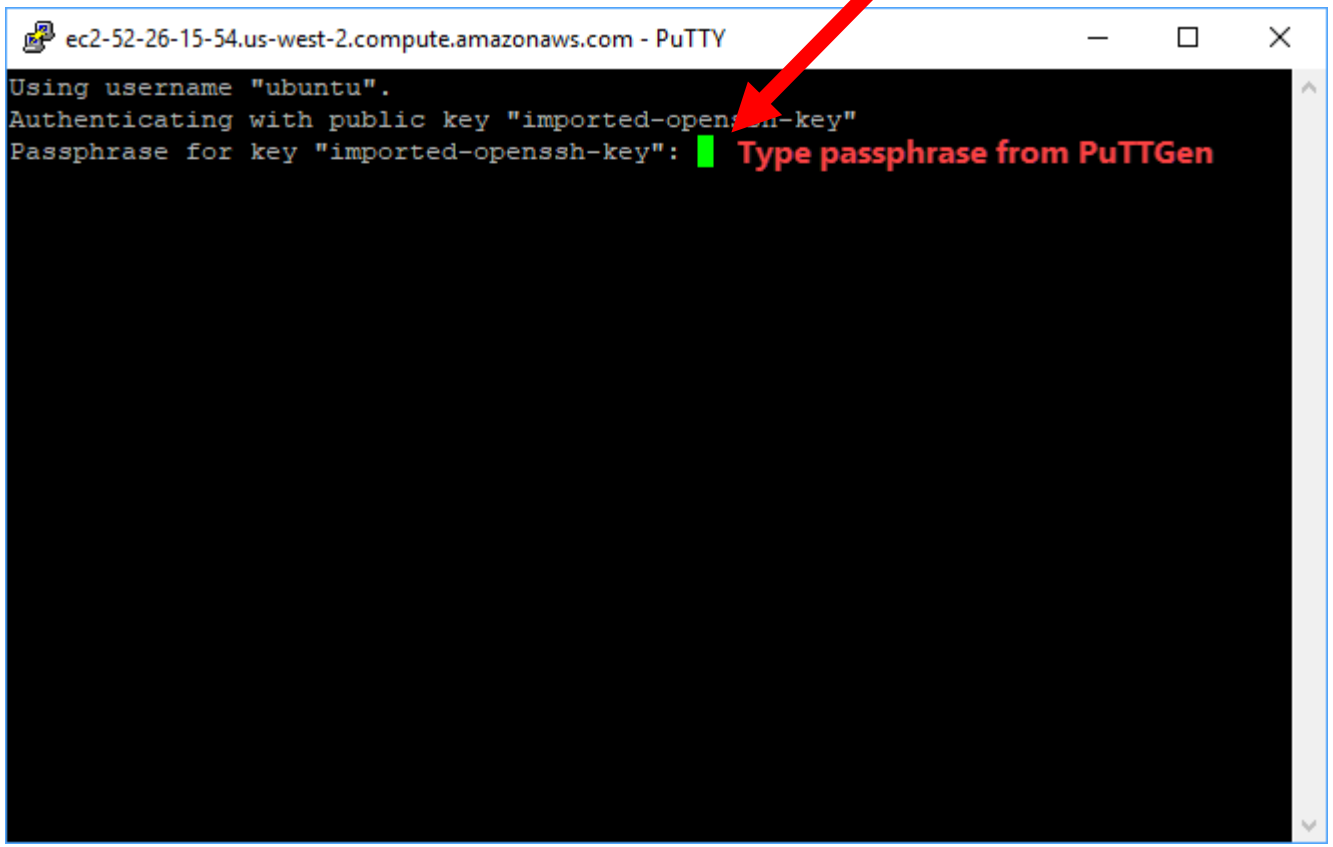




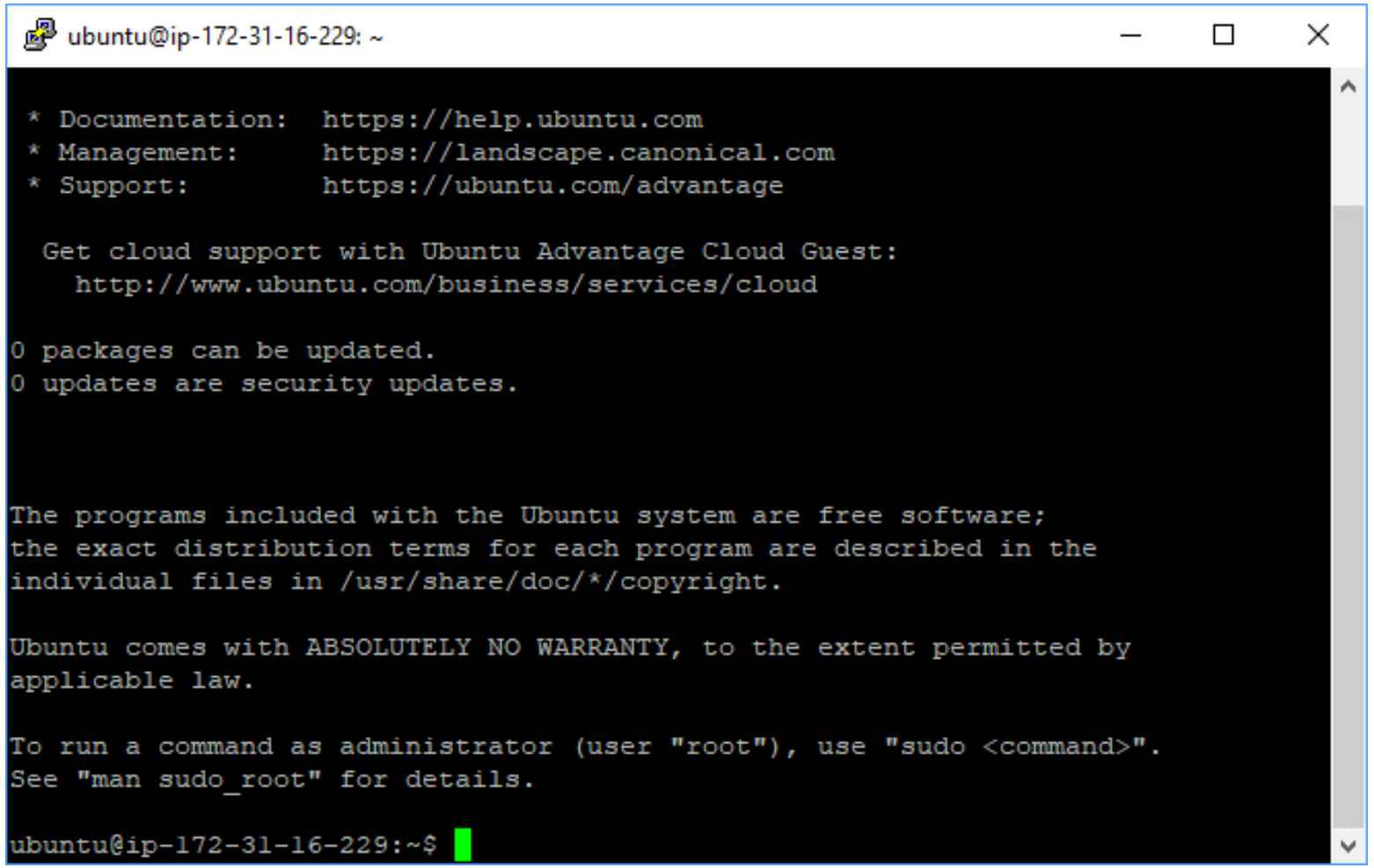
Connect to AWS instance







Command Prompt shows a success connection



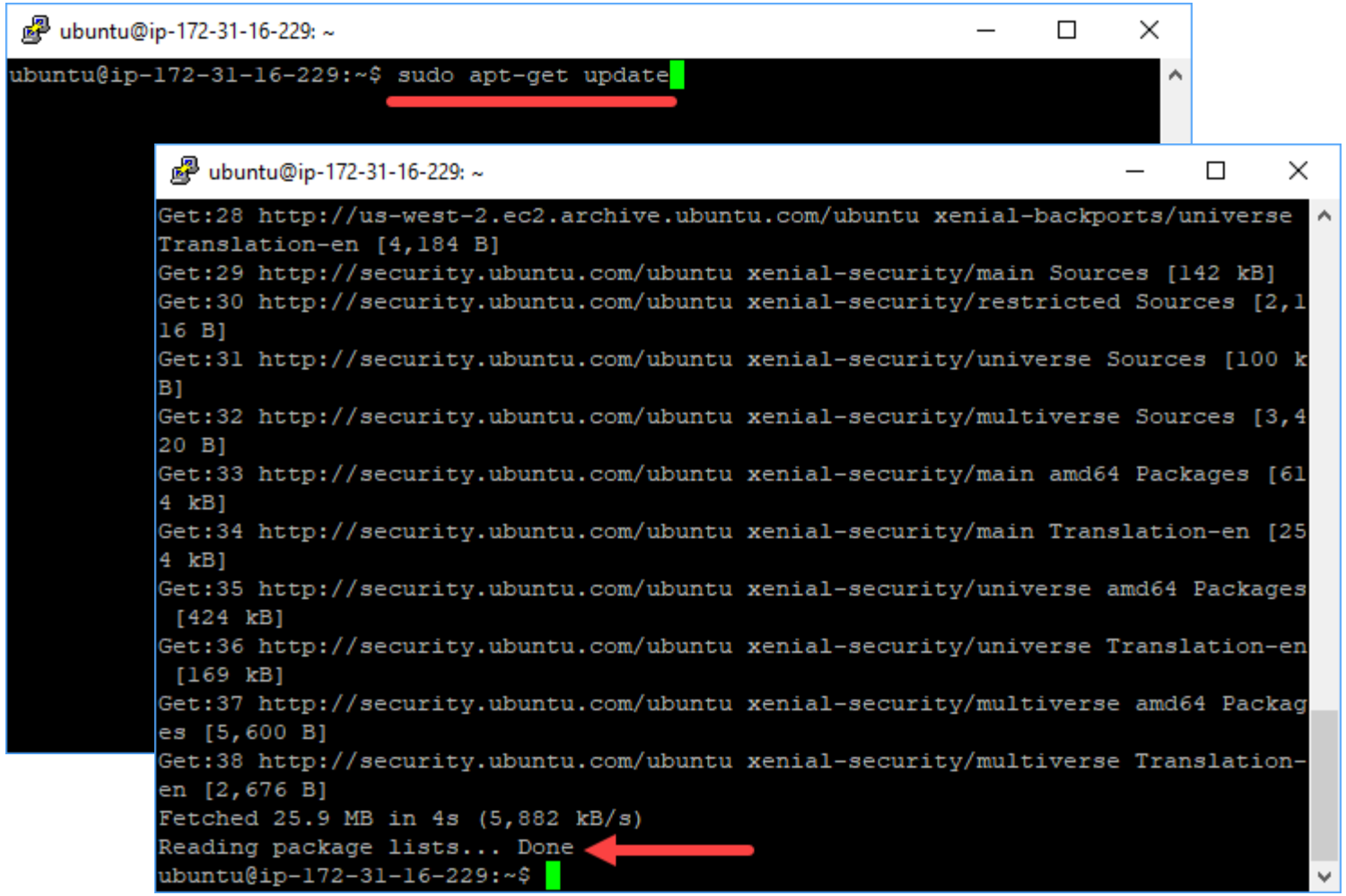
2. Installing Apache Hadoop

Installing Hadoop and Ecosystem

1. Update System Software Repository
2. Configure SSH
3. Install Java
4. Download/Extract Hadoop
5. Install Hadoop
6. Configure Hadoop
7. Format Namenode
8. Start Hadoop
9. Access Hadoop Web Console
10. Import Data to HDFS
    using Hadoop Command Line
11. Stop Hadoop

2.1 Update System Software Repository

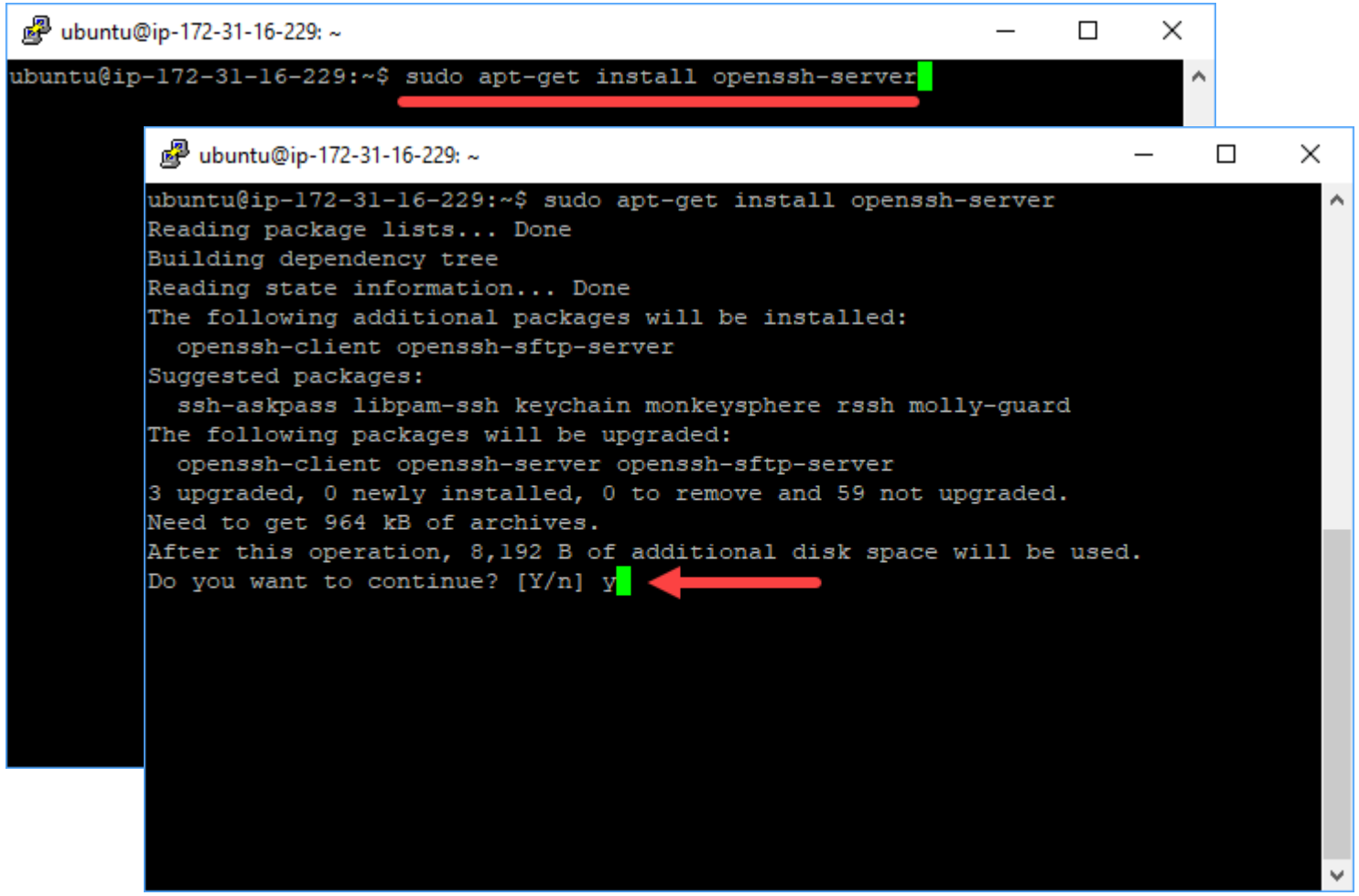
$sudo apt-get update



2.2 Configure SSH

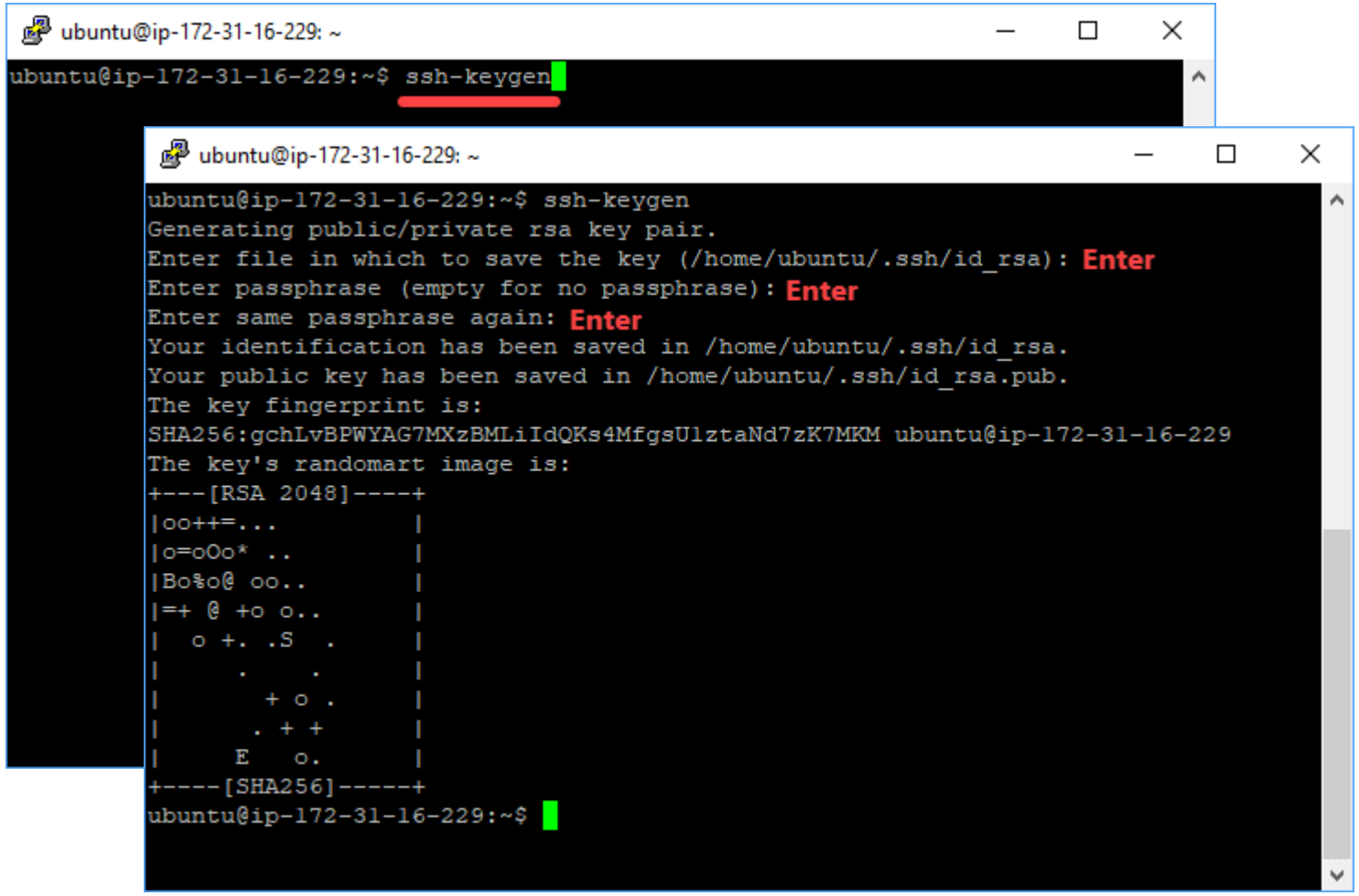
Install SSH:

$sudo apt-get install openssh-server



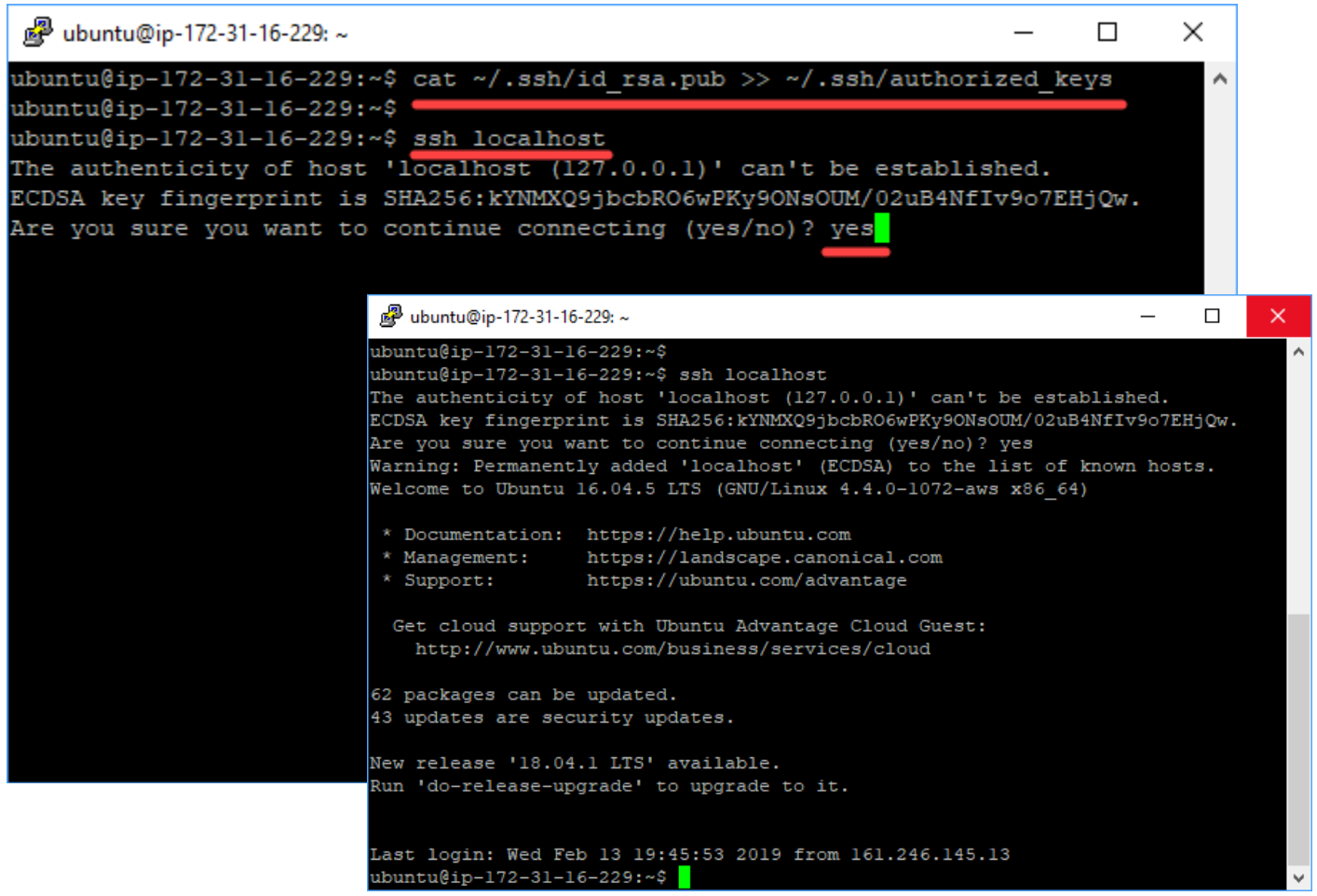
Create Private Key with empty passphrase:

`$ssh-keygen`



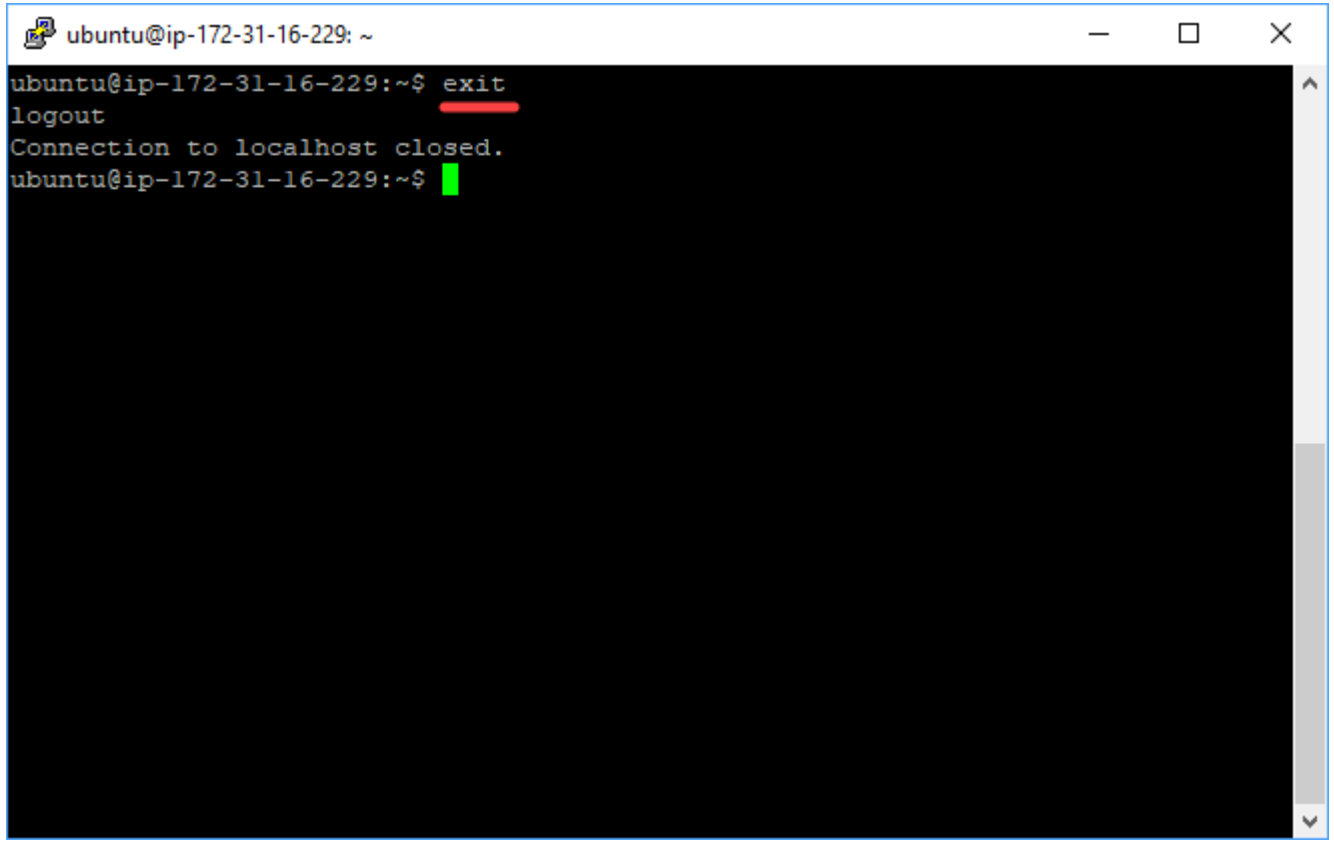
Append and test new key:

```
$cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
$ssh localhost
```



Logout:

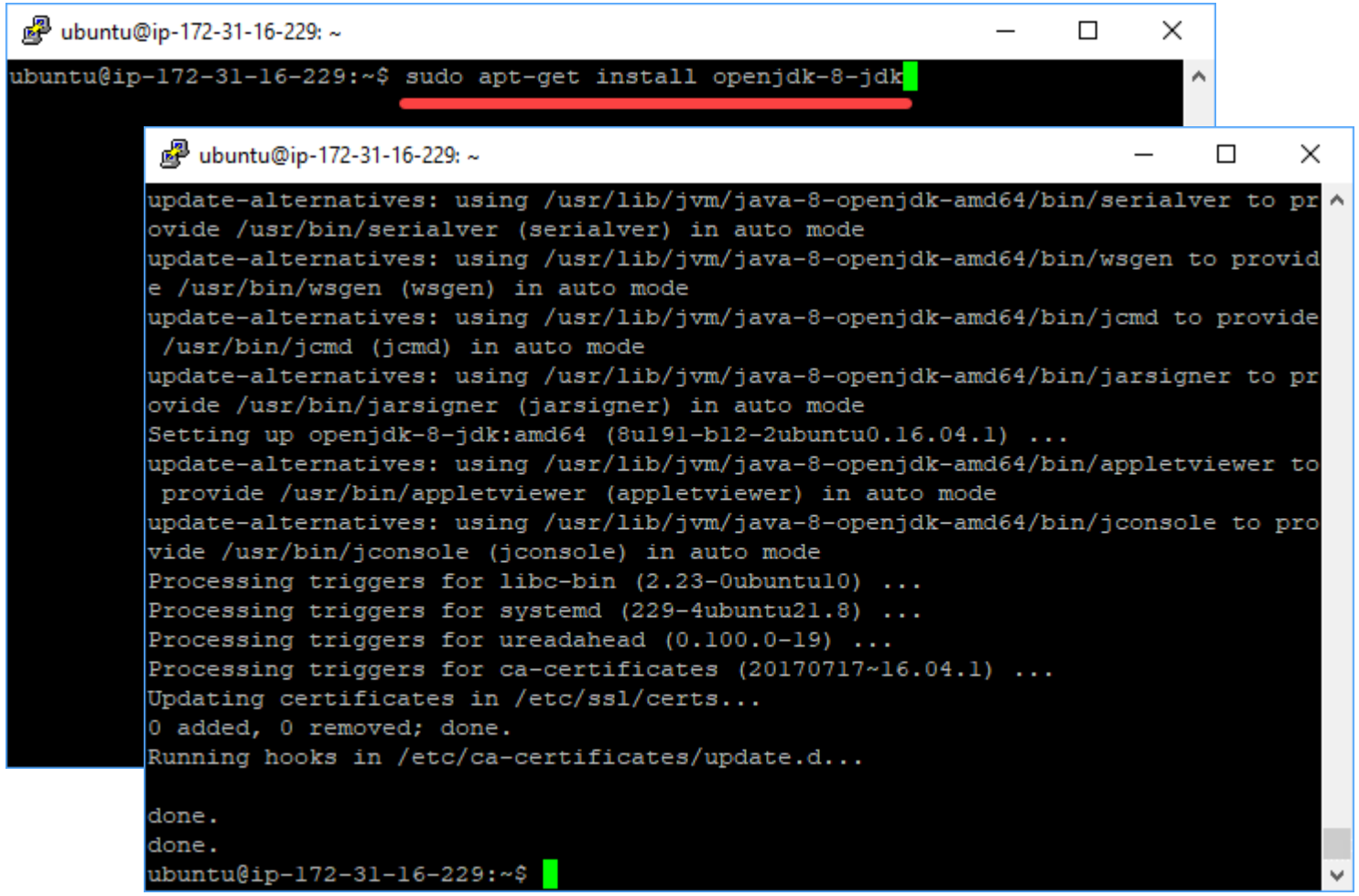
`$exit`


ubuntu@ip-172-31-16-229: ~
ubuntu@ip-172-31-16-229:~$ exit
logout
Connection to localhost closed.
ubuntu@ip-172-31-16-229:~$

2.3 Install Java

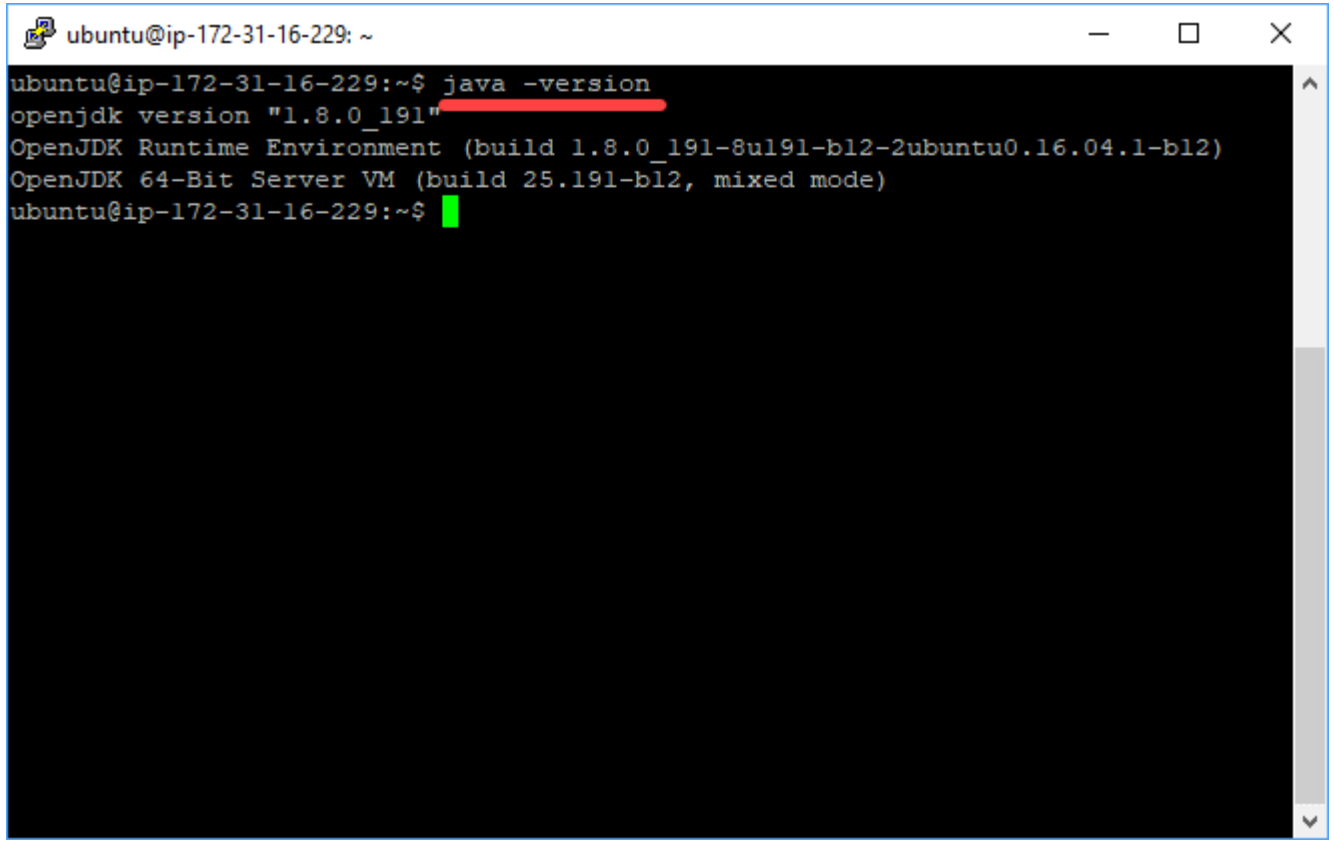
Install Java:

$sudo apt-get install openjdk-8-jdk



Check Java version:

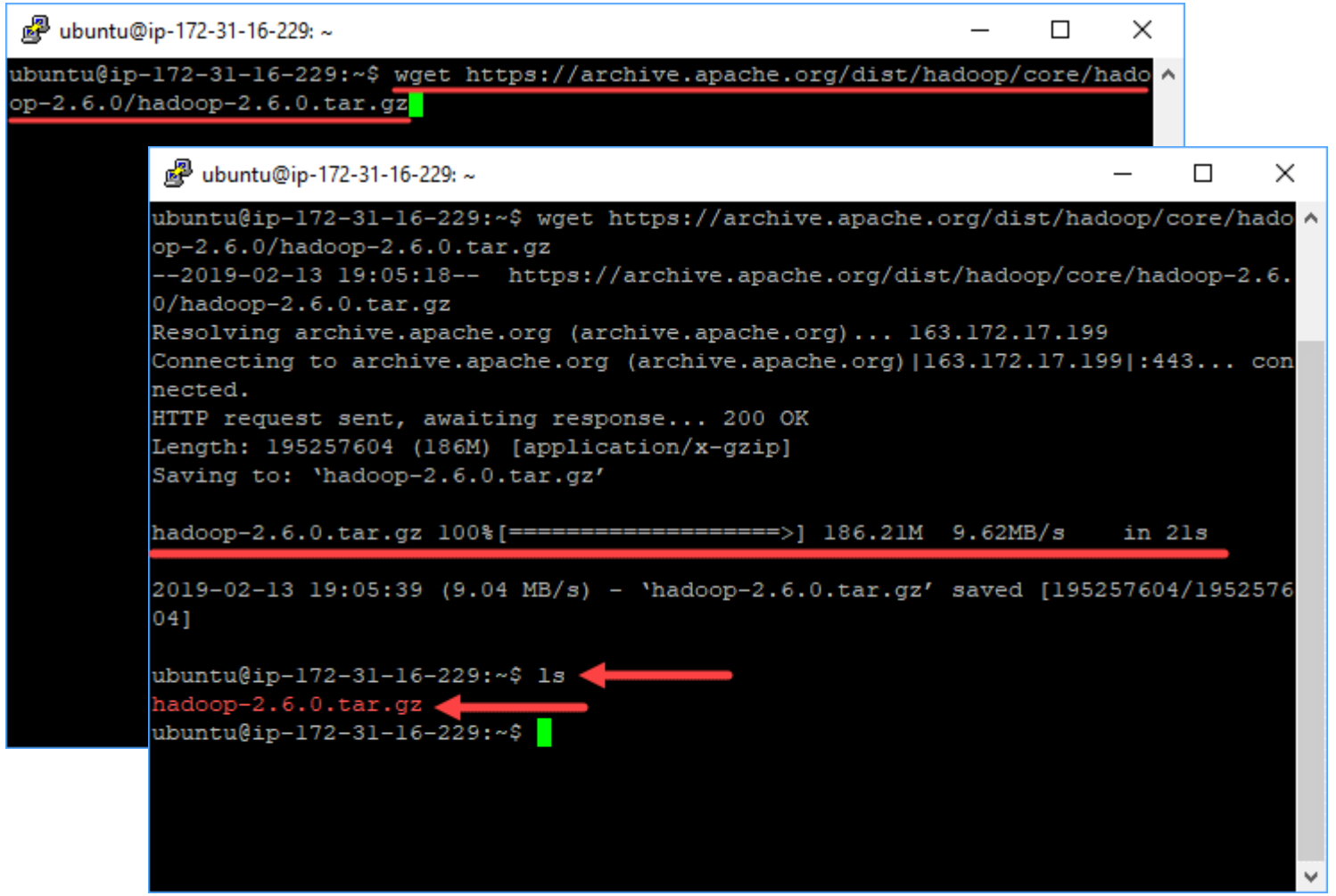
`$java -version`



2.4 Download and Extract Hadoop

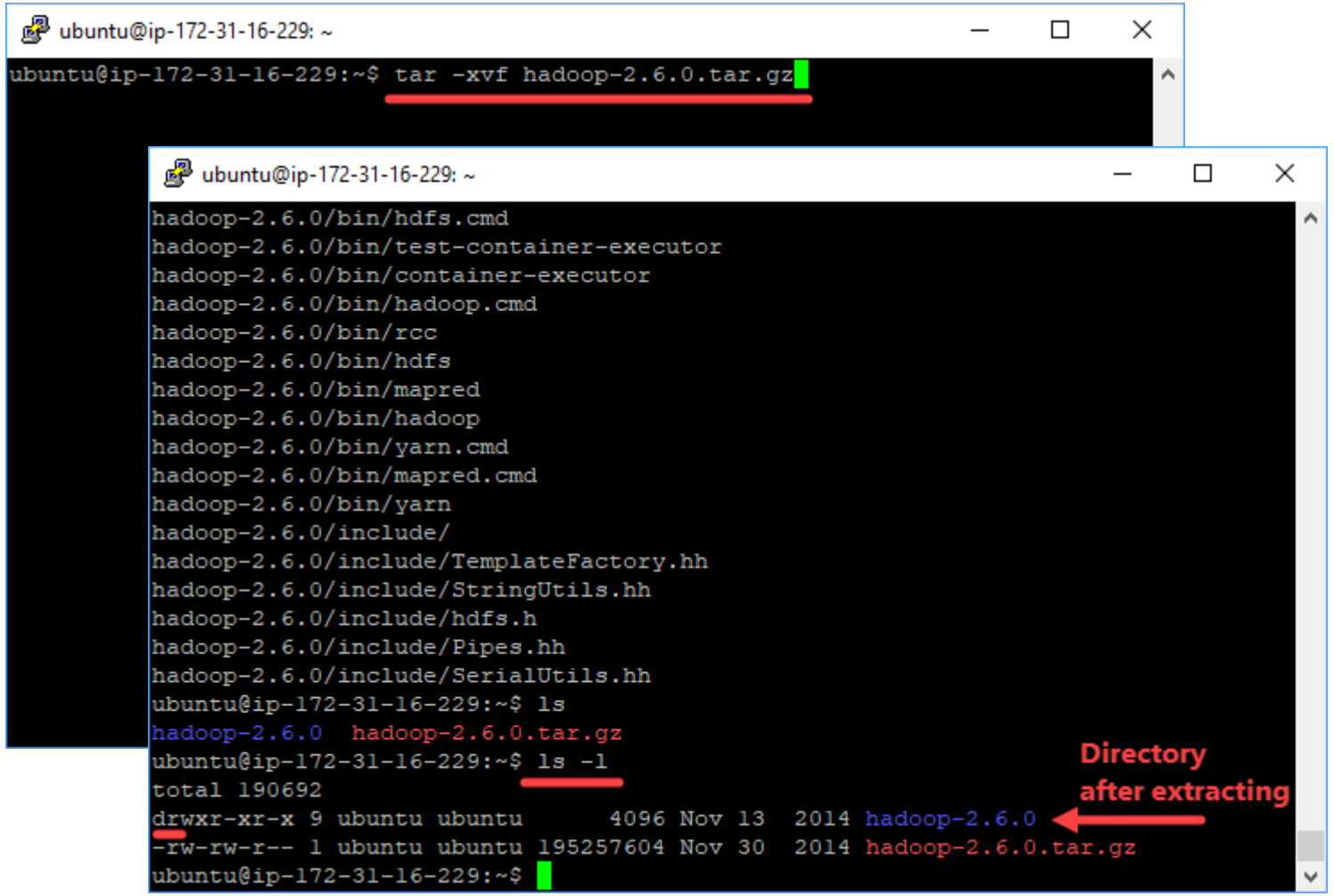
Download Hadoop package:

$wget https://archive.apache.org/dist/hadoop/core/hadoop-2.6.0/hadoop-2.6.0.tar.gz



