Single Node Hadoop Cluster Setup

This document describes how to create Hadoop Single Node cluster in just 30 Minutes on Amazon EC2 cloud. You will learn following topics.

Click Here to watch these steps in Video Instructions

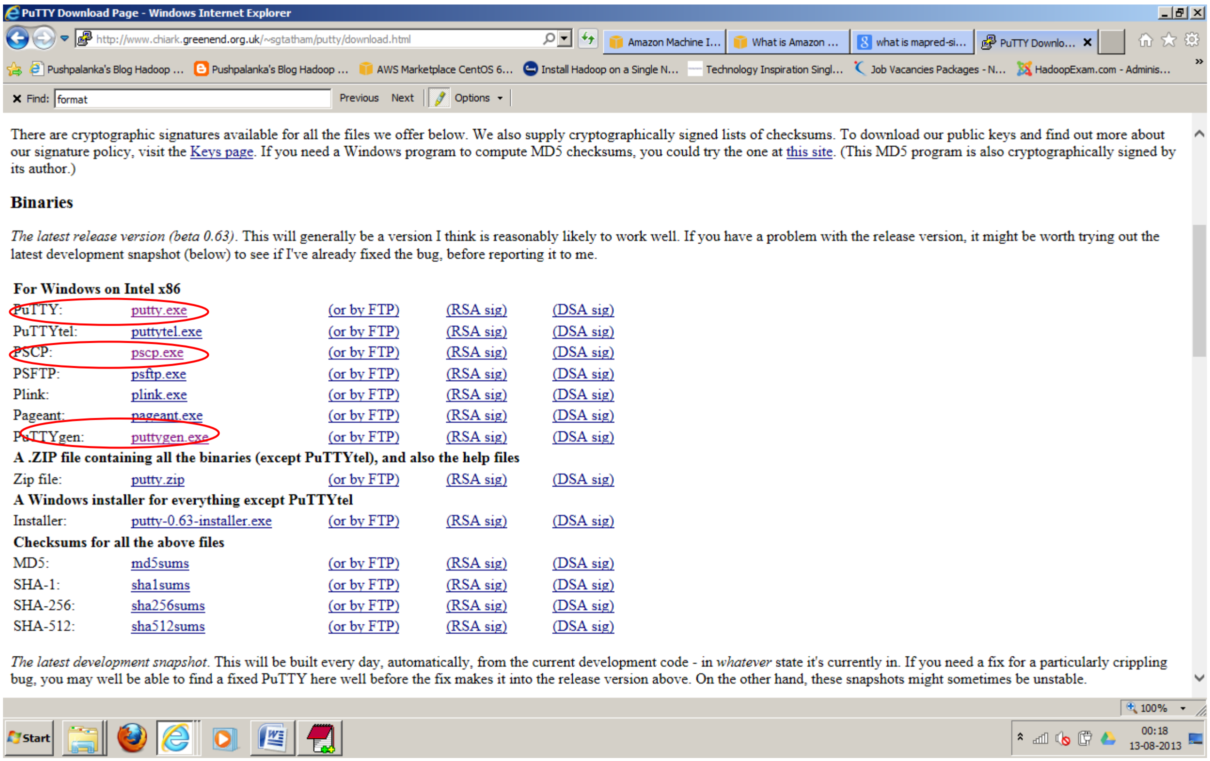
- How to create instance on Amazon EC2
- How to connect that Instance Using putty
- Installing Hadoop framework on this instance
- Run sample wordcount example which come with Hadoop framework.

Watch This Video for Full Instructions with example.

Following Software require on your local windows machine

1. **Putty**: To connect amazo ec2 instance.
2. **puttygen**: create private key from .ppm file
3. **pscp** : to copy file from your local filesytem to amazon instance

Download all three tools from http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html


PuTTY Download Page - Windows Internet Explorer
http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
Amazon Machine I...
What is Amazon ...
what is mapred-si...
PuTTY Downlo...
Pushpalanka's Blog Hadoop ...
Pushpalanka's Blog Hadoop ...
AWS Marketplace CentOS 6...
Install Hadoop on a Single N...
Technology Inspiration Singl...
Job Vacancies Packages - N...
HadoopExam.com - Adminis...
Find: format
Previous
Next
Options
100%
Start
00:18
13-08-2013

There are cryptographic signatures available for all the files we offer below. We also supply cryptographically signed lists of checksums. To download our public keys and find out more about our signature policy, visit the Keys page. If you need a Windows program to compute MD5 checksums, you could try the one at this site. (This MD5 program is also cryptographically signed by its author.)