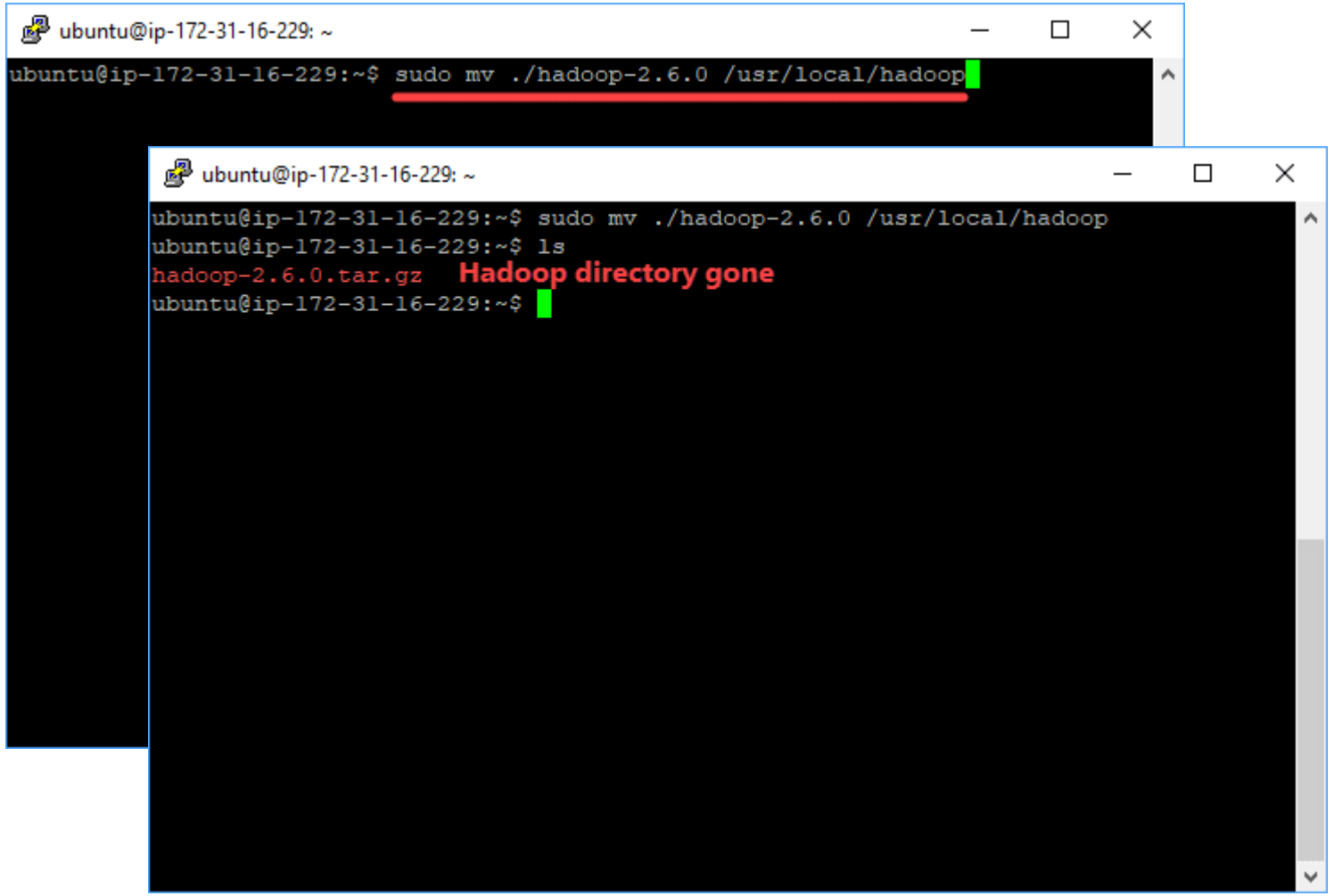
Extract Hadoop package:

`$tar -xvf hadoop-2.6.0.tar.gz`



Move extracted Hadoop directory:

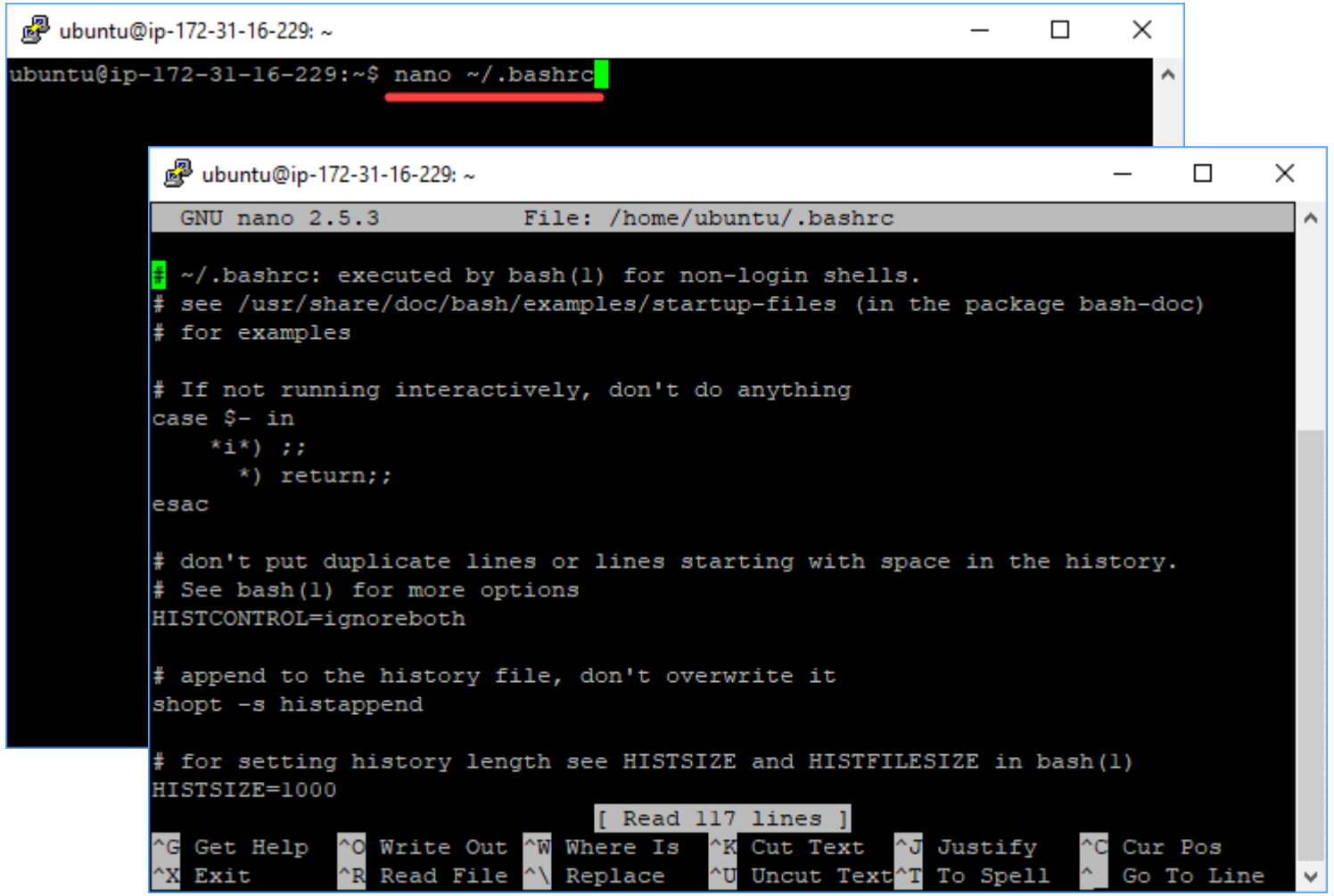
`$sudo mv ./hadoop-2.6.0 /usr/local/hadoop`



2.5 Install Hadoop

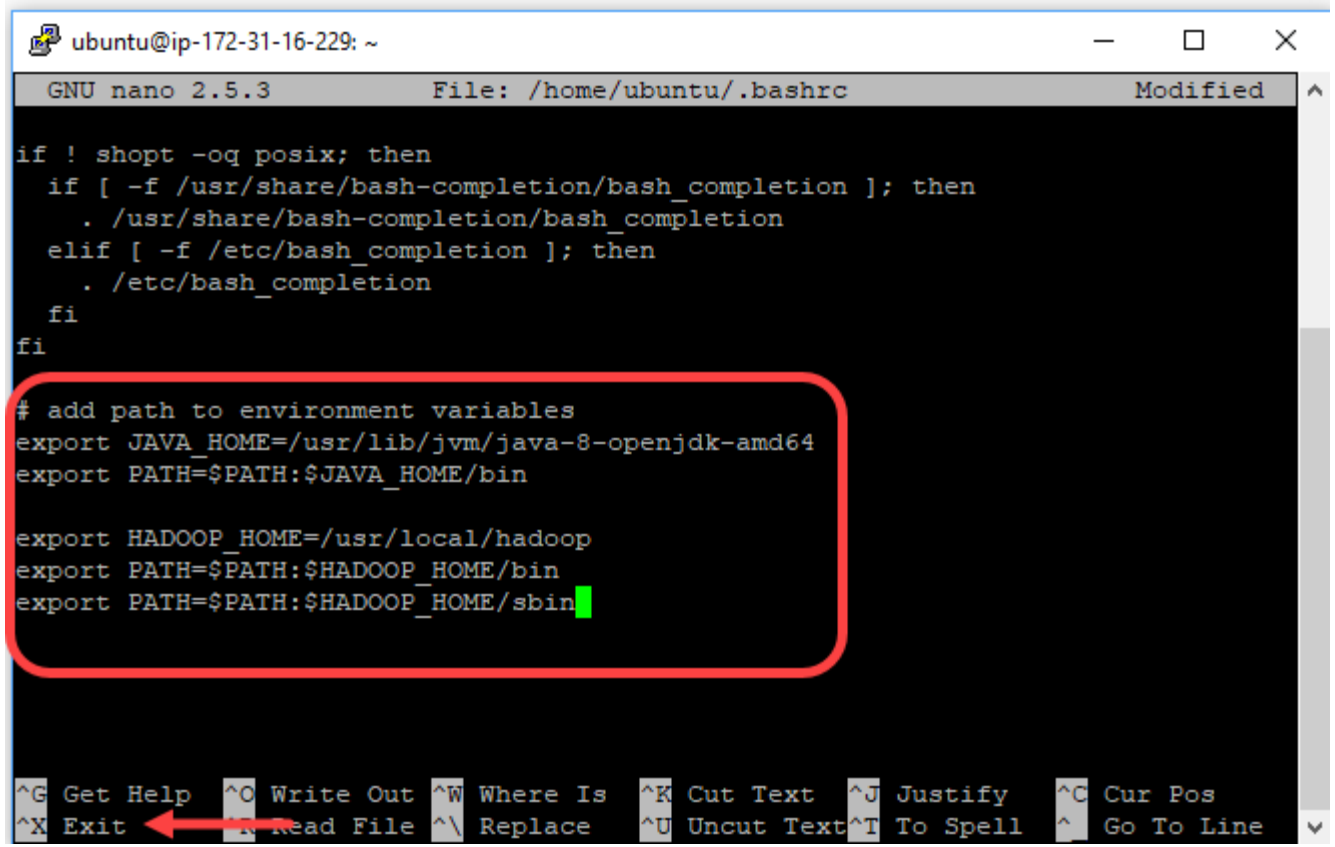
Add path to environment variables:

$nano ~/.bashrc



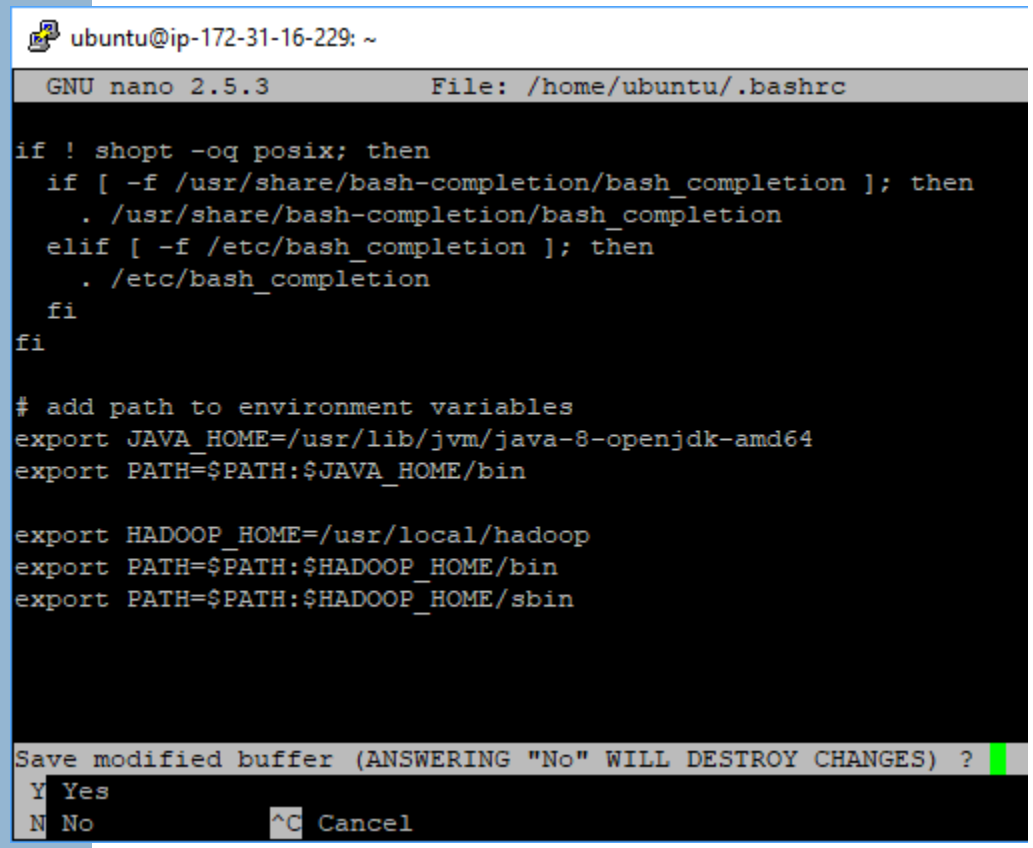
Add these lines at the bottom:

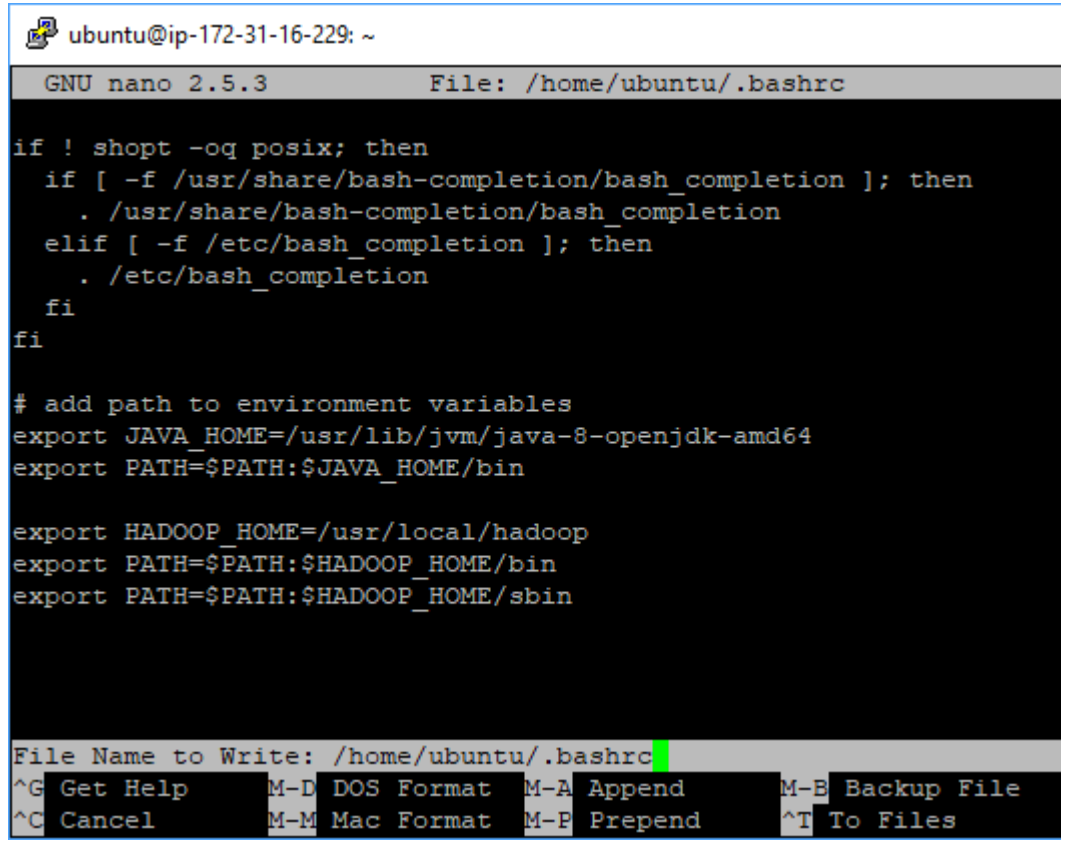
```
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
export PATH=$PATH:$JAVA_HOME/bin

export HADOOP_HOME=/usr/local/hadoop
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin
```



Save file:

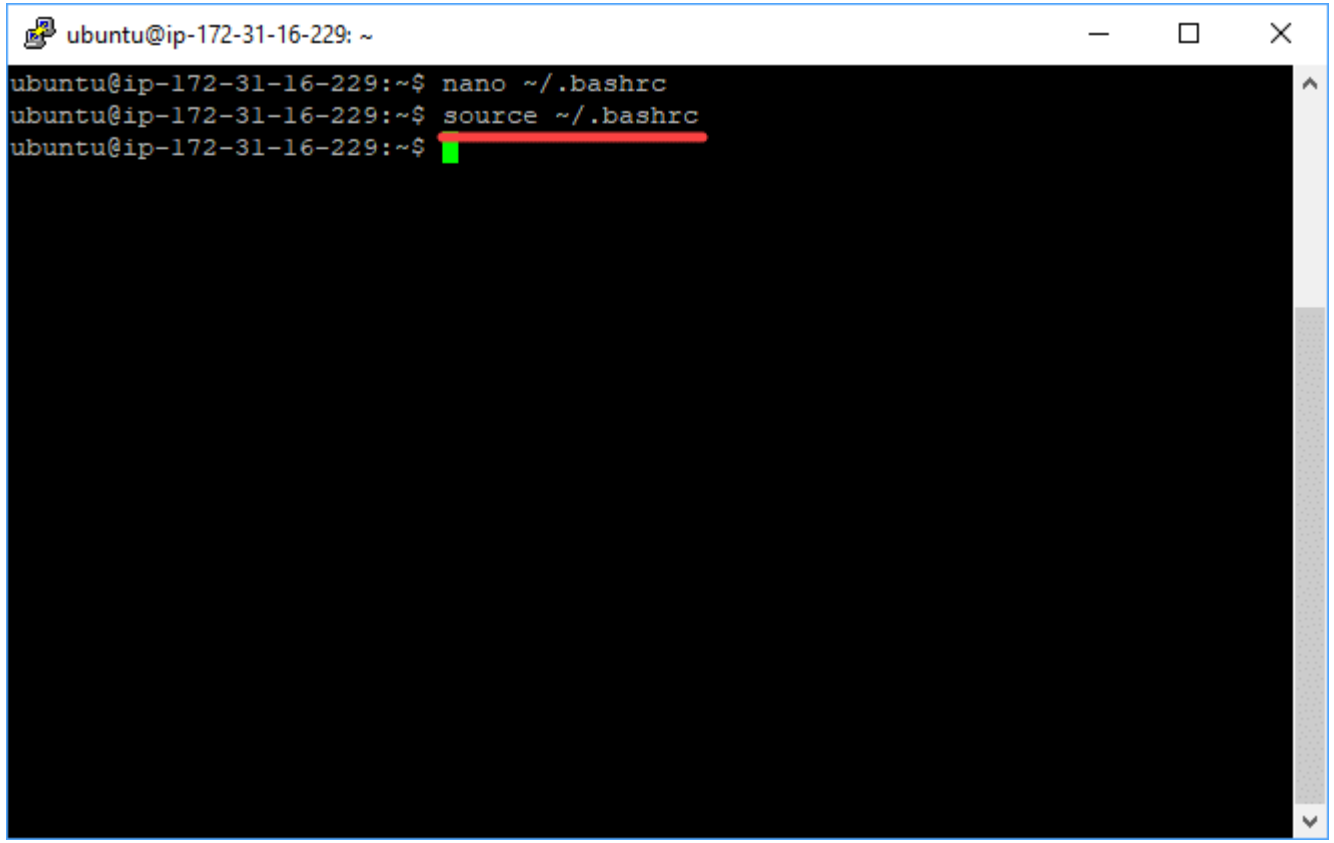
Ctrl+x, press y, and press enter





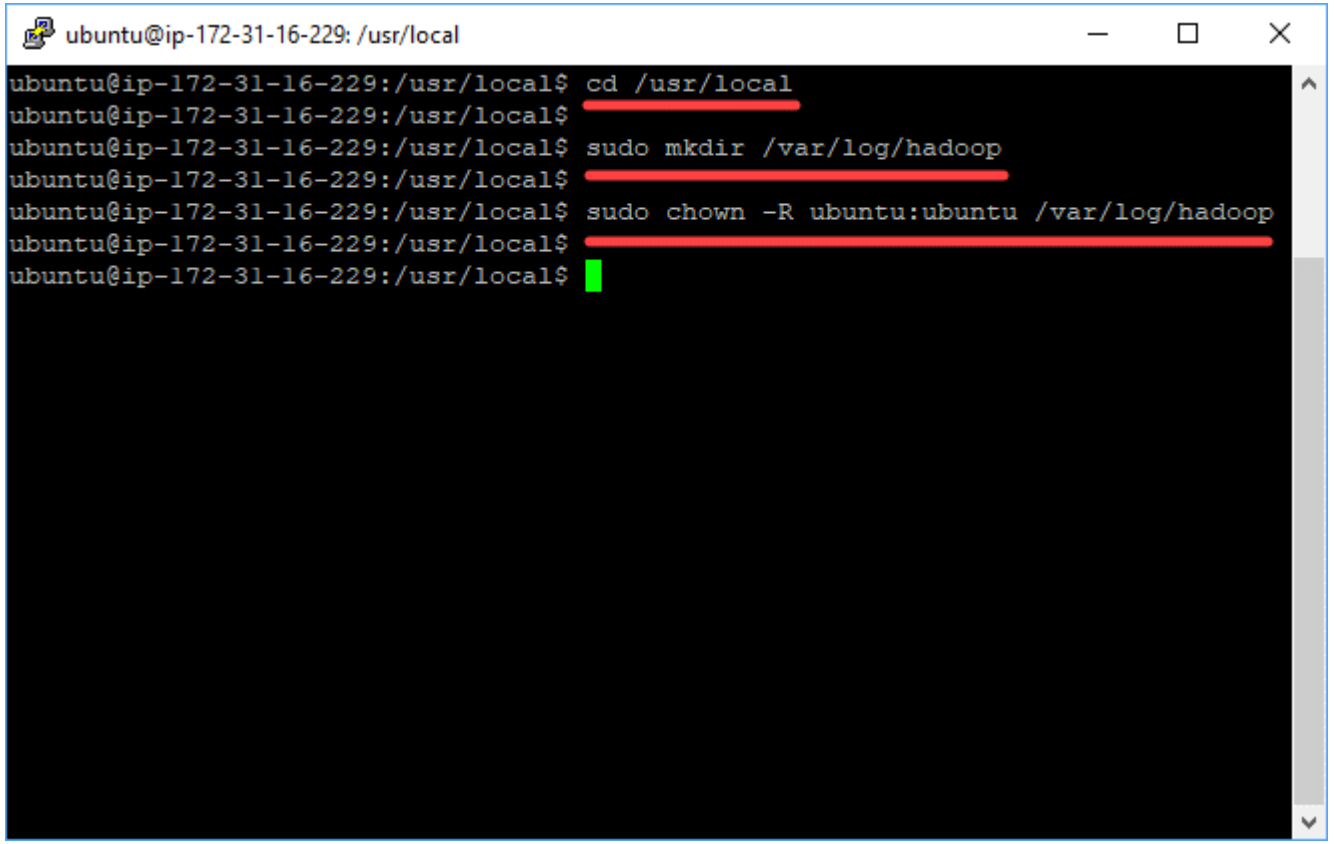
Execute environment variables:

`$source ~/.bashrc`



Create Hadoop log directory:

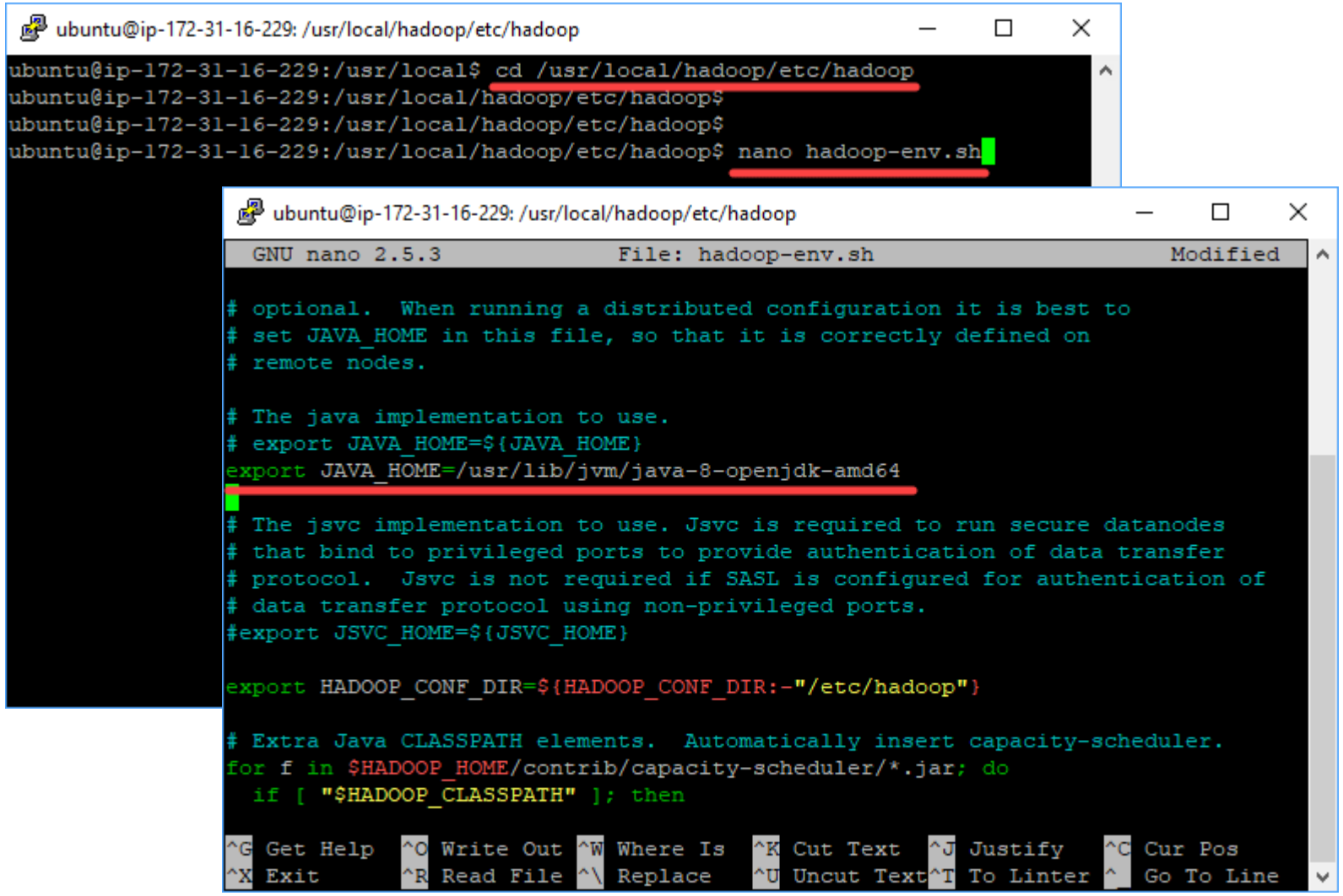
```
$cd /usr/local
$sudo mkdir /var/log/hadoop
$sudo chown –R ubuntu:ubuntu /var/log/hadoop
```



Edit Hadoop shell script:

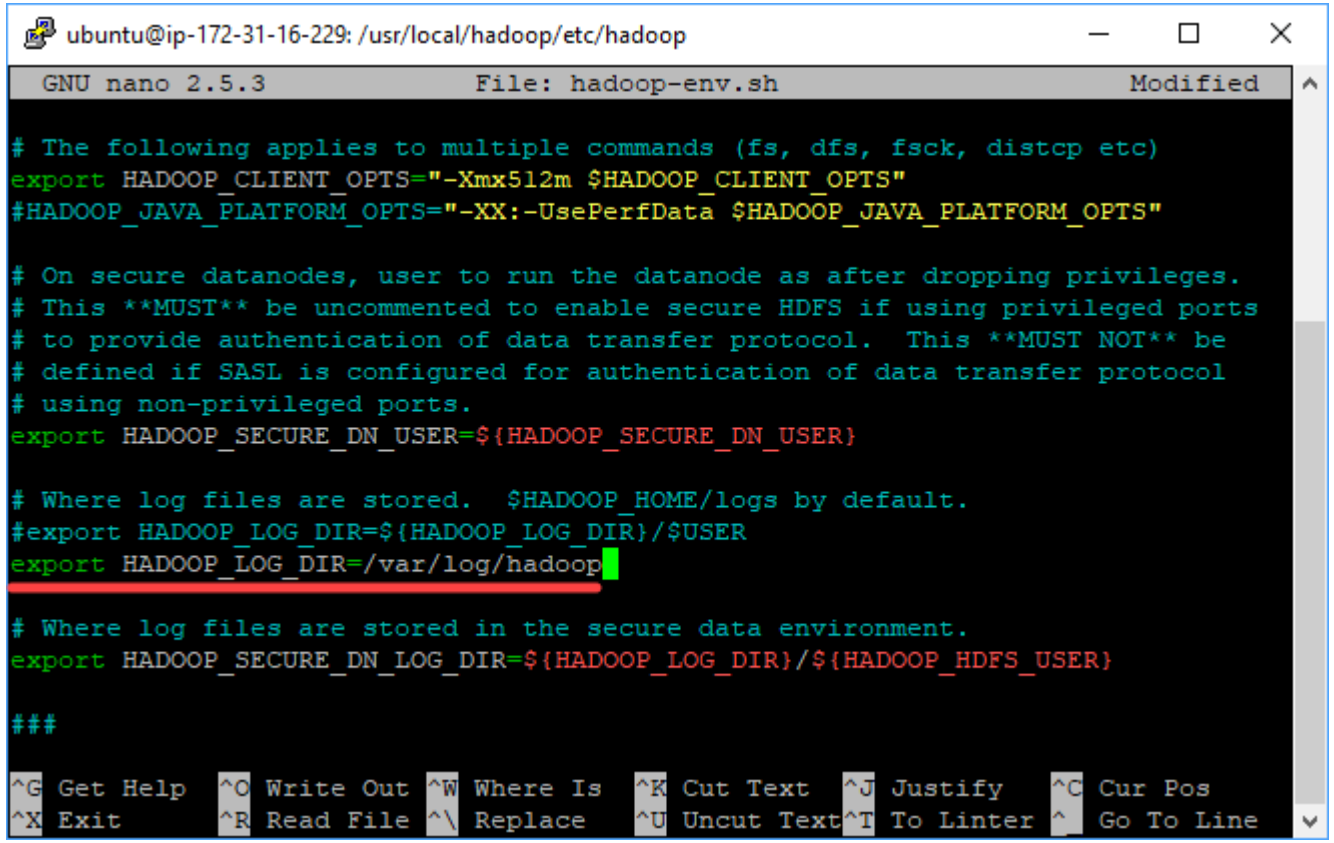
$cd /usr/local/hadoop/etc/hadoop

$nano hadoop-env.sh



Edit Hadoop shell script:

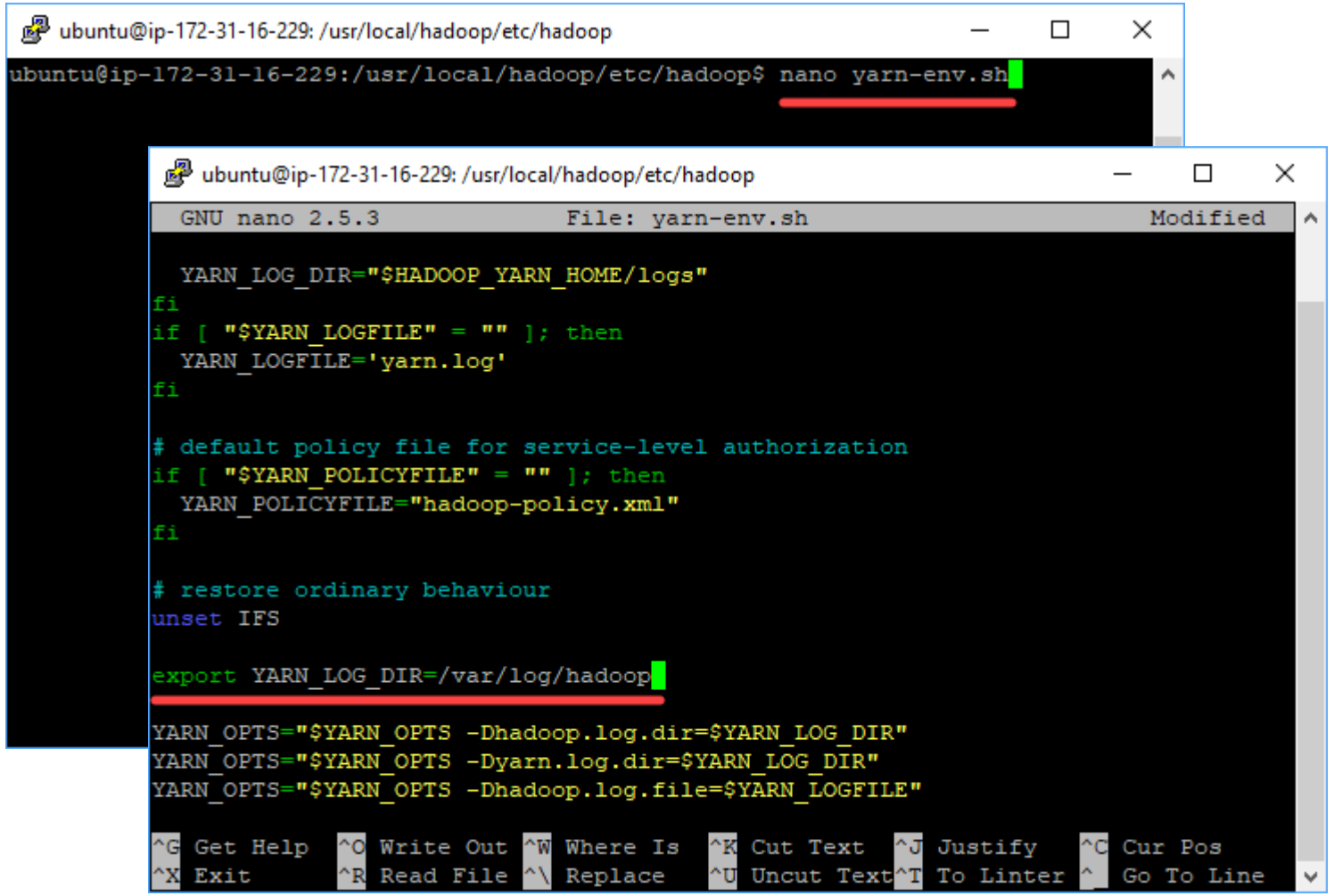
JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
export HADOOP_LOG_DIR=/var/log/hadoop



Edit YARN shell script:

$nano yarn-env.sh

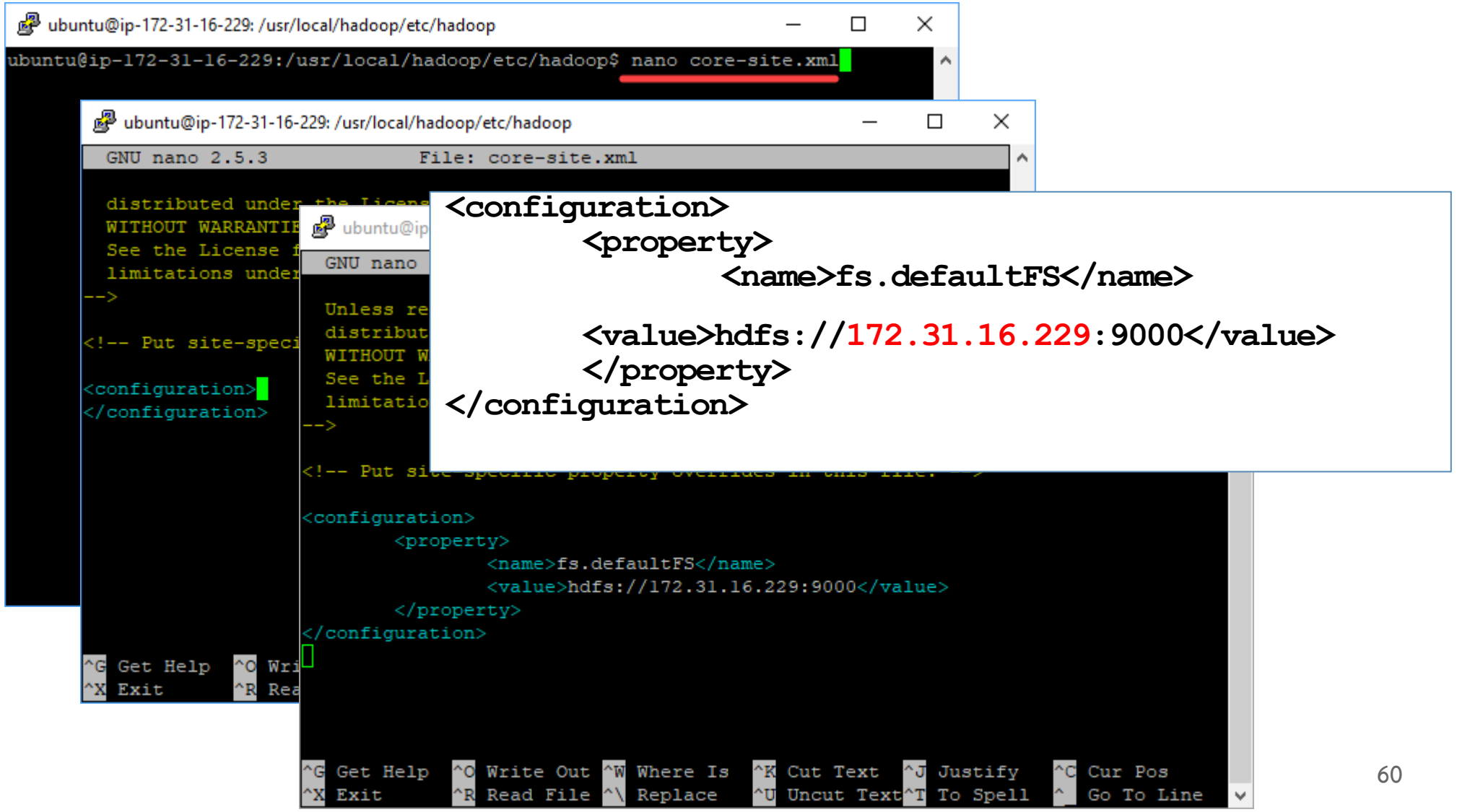
export YARN_LOG_DIR=/var/log/hadoop



2.6 Configure Hadoop

Edit hadoop-core-site.xml:

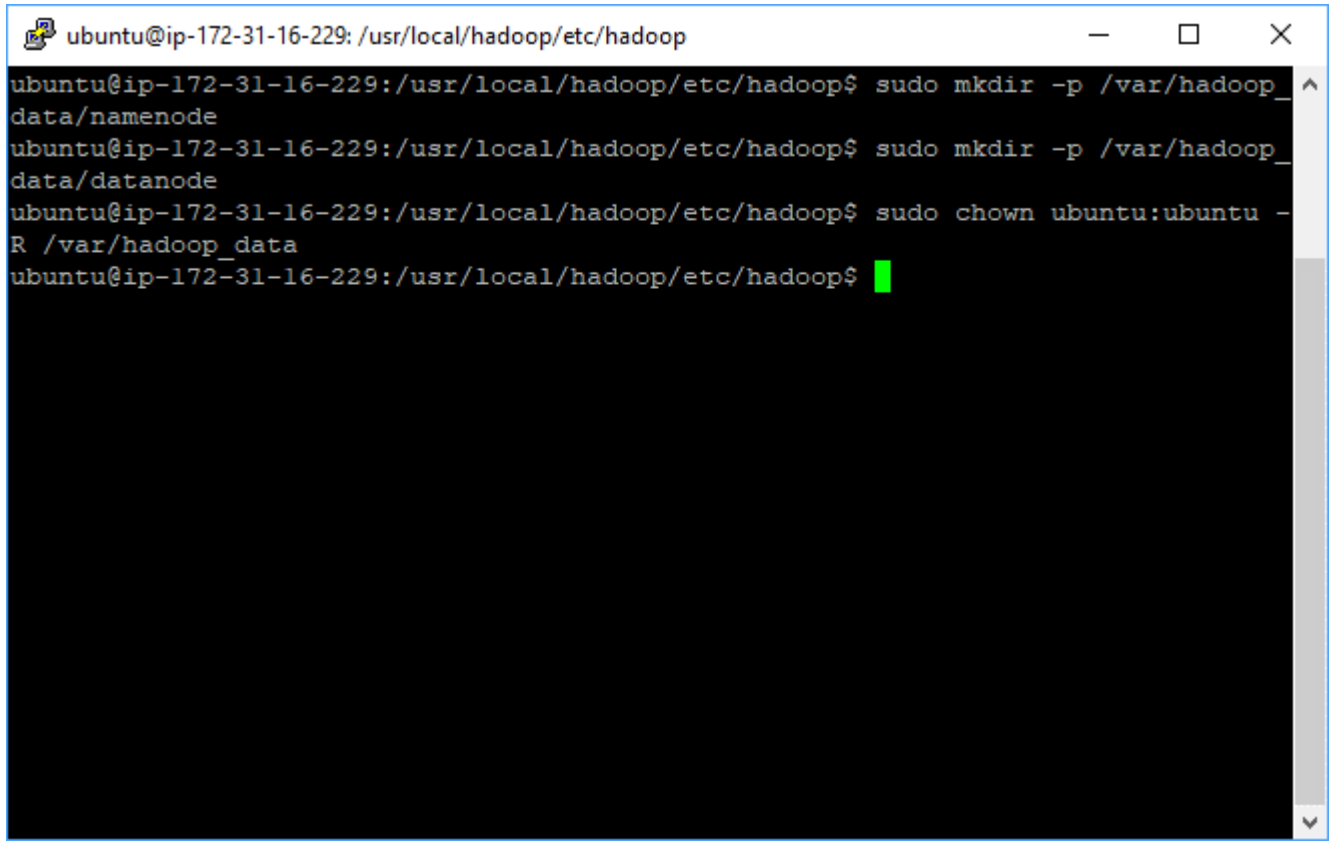
$nano core-site.xml




```
<configuration>
        <property>
                <name>fs.defaultFS</name>
                <value>hdfs://172.31.16.229:9000</value>
        </property>
</configuration>
```

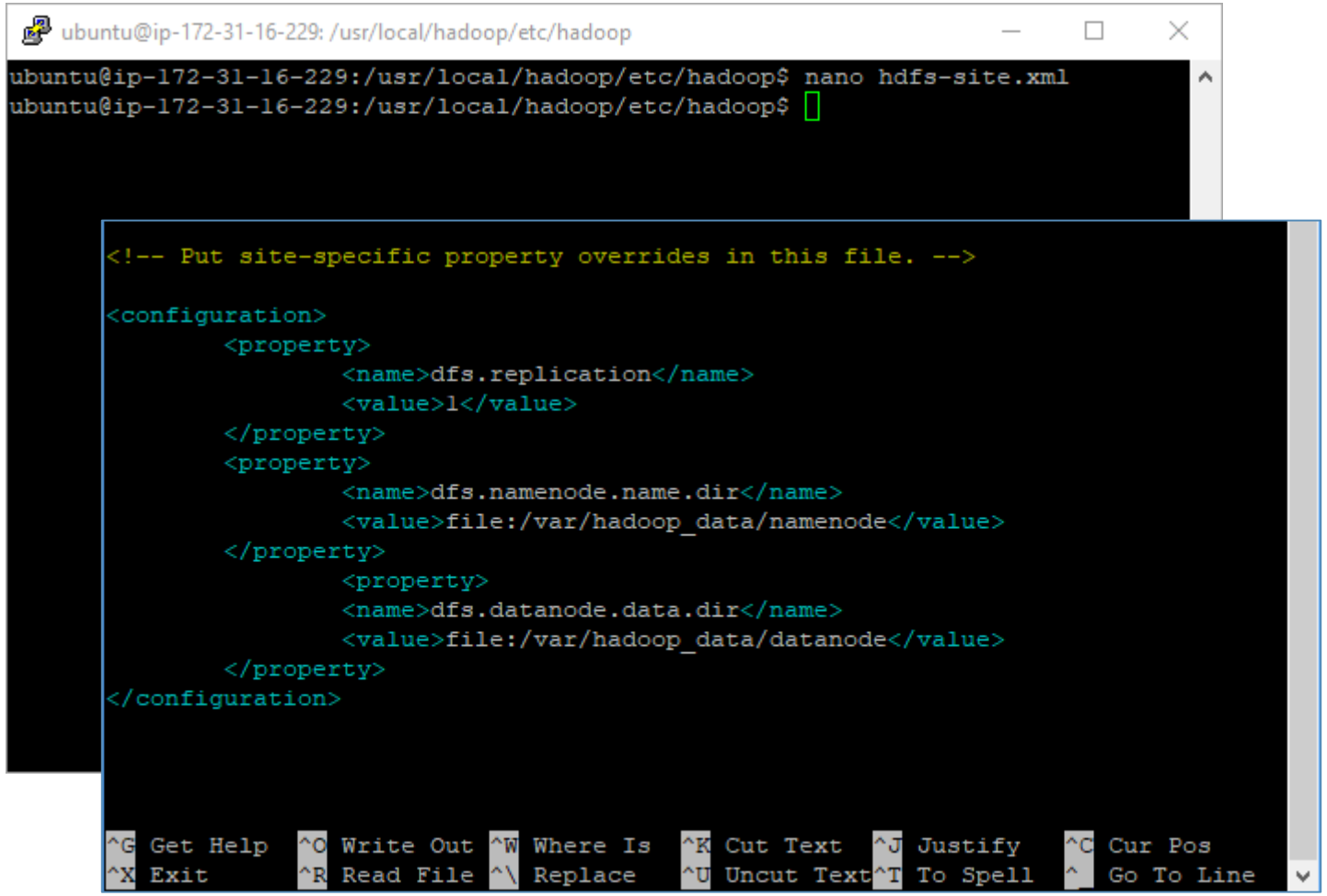
Create directories for Namenode and Datanode:

```
$sudo mkdir -p /var/hadoop_data/namenode
$sudo mkdir -p /var/hadoop_data/datanode
$sudo chown ubuntu:ubuntu -R /var/hadoop_data
```



Edit hdfs-site.xml:

$nano hdfs-site.xml



Edit hdfs-site.xml:

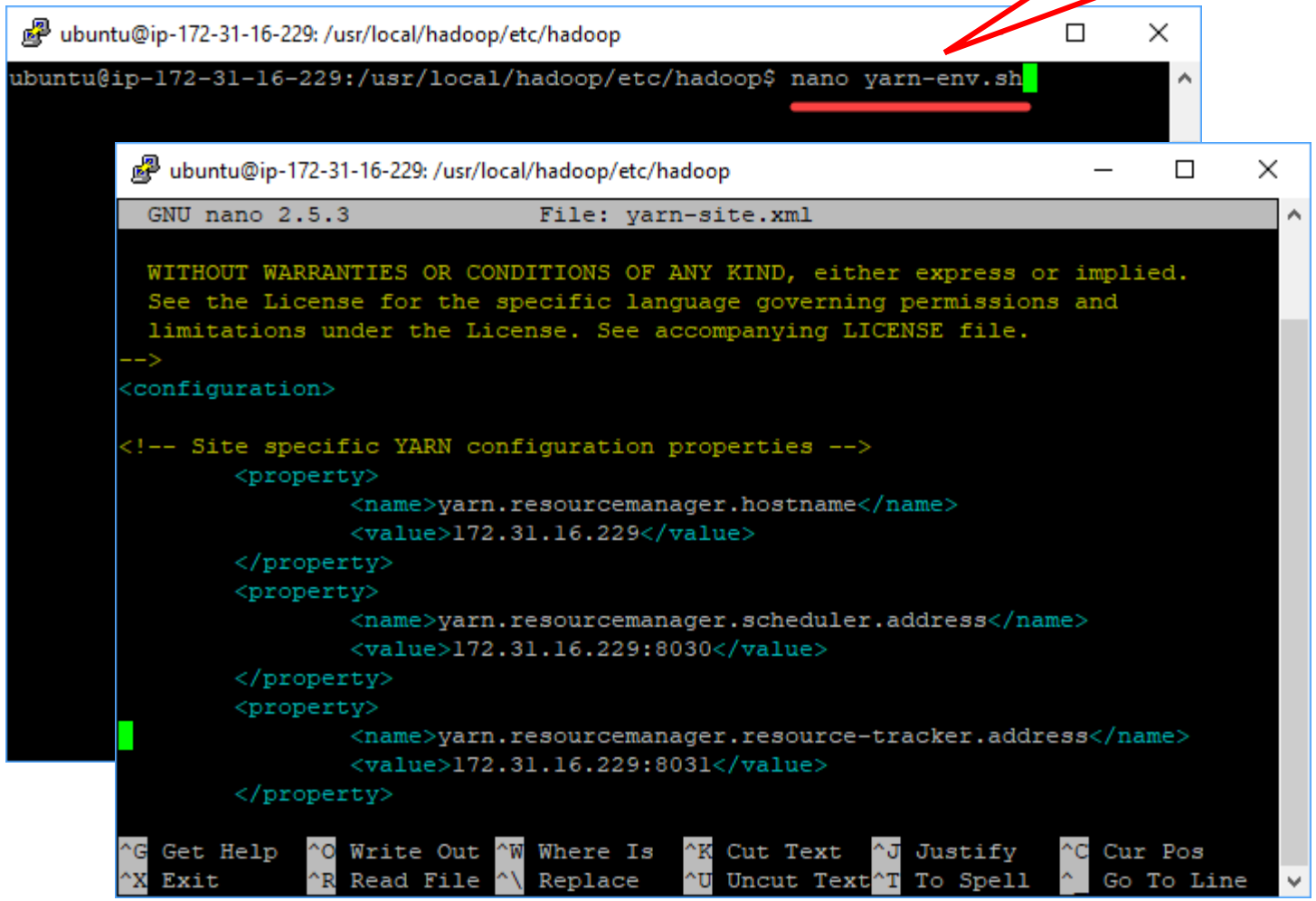
```
<configuration>
        <property>
                <name>dfs.replication</name>
                <value>1</value>
        </property>
        <property>
                <name>dfs.namenode.name.dir</name>
                <value>file:/var/hadoop_data/namenode</value>
        </property>
                <property>
                <name>dfs.datanode.data.dir</name>
                <value>file:/var/hadoop_data/datanode</value>
        </property>
</configuration>
```

Edit yarn-site.xml:

$nano yarn-site.xml

change to

nano yarn-site.xml

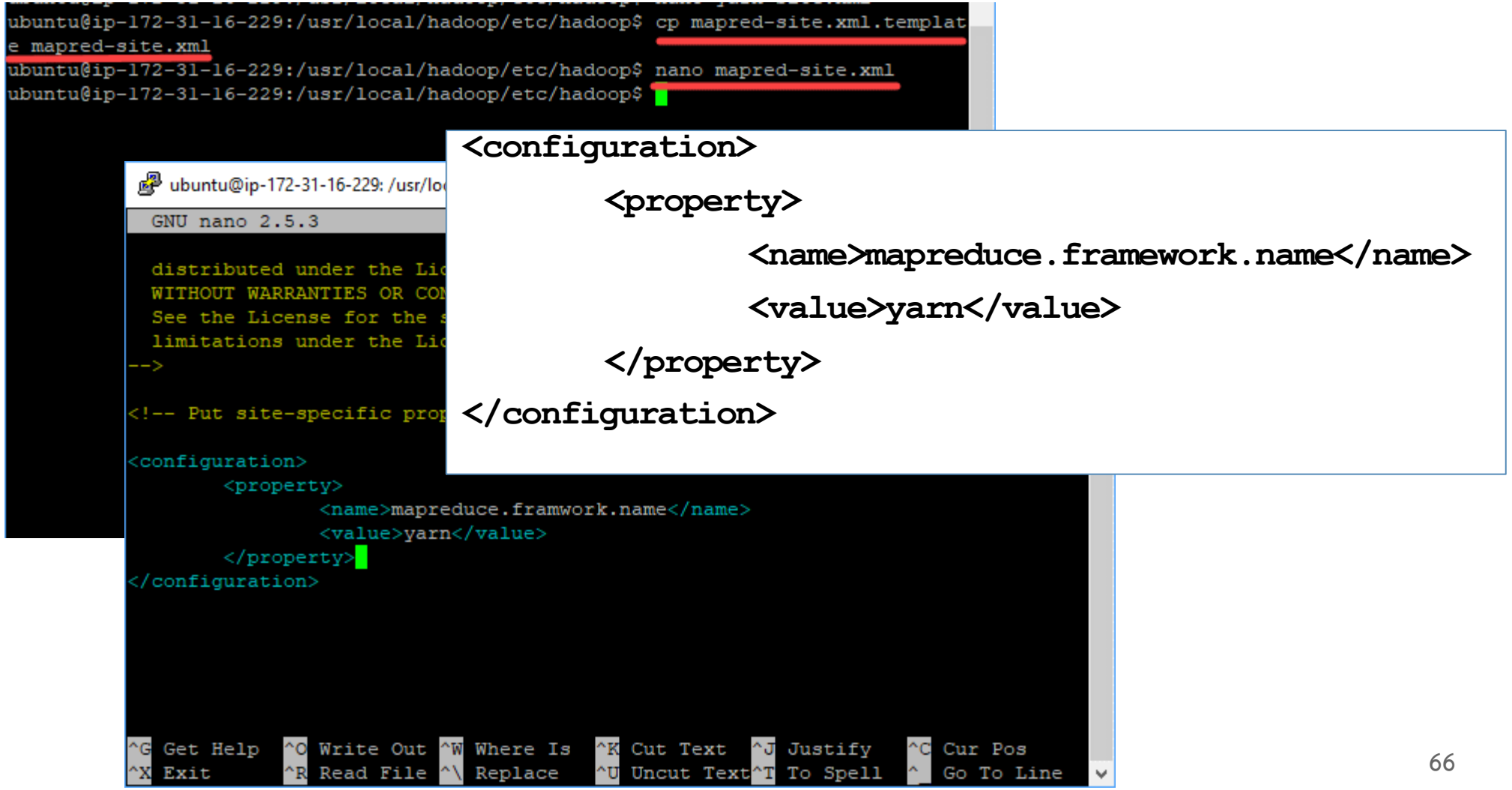


Edit yarn-site.xml:

```
<configuration>
        <property>
                <name>yarn.resourcemanager.hostname</name>
                <value>172.31.16.229</value>
        </property>
        <property>
                <name>yarn.resourcemanager.scheduler.address</name>
                <value>172.31.16.229:8030</value>
        </property>
        <property>
                <name>yarn.resourcemanager.resource-tracker.address</name>
                <value>172.31.16.229:8031</value>
        </property>
        <property>
                <name>yarn.resourcemanager.address</name>
                <value>172.31.16.229:8032</value>
        </property>
        <property>
                <name>yarn.resourcemanager.admin.address</name>
                <value>172.31.16.229:8033</value>
        </property>
        <property>
                <name>yarn.resourcemanager.webapp.address</name>
                <value>172.31.16.229:8088</value>
        </property>
        <property>
                <name>yarn.nodemanager.aux-services</name>
                <value>mapreduce_shuffle</value>
        </property>
        <property>
                <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
                <value>org.apache.hadoop.mapred.ShuffleHandler</value>
        </property>
</configuration>
```

Edit mapred-site.xml:

```
$cp mapred-site.xml.template mapred-site.xml
$nano mapred-site.xml
```

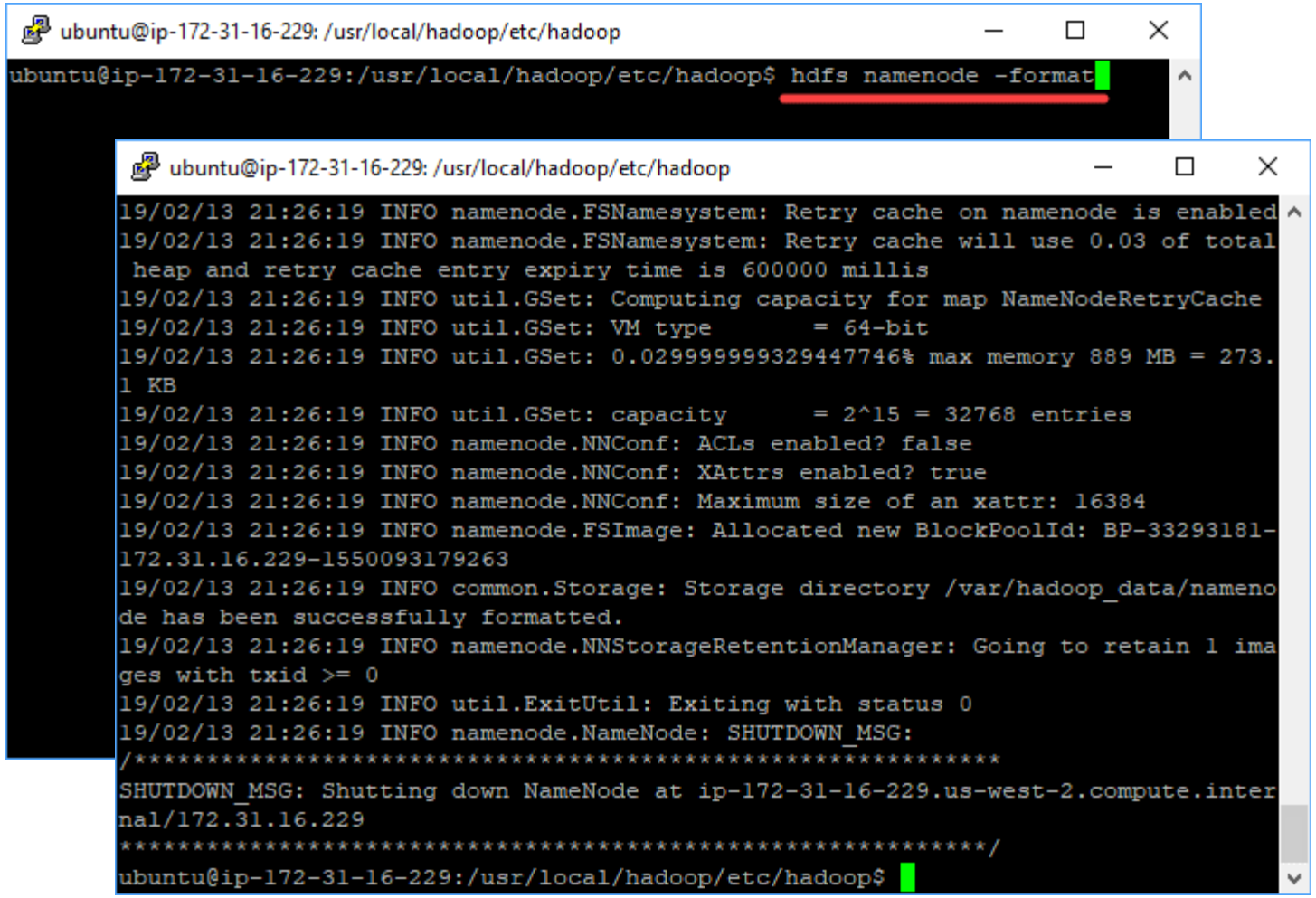



```
<configuration>
        <property>
                <name>mapreduce.framework.name</name>
                <value>yarn</value>
        </property>
</configuration>
```

2.7 Format Namenode

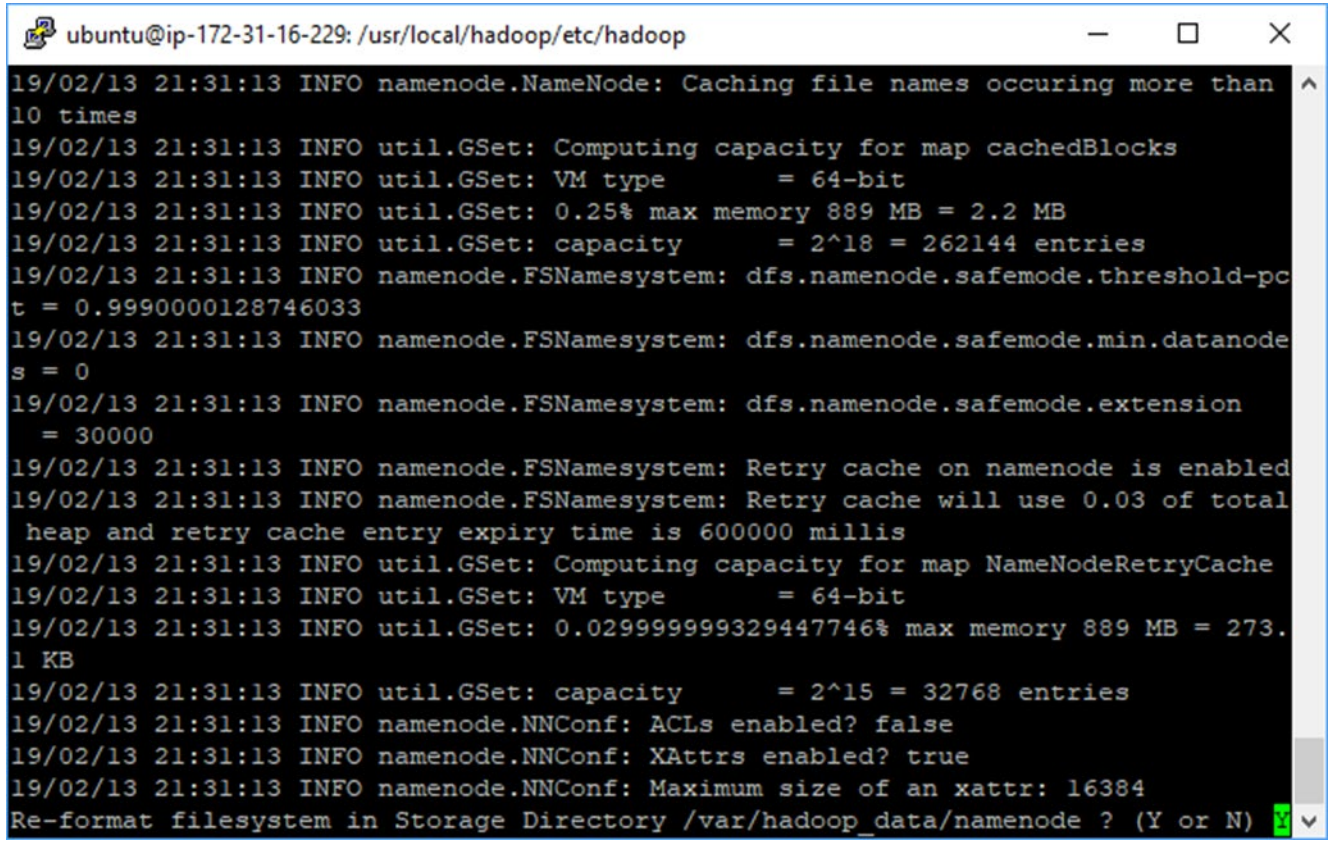
Formatting Namenode:

`$hdfs namenode -format`



Reformatting Namenode to verify:

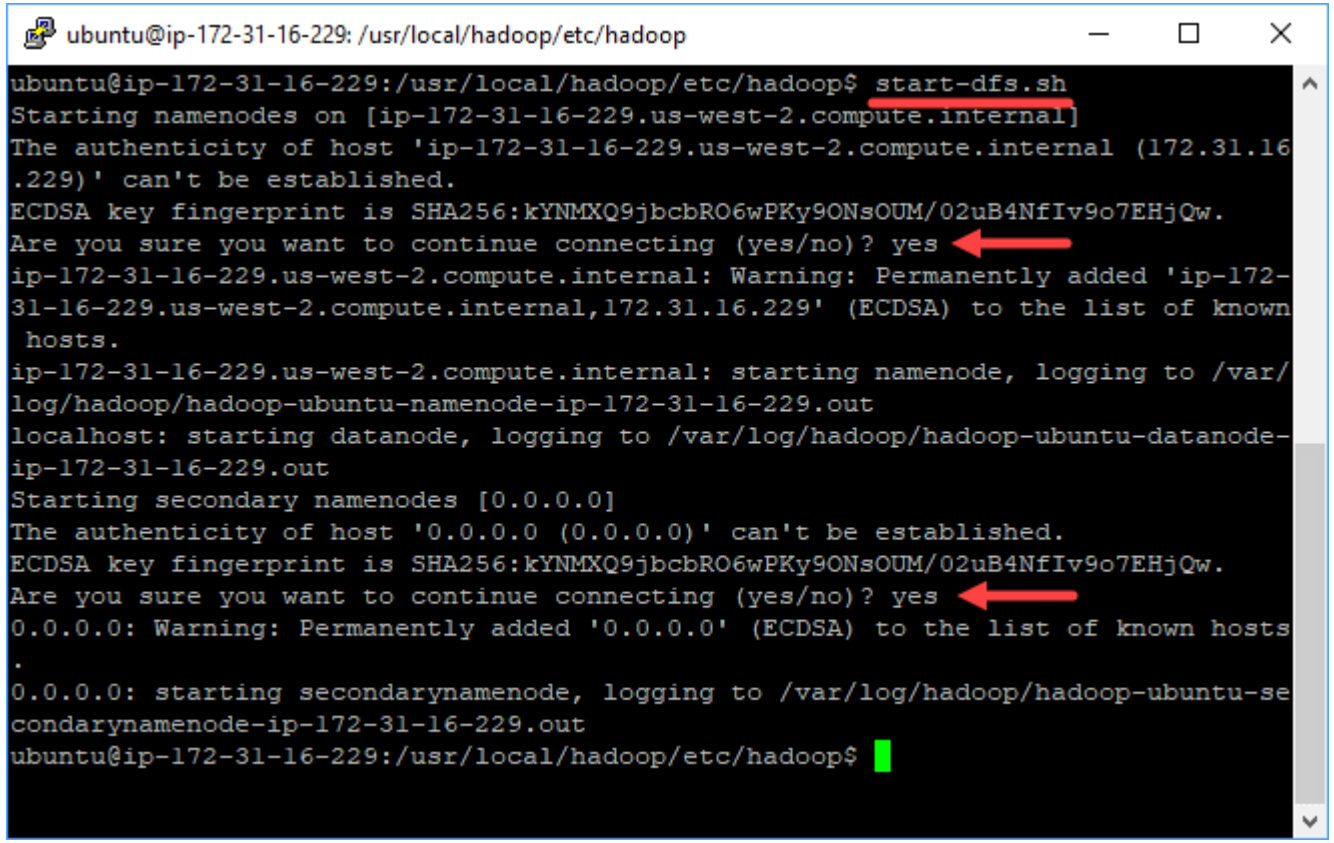
`$hdfs namenode -format`



2.8 Start Hadoop

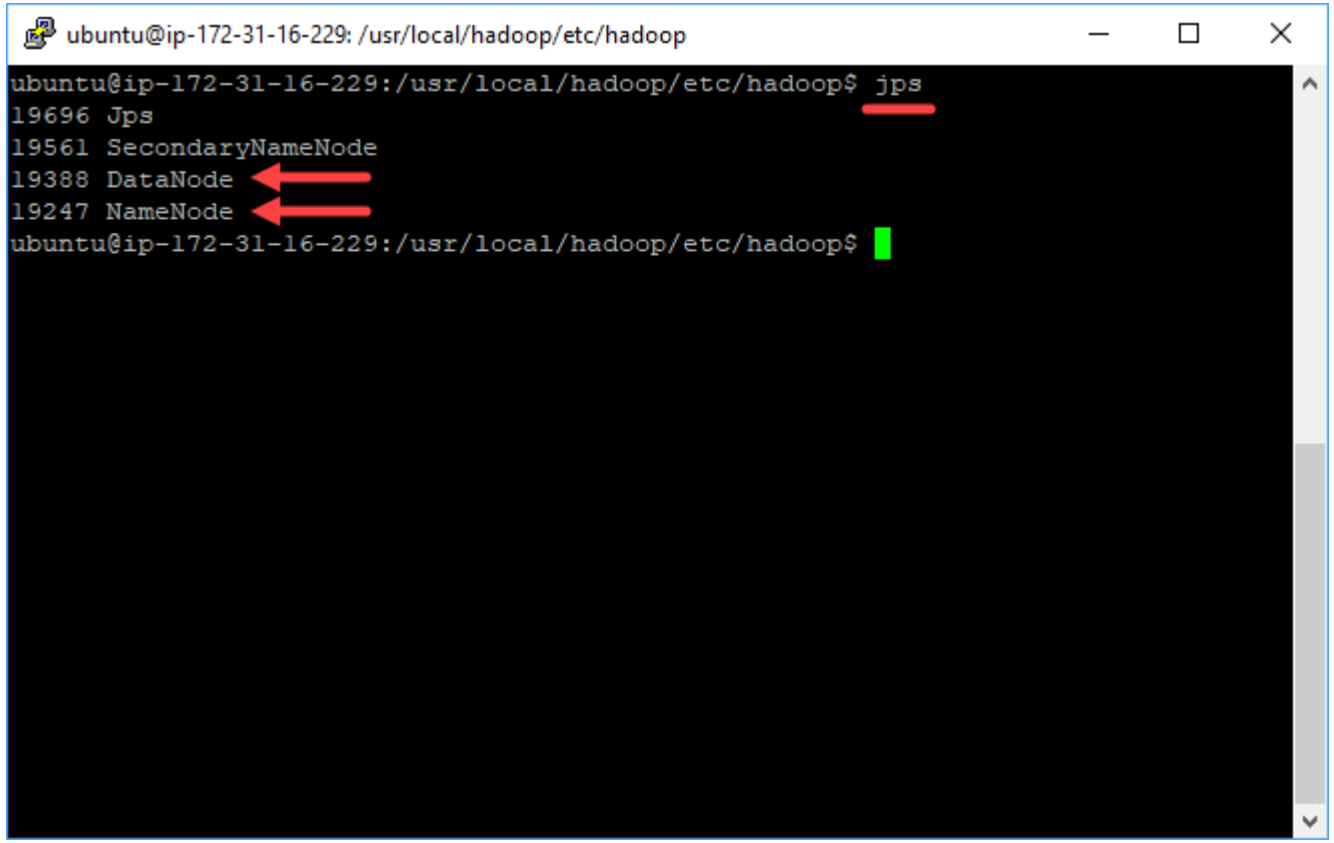
Start Namenode and Datanode:

`$start-dfs.sh`



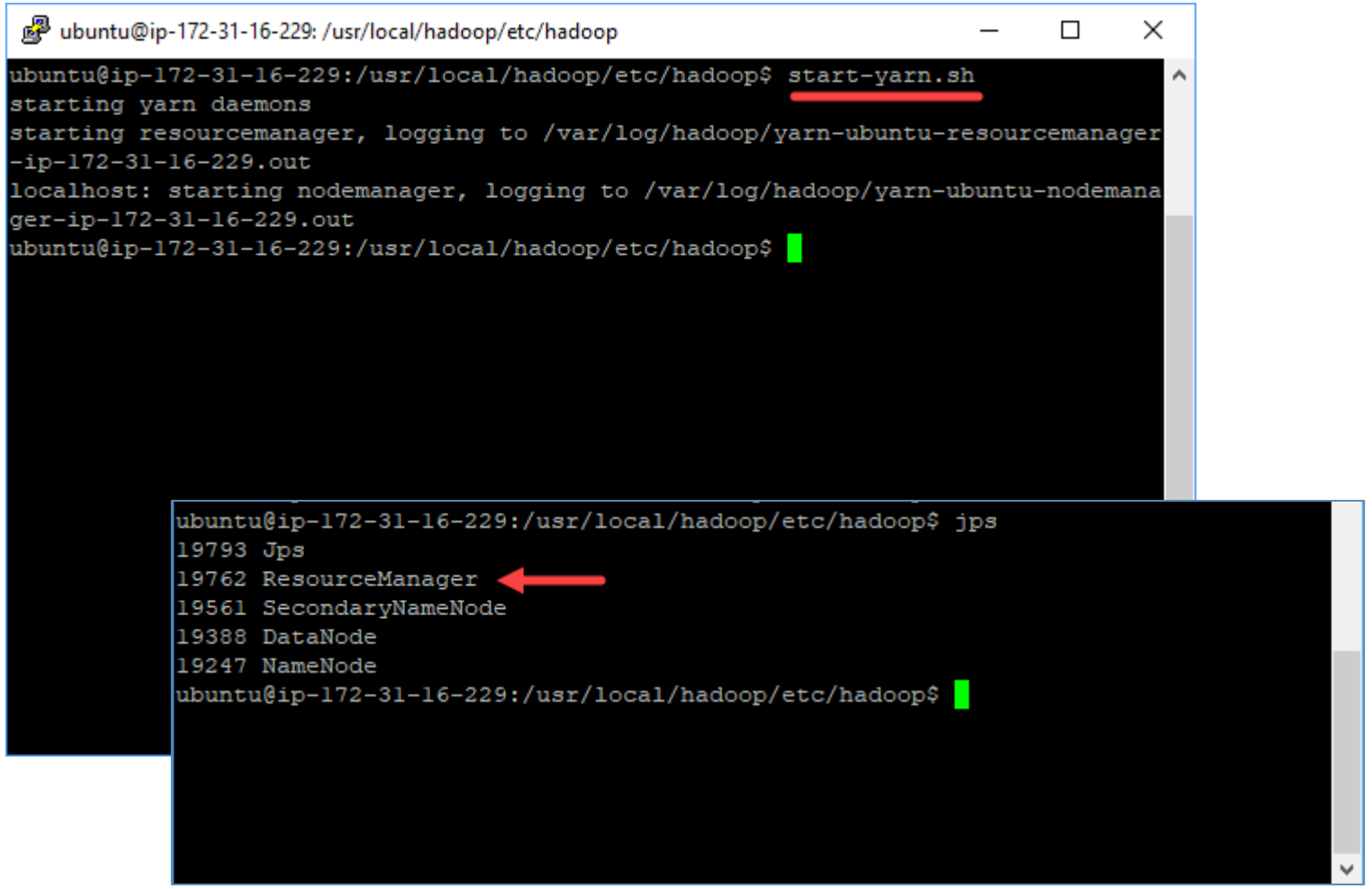
Verify Namenode and Datanode:

`$jps`



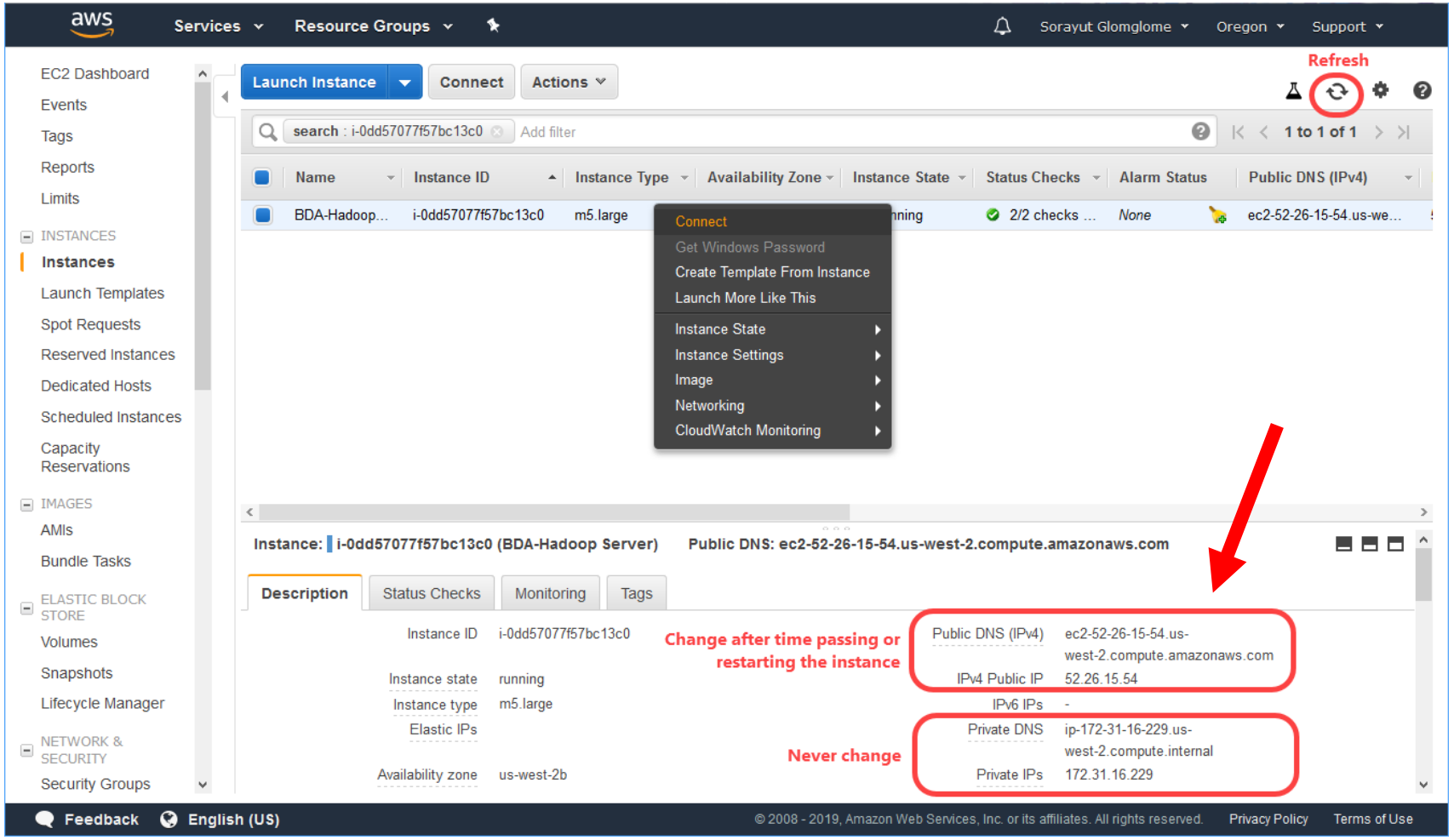
Start YARN:

`$start-yarn.sh`



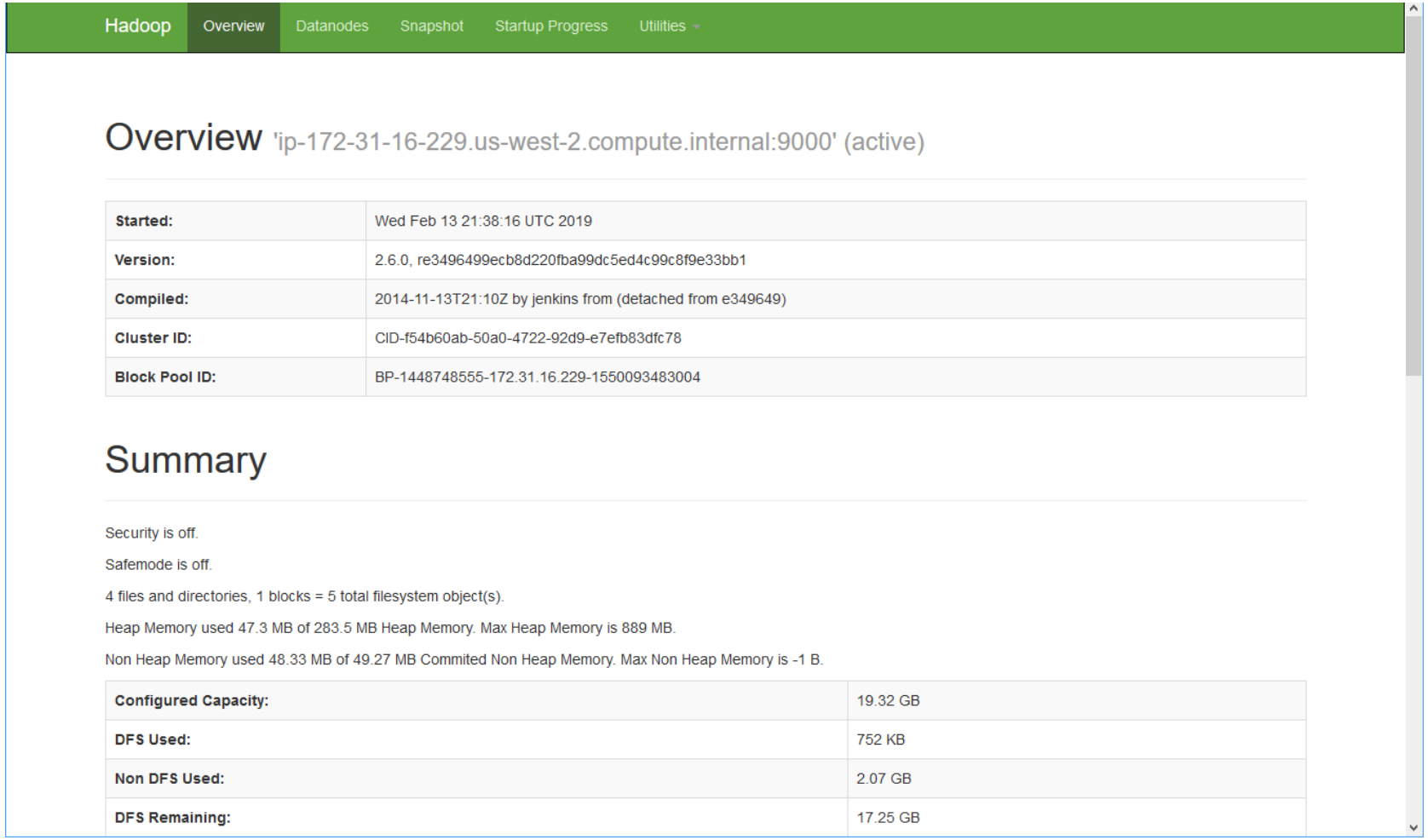
2.9 Access Hadoop Web Console

Checking Public IP and DNS



Namenode Information:

http://52.26.15.54:50070



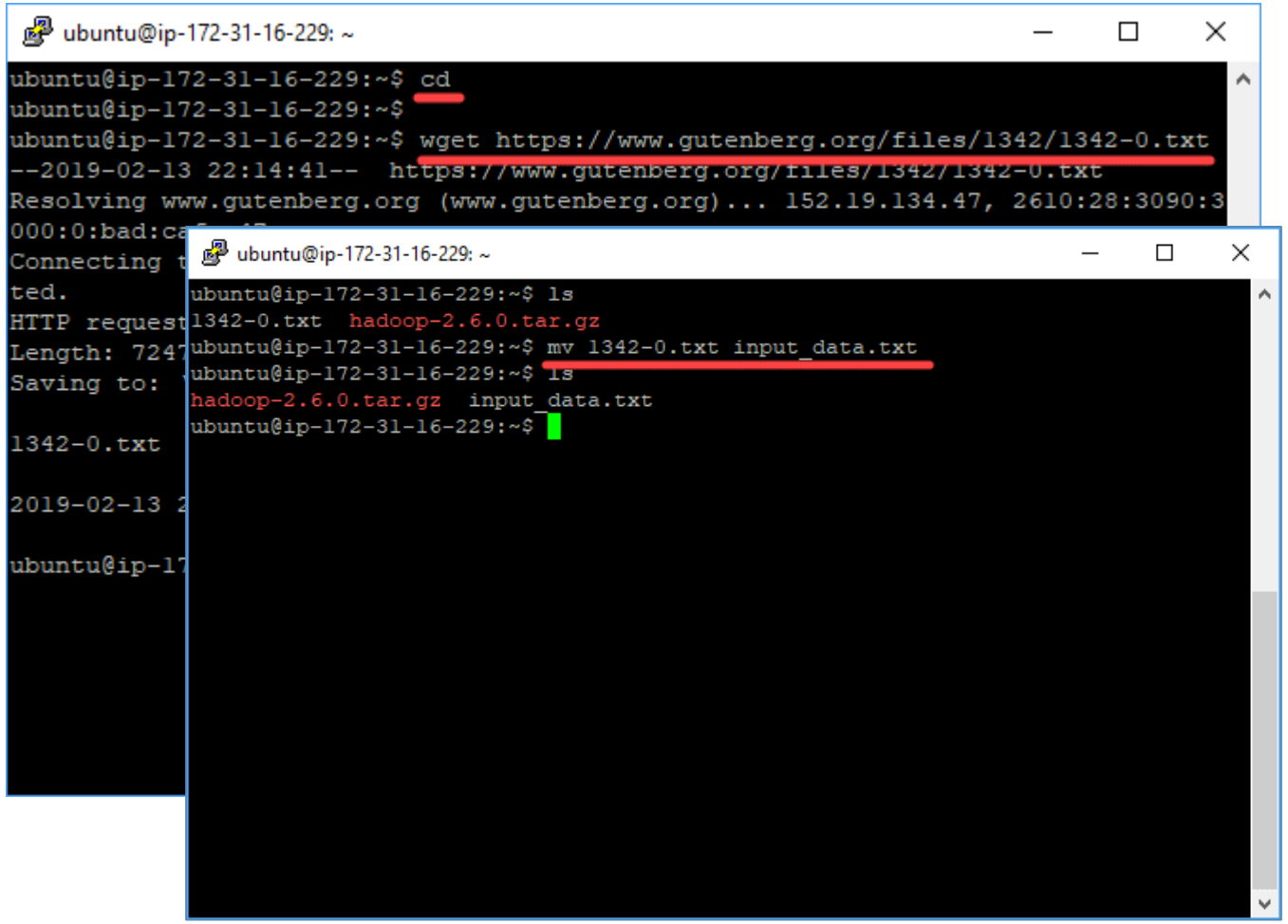
Hadoop Port Numbers:

	Daemon	Default Port	Configuration Parameter in conf/*-site.xml
HDFS	Namenode Web	50070	dfs.http.address
	Datanodes	50075	dfs.datanode.http.address
	Secondarynamenode	50090	dfs.secondary.http.address
Yarn	ResourceManager Web	8088	yarn.resourcemanager.webapp.address