Binaries

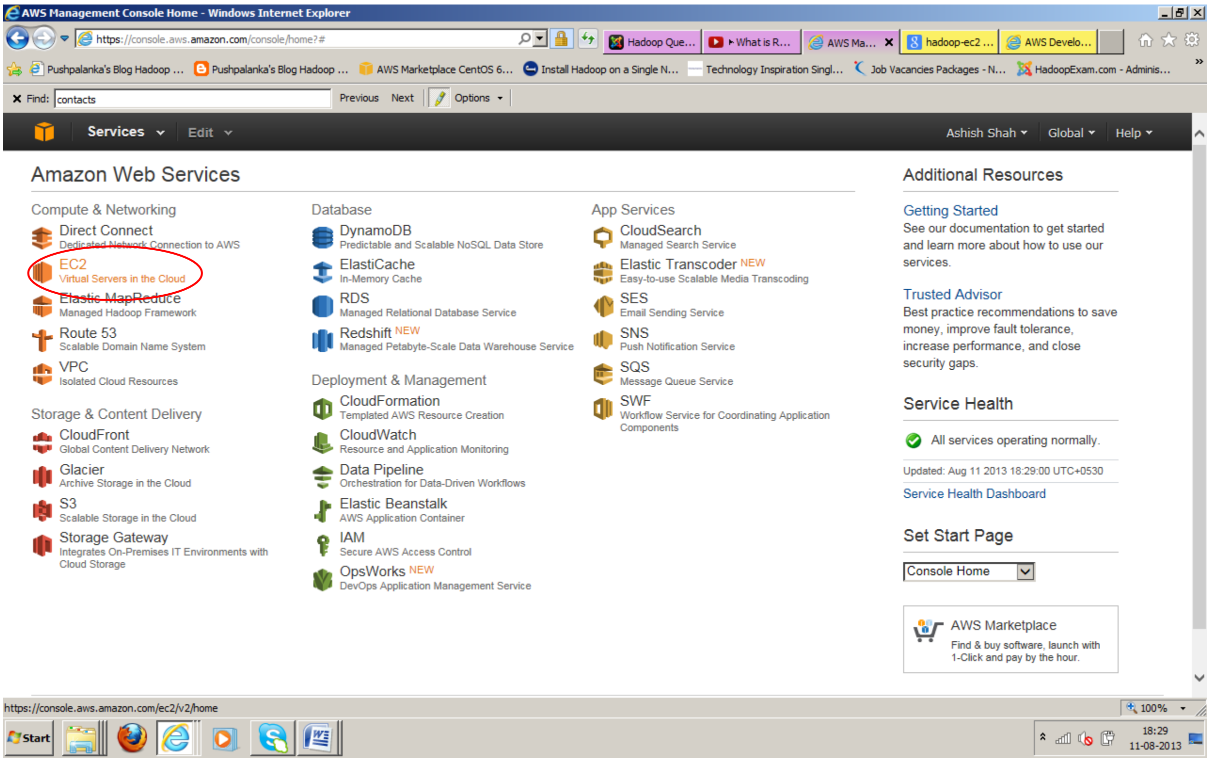
The latest release version (beta 0.63). This will generally be a version I think is reasonably likely to work well. If you have a problem with the release version, it might be worth trying out the latest development snapshot (below) to see if I've already fixed the bug, before reporting it to me.

For Windows on Intel x86				
PuTTY:	putty.exe	(or by FTP)	(RSA sig)	(DSA sig)
PuTTYtel:	puttytel.exe	(or by FTP)	(RSA sig)	(DSA sig)
PSCP:	pscp.exe	(or by FTP)	(RSA sig)	(DSA sig)
PSFTP:	psftp.exe	(or by FTP)	(RSA sig)	(DSA sig)
Plink:	plink.exe	(or by FTP)	(RSA sig)	(DSA sig)
Pageant:	pageant.exe	(or by FTP)	(RSA sig)	(DSA sig)
PuTTYgen:	puttygen.exe	(or by FTP)	(RSA sig)	(DSA sig)
A .ZIP file containing all the binaries (except PuTTYtel), and also the help files				
Zip file:	putty.zip	(or by FTP)	(RSA sig)	(DSA sig)
A Windows installer for everything except PuTTYtel				
Installer:	putty-0.63-installer.exe	(or by FTP)	(RSA sig)	(DSA sig)
Checksums for all the above files				
MD5:	md5sums	(or by FTP)	(RSA sig)	(DSA sig)
SHA-1:	sha1sums	(or by FTP)	(RSA sig)	(DSA sig)
SHA-256:	sha256sums	(or by FTP)	(RSA sig)	(DSA sig)
SHA-512:	sha512sums	(or by FTP)	(RSA sig)	(DSA sig)

The latest development snapshot. This will be built every day, automatically, from the current development code - in *whatever* state it's currently in. If you need a fix for a particularly crippling bug, you may well be able to find a fixed PuTTY here well before the fix makes it into the release version above. On the other hand, these snapshots might sometimes be unstable.

1. It requires you have Amazon AWS account. So create/signup Amazon EC2 account by going to http://aws.amazon.com/ . It also requires you to enter your credit card details. However, it would not charge until you use paid resources.

Click to Amzon EC2 Console under Compute & Networking


AWS Management Console Home - Windows Internet Explorer
https://console.aws.amazon.com/console/home?#
Hadoop Que...
What is R...
AWS Ma...
hadoop-ec2 ...
AWS Develo...
Pushpalanka's Blog Hadoop ...
Pushpalanka's Blog Hadoop ...
AWS Marketplace CentOS 6...
Install Hadoop on a Single N...
Technology Inspiration Singl...
Job Vacancies Packages - N...
HadoopExam.com - Adminis...
Find: contacts
Previous
Next
Options
Services
Edit
Ashish Shah
Global
Help
https://console.aws.amazon.com/ec2/v2/home
100%
Start
18:29
11-08-2013

Amazon Web Services

Compute & Networking

Direct Connect
Dedicated Network Connection to AWS

EC2
Virtual Servers in the Cloud

Elastic MapReduce
Managed Hadoop Framework

Route 53
Scalable Domain Name System

VPC
Isolated Cloud Resources

Storage & Content Delivery

CloudFront
Global Content Delivery Network

Glacier
Archive Storage in the Cloud

S3
Scalable Storage in the Cloud

Storage Gateway
Integrates On-Premises IT Environments with Cloud Storage

Database

DynamoDB
Predictable and Scalable NoSQL Data Store

ElastiCache
In-Memory Cache

RDS
Managed Relational Database Service

Redshift NEW
Managed Petabyte-Scale Data Warehouse Service

Deployment & Management

CloudFormation
Templated AWS Resource Creation

CloudWatch
Resource and Application Monitoring

Data Pipeline
Orchestration for Data-Driven Workflows

Elastic Beanstalk
AWS Application Container

IAM
Secure AWS Access Control

OpsWorks NEW
DevOps Application Management Service

App Services

CloudSearch
Managed Search Service

Elastic Transcoder NEW
Easy-to-use Scalable Media Transcoding

SES
Email Sending Service

SNS
Push Notification Service

SQS
Message Queue Service

SWF
Workflow Service for Coordinating Application Components

Additional Resources

Getting Started
See our documentation to get started and learn more about how to use our services.

Trusted Advisor
Best practice recommendations to save money, improve fault tolerance, increase performance, and close security gaps.

Service Health

All services operating normally.