2.10 Import Data to HDFS using Hadoop Command Line

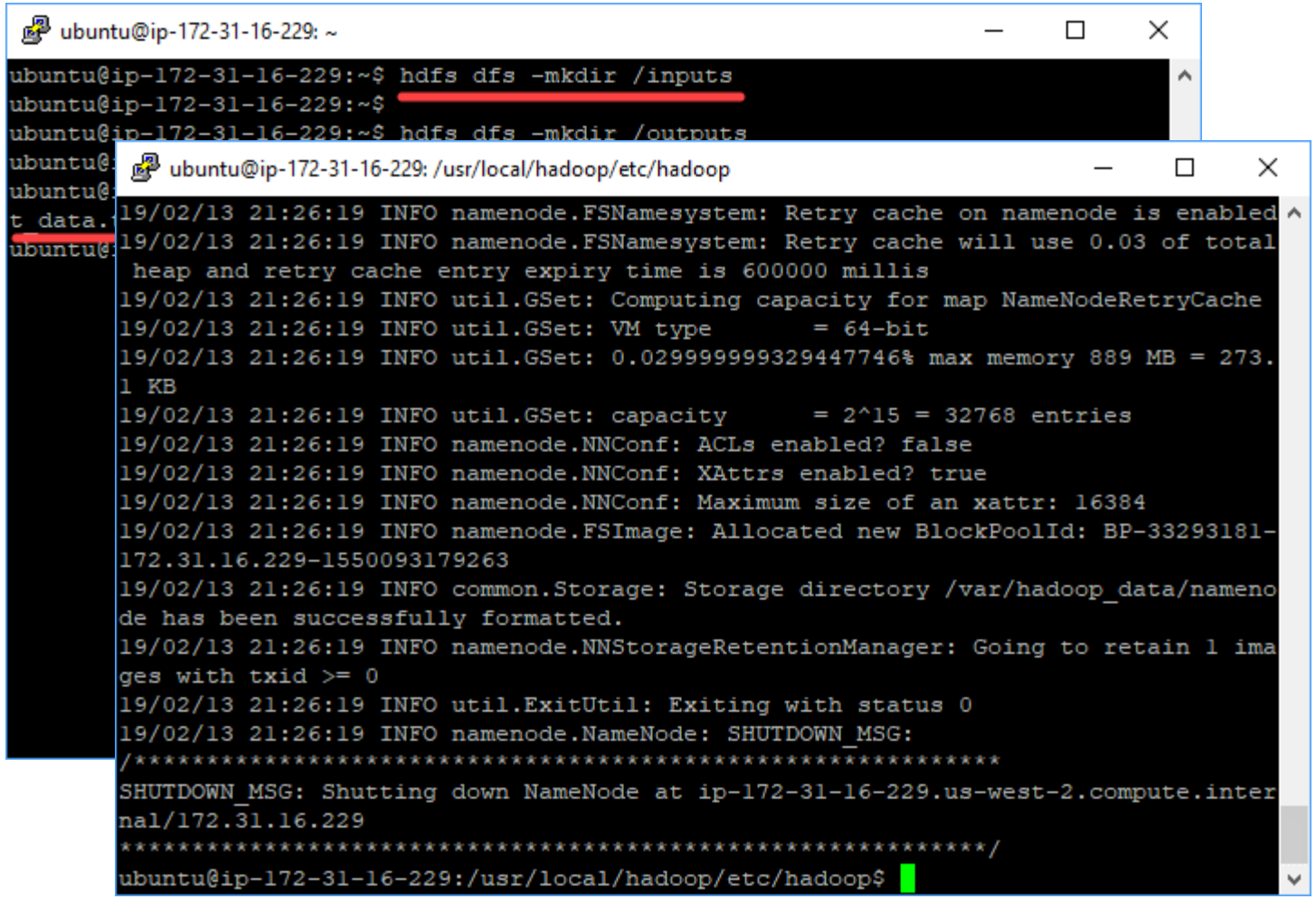
Get and rename file :

```
$cd
$wget https://www.gutenberg.org/files/1342/1342-0.txt
$mv 1342-0.txt input_data.txt
```



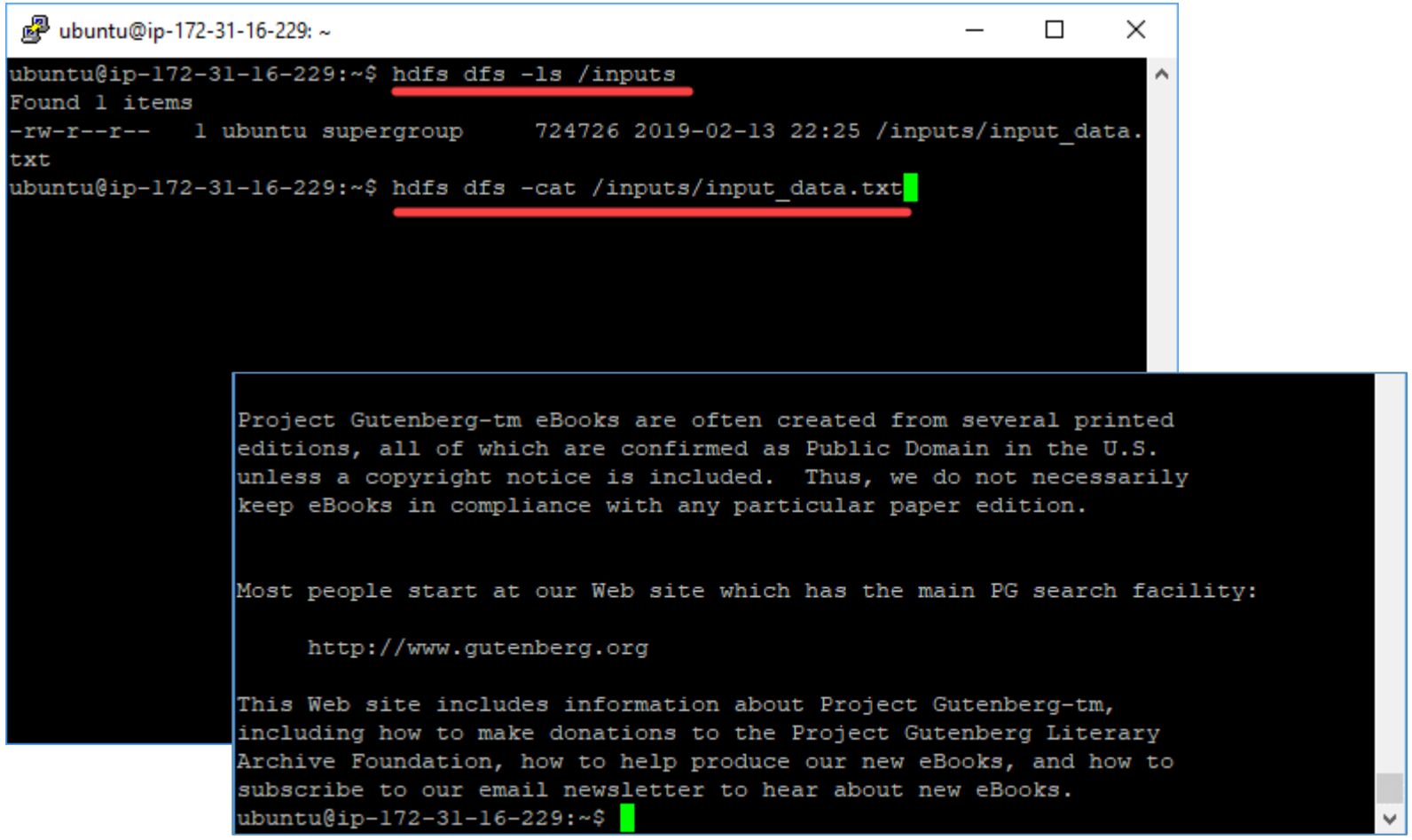
Importing data to HDFS:

```
$hdfs dfs -mkdir /inputs
$hdfs dfs -mkdir /outputs
$hdfs dfs -copyFromLocal ./input_data.txt /inputs/input_data.txt
```



Verify data in HDFS:

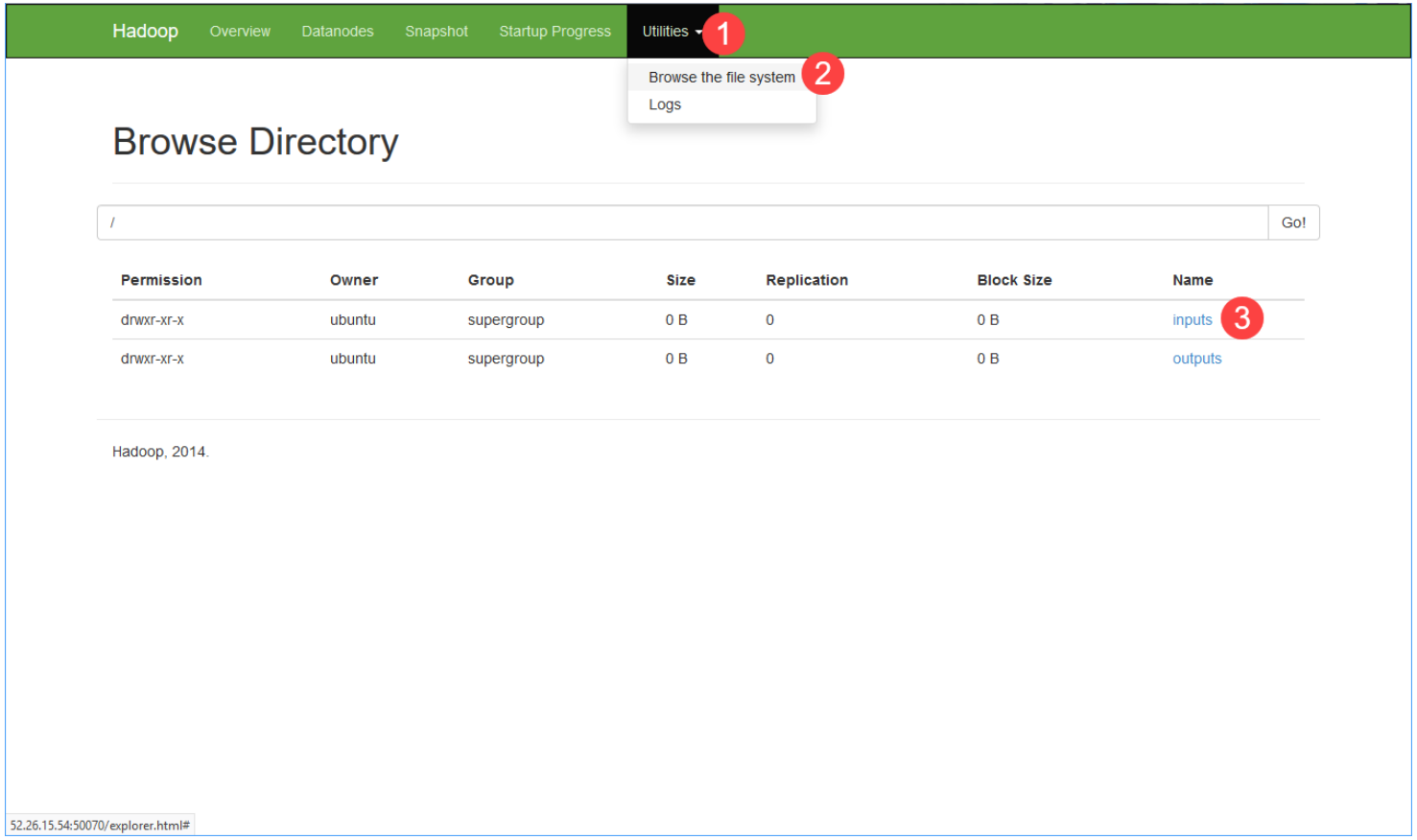
```
$hdfs dfs -ls /inputs
$hdfs dfs -cat /inputs/input_data.txt
```



Verify data using web console:

http://52.26.15.54:50070

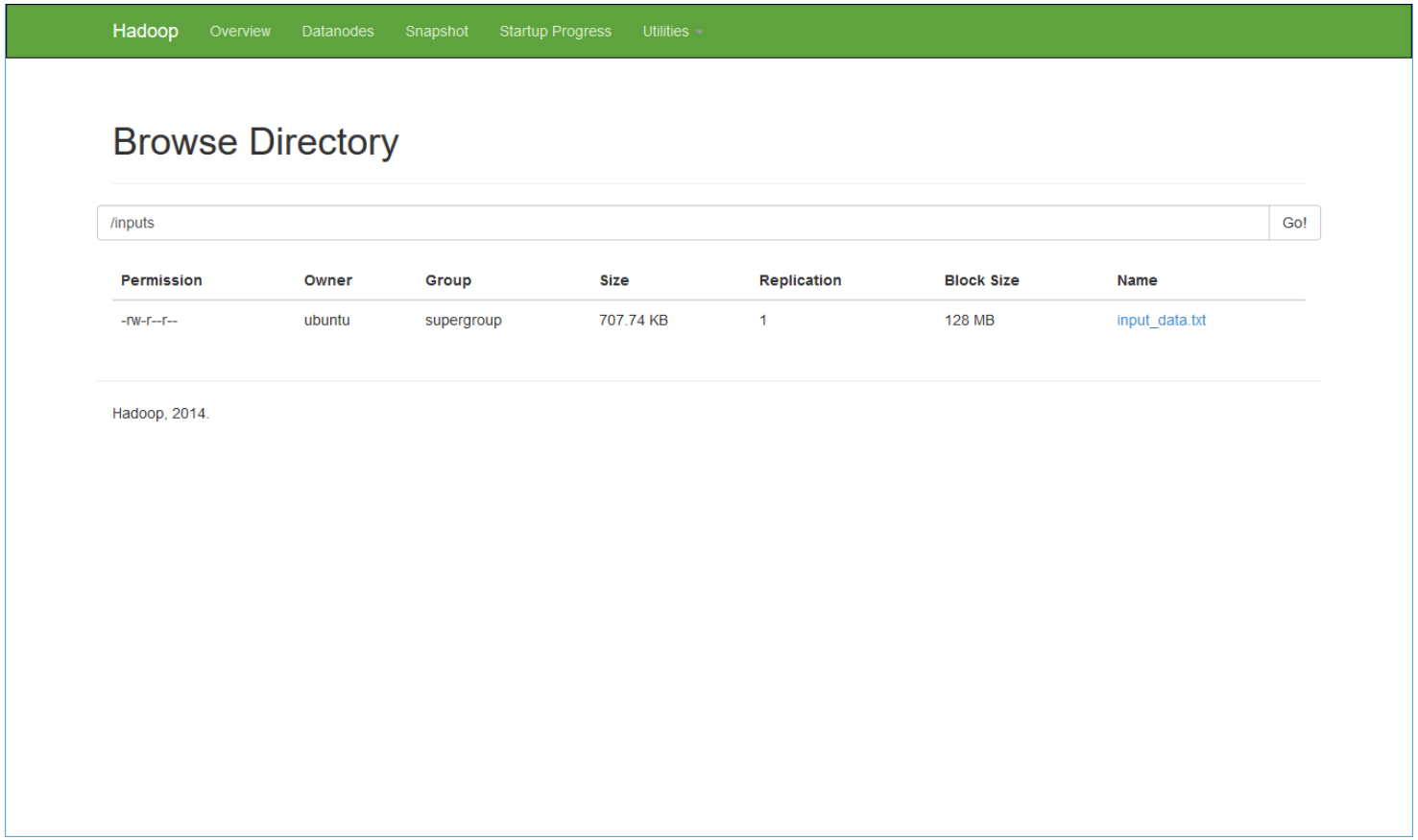
Utilities -> Browse the file system ->inputs



Verify by web console:

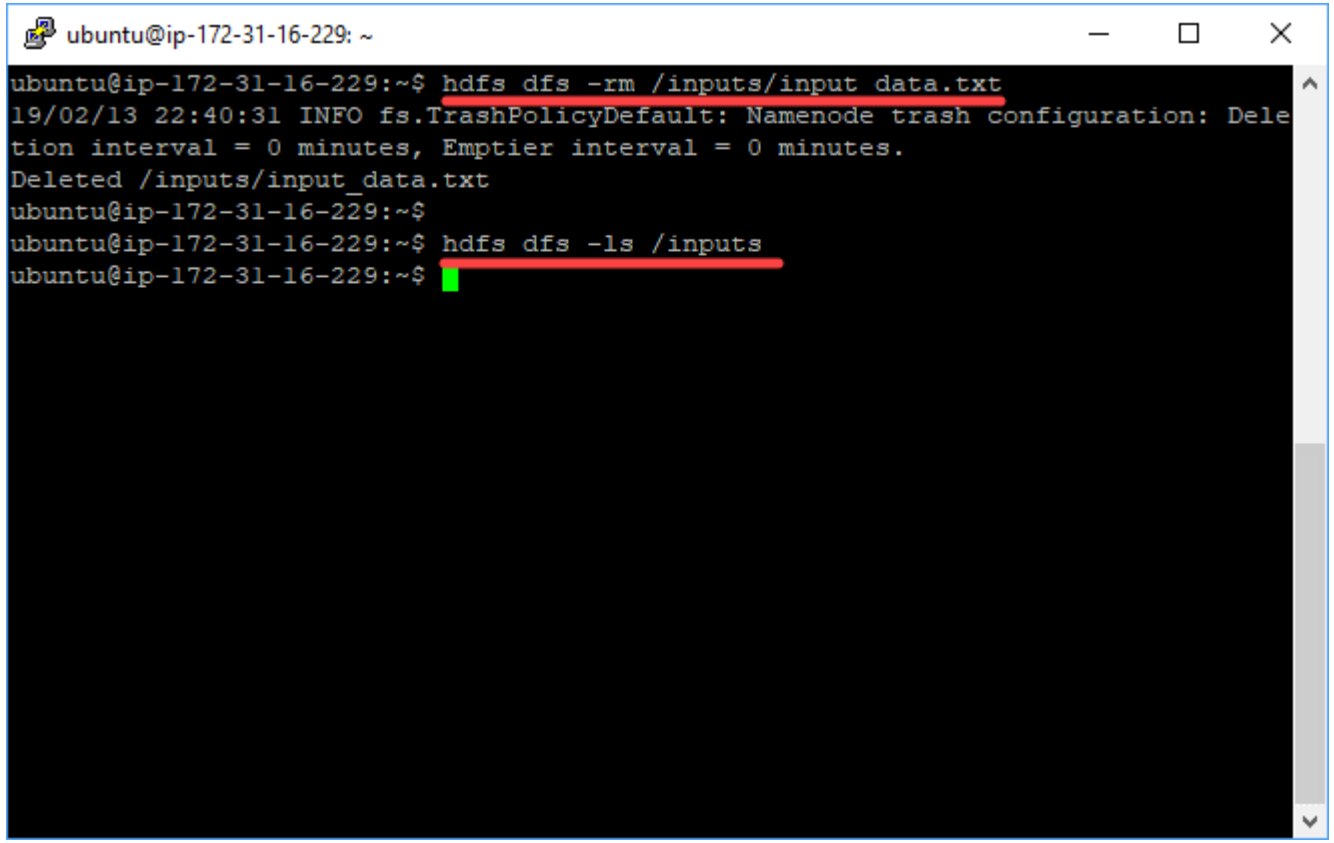
http://52.26.15.54:50070

Utilities -> Browse the file system ->inputs



Remove data from HDFS:

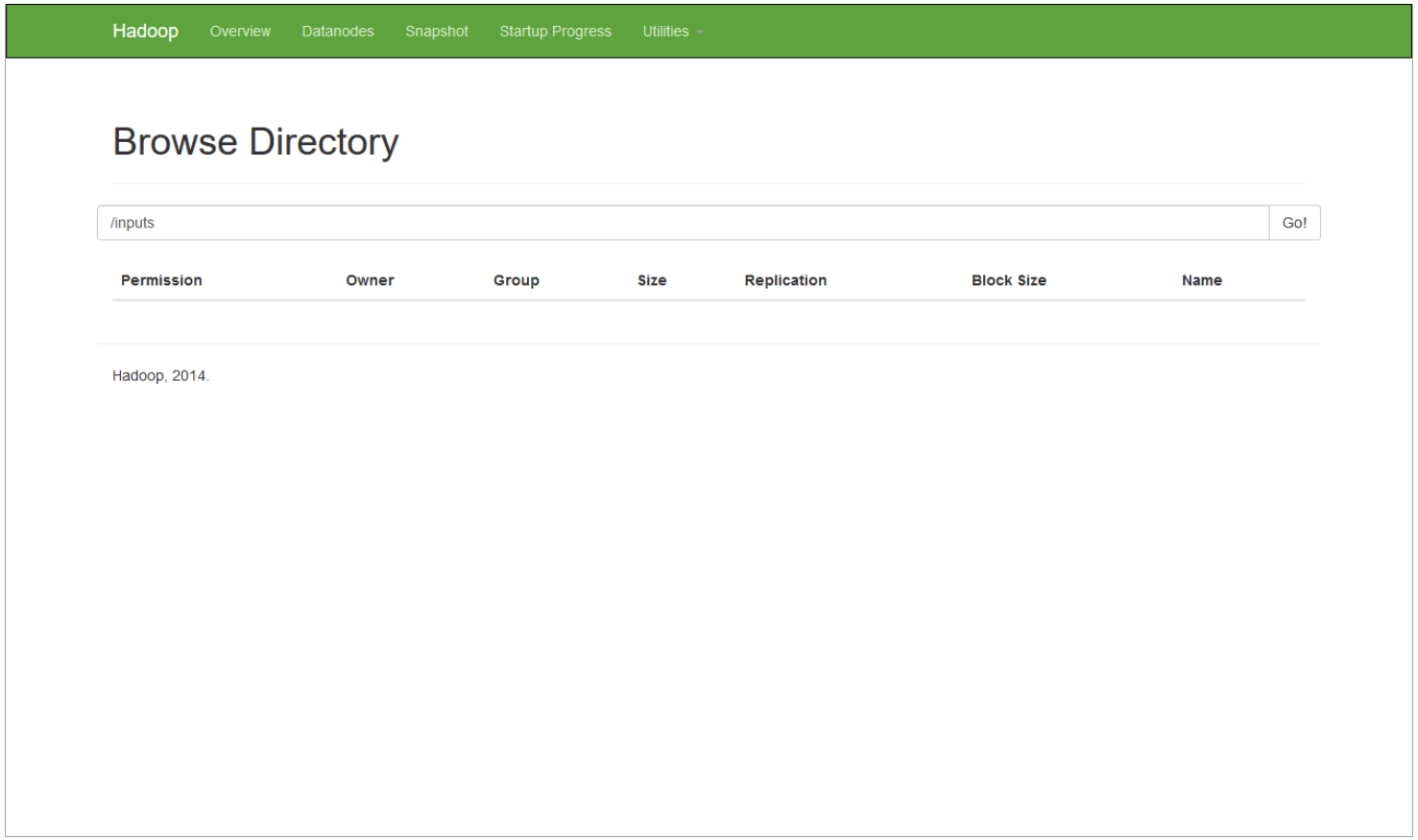
$hdfs dfs -rm /inputs/input_data.txt



Verify using web console:

http://52.26.15.54:50070

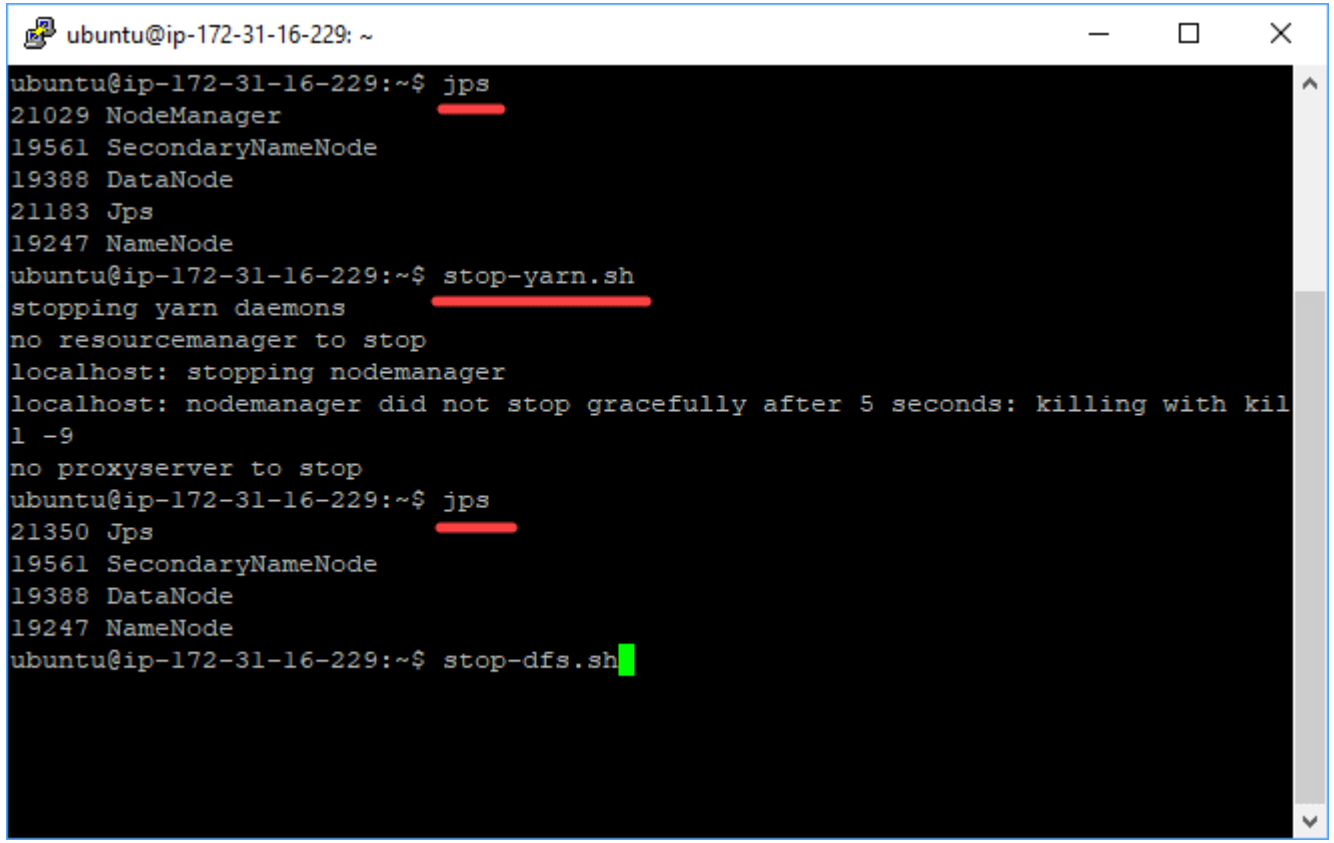
Utilities -> Browse the file system ->inputs