Updated: Aug 11 2013 18:29:00 UTC+0530

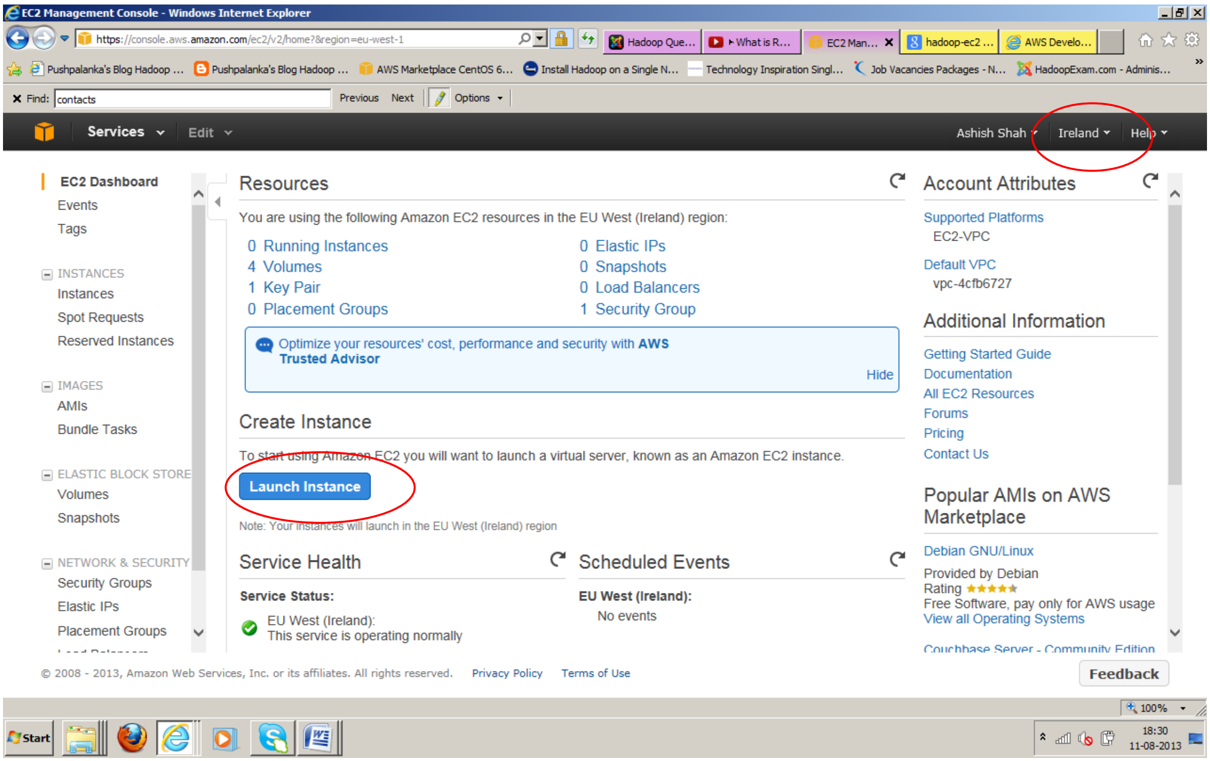
Service Health Dashboard

Set Start Page

Console Home

AWS Marketplace
Find & buy software, launch with 1-Click and pay by the hour.

2. Once you are in then Click Launch Instance (chose EU West Ireland region). This will create a Virtual Machine Instance in the cloud. And you have to provide the configuration which you can see in next steps.



Resources

You are using the following Amazon EC2 resources in the EU West (Ireland) region:

0 Running Instances
4 Volumes
1 Key Pair
0 Placement Groups
0 Elastic IPs
0 Snapshots
0 Load Balancers
1 Security Group

Optimize your resources' cost, performance and security with AWS Trusted Advisor

Hide

Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

Launch Instance

Note: Your instances will launch in the EU West (Ireland) region

Service Health

Service Status:

EU West (Ireland):
This service is operating normally

Scheduled Events

EU West (Ireland):
No events

Account Attributes

Supported Platforms
EC2-VPC

Default VPC
vpc-4cfb6727

Additional Information

Getting Started Guide
Documentation
All EC2 Resources
Forums
Pricing
Contact Us

Popular AMIs on AWS Marketplace

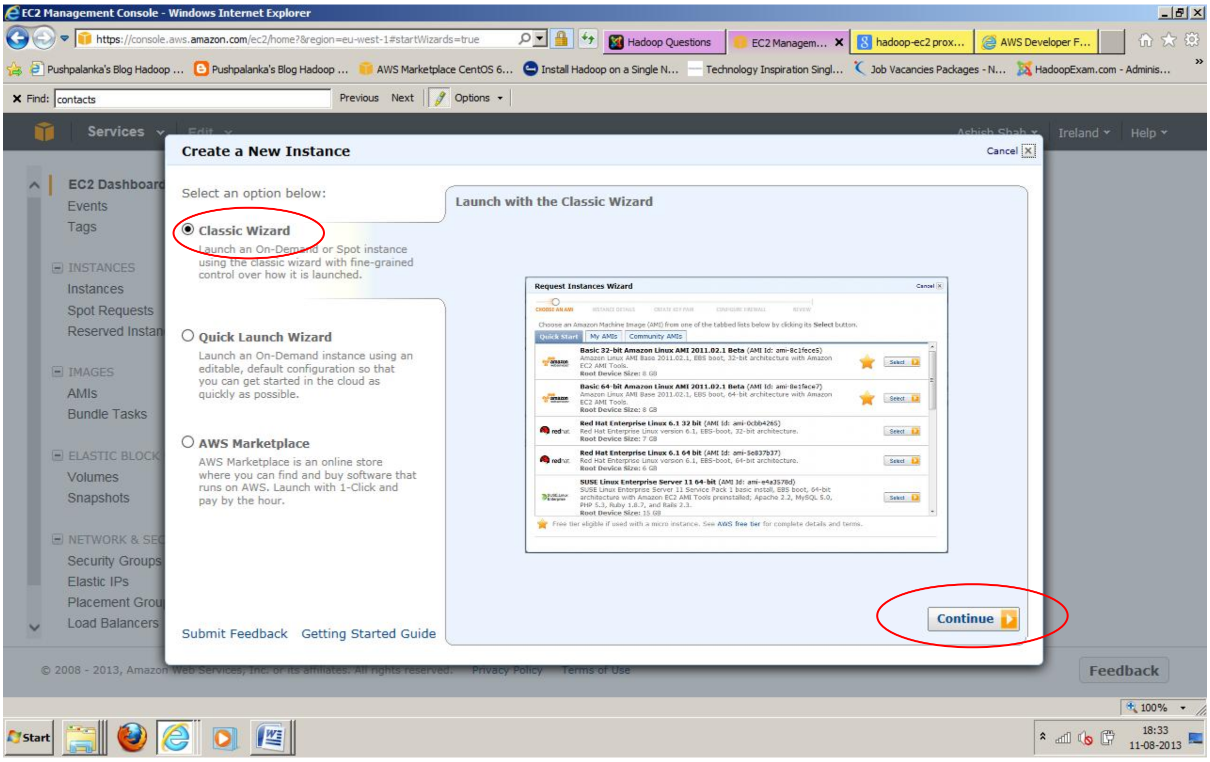
Debian GNU/Linux
Provided by Debian
Rating ★★★★★
Free Software, pay only for AWS usage
View all Operating Systems

Couchbase Server - Community Edition

© 2008 - 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Feedback

3. Select Classic Wizard and then press continue.



Create a New Instance Cancel ☒

Select an option below:

● **Classic Wizard**
Launch an On-Demand or Spot instance using the classic wizard with fine-grained control over how it is launched.

○ **Quick Launch Wizard**
Launch an On-Demand instance using an editable, default configuration so that you can get started in the cloud as quickly as possible.

○ **AWS Marketplace**
AWS Marketplace is an online store where you can find and buy software that runs on AWS. Launch with 1-Click and pay by the hour.

Launch with the Classic Wizard

Request Instances Wizard Cancel ☒

CHOOSE AN AMI · INSTANCE DETAILS · CREATE KEY PAIR · CONFIGURE FIREWALL · REVIEW

Choose an Amazon Machine Image (AMI) from one of the tabbed lists below by clicking its **Select** button.

Quick Start | My AMIs | Community AMIs

Basic 32-bit Amazon Linux AMI 2011.02.1 Beta (AMI Id: ami-8c1fece5)
Amazon Linux AMI Base 2011.02.1, EBS boot, 32-bit architecture with Amazon EC2 AMI Tools.
Root Device Size: 8 GB

Basic 64-bit Amazon Linux AMI 2011.02.1 Beta (AMI Id: ami-8e1fece7)
Amazon Linux AMI Base 2011.02.1, EBS boot, 64-bit architecture with Amazon EC2 AMI Tools.
Root Device Size: 8 GB

Red Hat Enterprise Linux 6.1 32 bit (AMI Id: ami-0cbb4265)
Red Hat Enterprise Linux version 6.1, EBS-boot, 32-bit architecture.
Root Device Size: 7 GB

Red Hat Enterprise Linux 6.1 64 bit (AMI Id: ami-5e837b37)
Red Hat Enterprise Linux version 6.1, EBS-boot, 64-bit architecture.
Root Device Size: 6 GB

SUSE Linux Enterprise Server 11 64-bit (AMI Id: ami-e4a3578d)
SUSE Linux Enterprise Server 11 Service Pack 1 basic install, EBS boot, 64-bit architecture with Amazon EC2 AMI Tools preinstalled; Apache 2.2, MySQL 5.0, PHP 5.3, Ruby 1.8.7, and Rails 2.3.
Root Device Size: 15 GB