2.11 Stop Hadoop

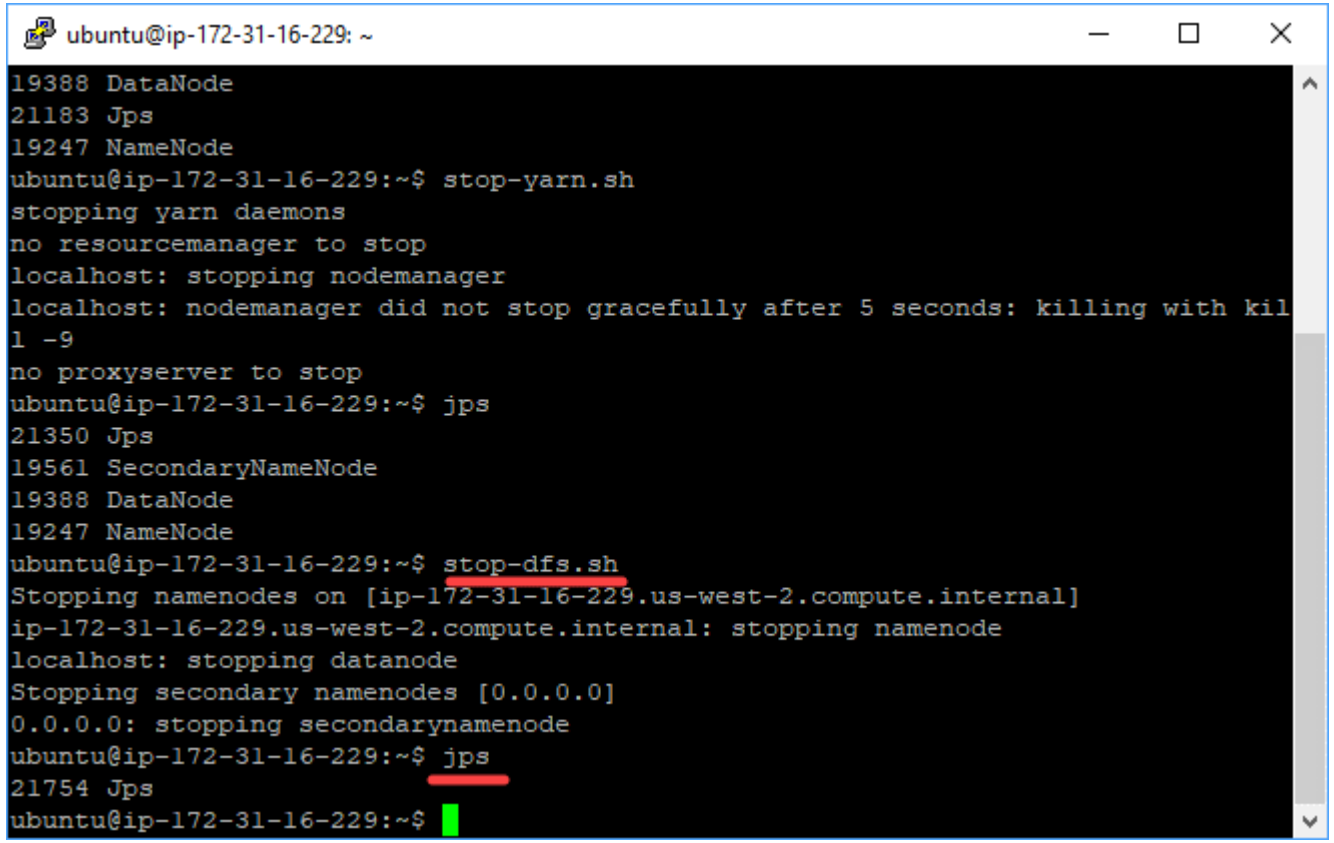
Stop YARN:

$stop-yarn.sh



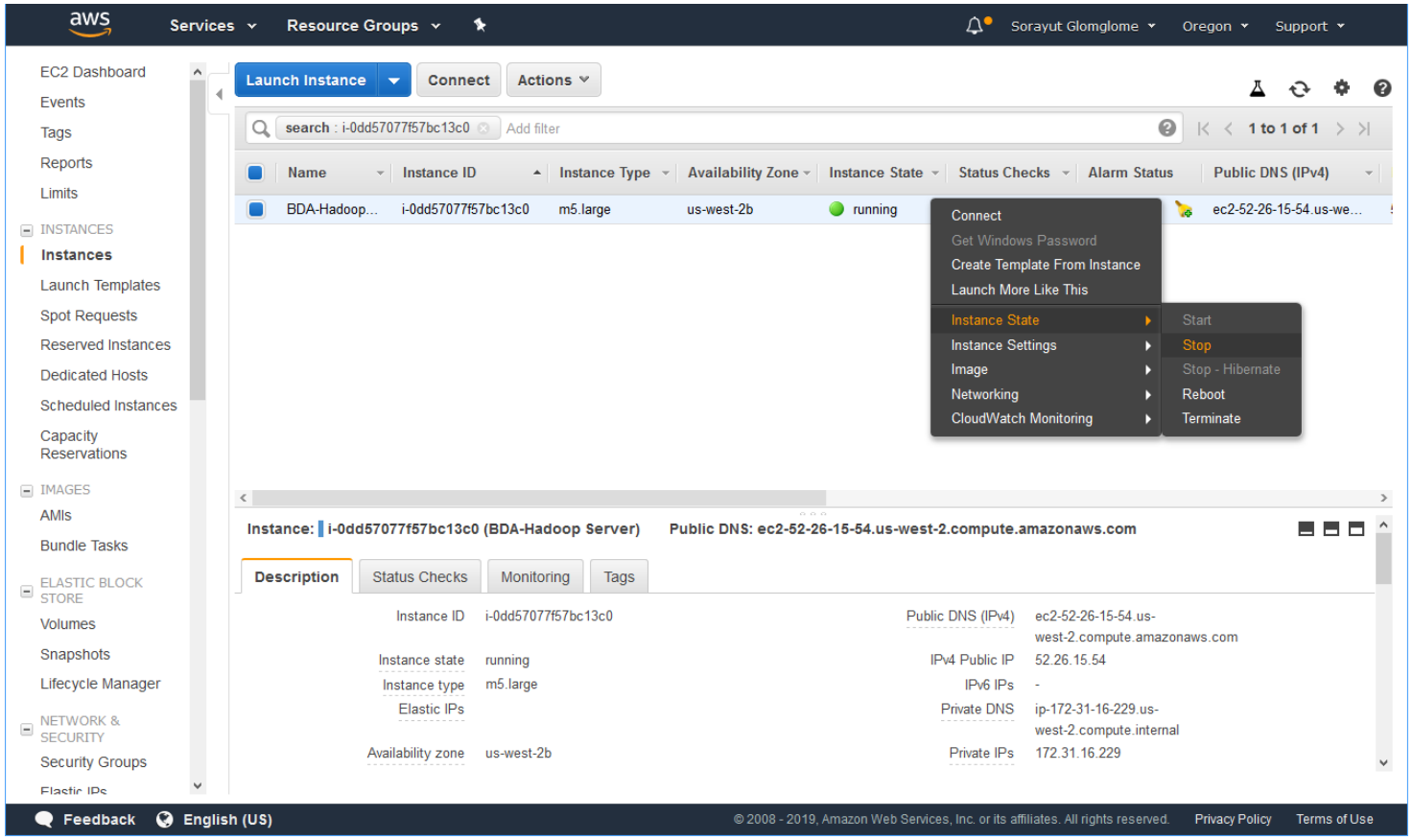
Stop DFS:

`$stop-dfs.sh`

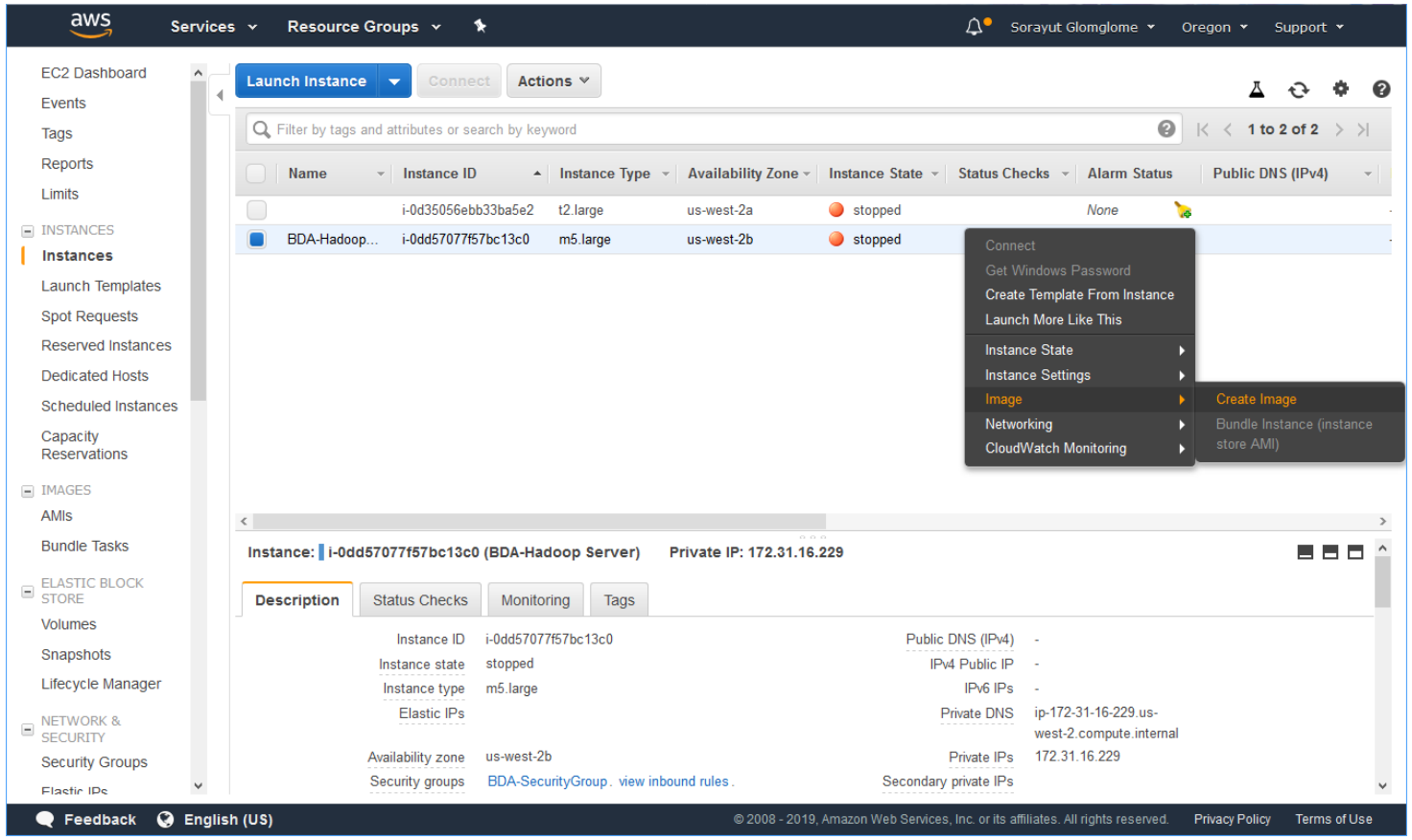


3. Create instance image

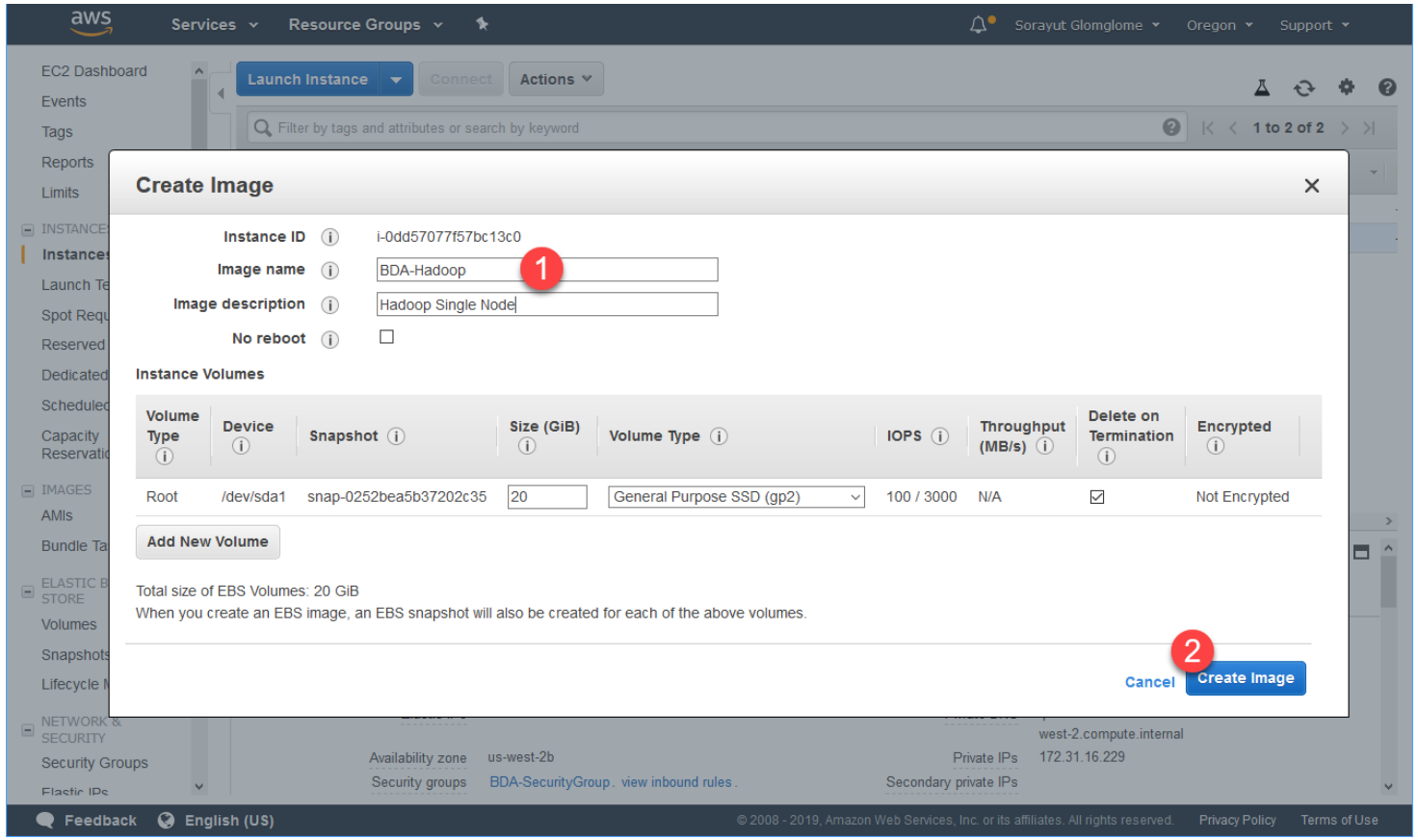
Stop instance:



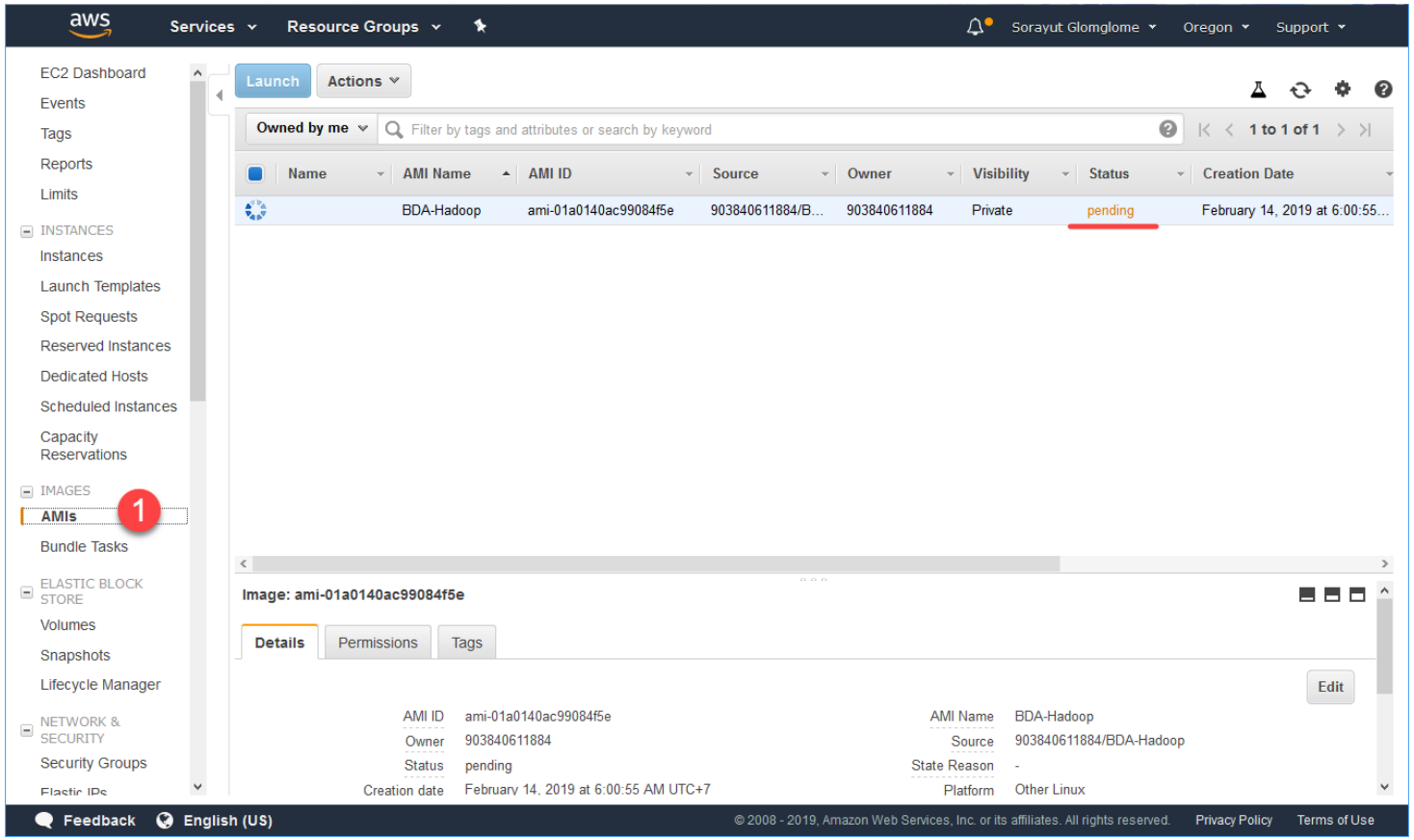
Create image:



Create image:



Waiting for creation:



Don't Forget to SUSPEND/TERMINATE Instance !!!!