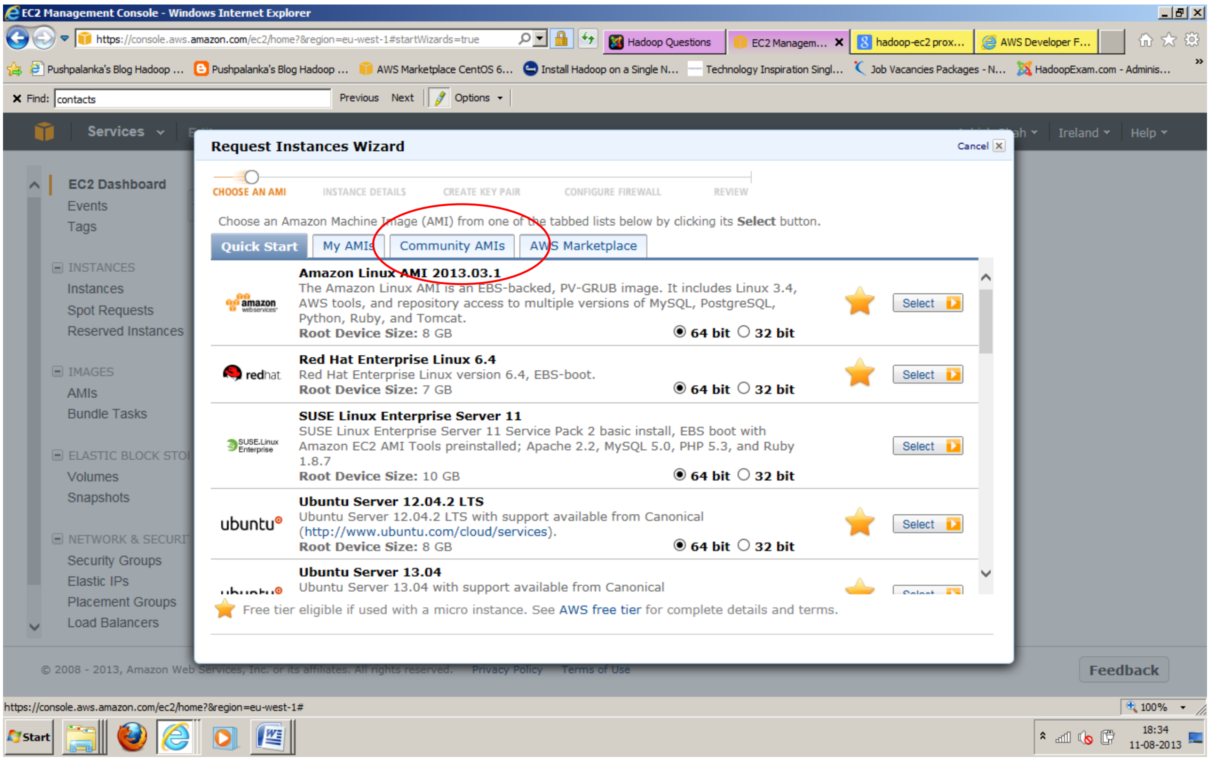
Free tier eligible if used with a micro instance. See AWS free tier for complete details and terms.

Continue

Submit Feedback   Getting Started Guide

© 2008 - 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved.   Privacy Policy   Terms of Use

4. Select Community AMI's
An Amazon Machine Image (AMI) is a special type of pre-configured operating system and virtual application software which is used to create a virtual machine within the Amazon Elastic Compute Cloud (EC2). It serves as the basic unit of deployment for services delivered using EC2.



Request Instances Wizard Cancel ☒

CHOOSE AN AMI · INSTANCE DETAILS · CREATE KEY PAIR · CONFIGURE FIREWALL · REVIEW

Choose an Amazon Machine Image (AMI) from one of the tabbed lists below by clicking its **Select** button.

Quick Start | My AMIs | Community AMIs | AWS Marketplace

Amazon Linux AMI 2013.03.1
The Amazon Linux AMI is an EBS-backed, PV-GRUB image. It includes Linux 3.4, AWS tools, and repository access to multiple versions of MySQL, PostgreSQL, Python, Ruby, and Tomcat.
Root Device Size: 8 GB   ● 64 bit ○ 32 bit   Select

Red Hat Enterprise Linux 6.4
Red Hat Enterprise Linux version 6.4, EBS-boot.
Root Device Size: 7 GB   ● 64 bit ○ 32 bit   Select

SUSE Linux Enterprise Server 11
SUSE Linux Enterprise Server 11 Service Pack 2 basic install, EBS boot with Amazon EC2 AMI Tools preinstalled; Apache 2.2, MySQL 5.0, PHP 5.3, and Ruby 1.8.7
Root Device Size: 10 GB   ● 64 bit ○ 32 bit   Select

Ubuntu Server 12.04.2 LTS
Ubuntu Server 12.04.2 LTS with support available from Canonical (http://www.ubuntu.com/cloud/services).
Root Device Size: 8 GB   ● 64 bit ○ 32 bit   Select

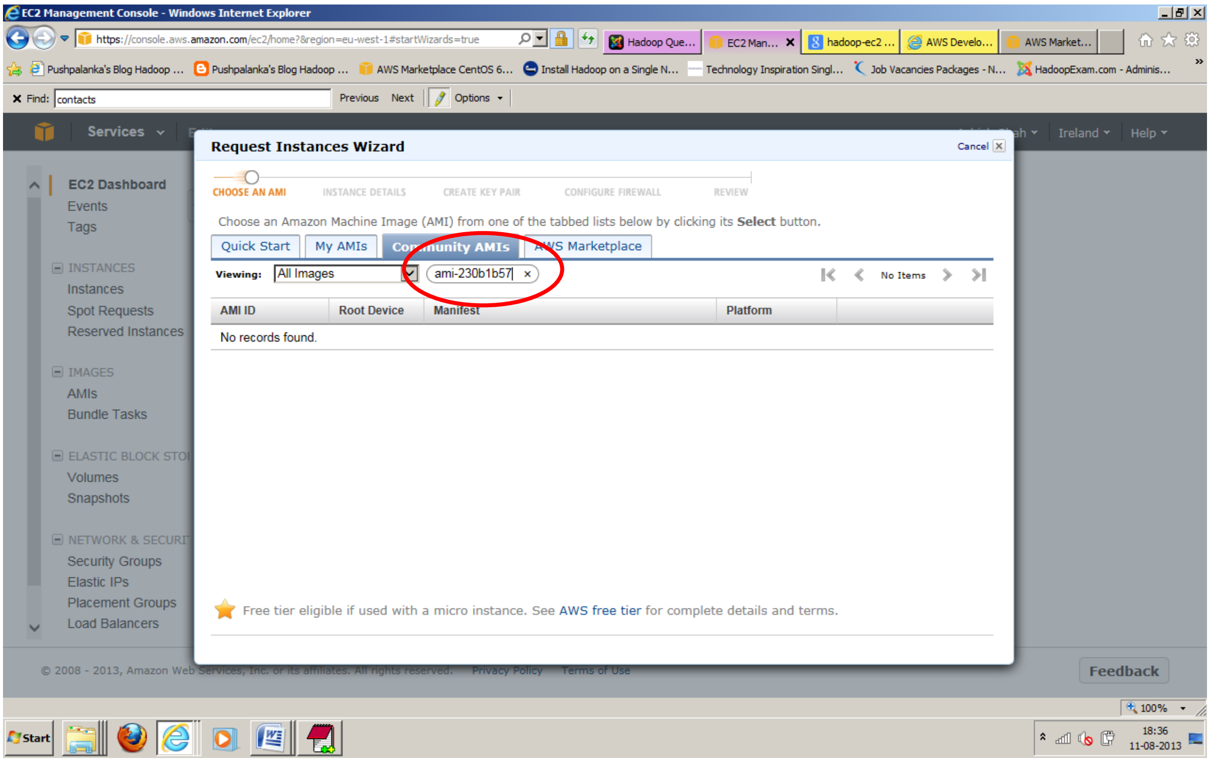
Ubuntu Server 13.04
Ubuntu Server 13.04 with support available from Canonical

Free tier eligible if used with a micro instance. See AWS free tier for complete details and terms.

© 2008 - 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved.   Privacy Policy   Terms of Use

https://console.aws.amazon.com/ec2/home?&region=eu-west-1#

5. Search for Community AMI: Now there are lot of pre-configured AMIs available in Amazon EC2 cloud. You can search in *AWS Marketplace* as well. We are choosing AMI's for CentOS linux 6.0 version and id for this is (***ami-230b1b57***)



Request Instances Wizard

Choose an Amazon Machine Image (AMI) from one of the tabbed lists below by clicking its **Select** button.

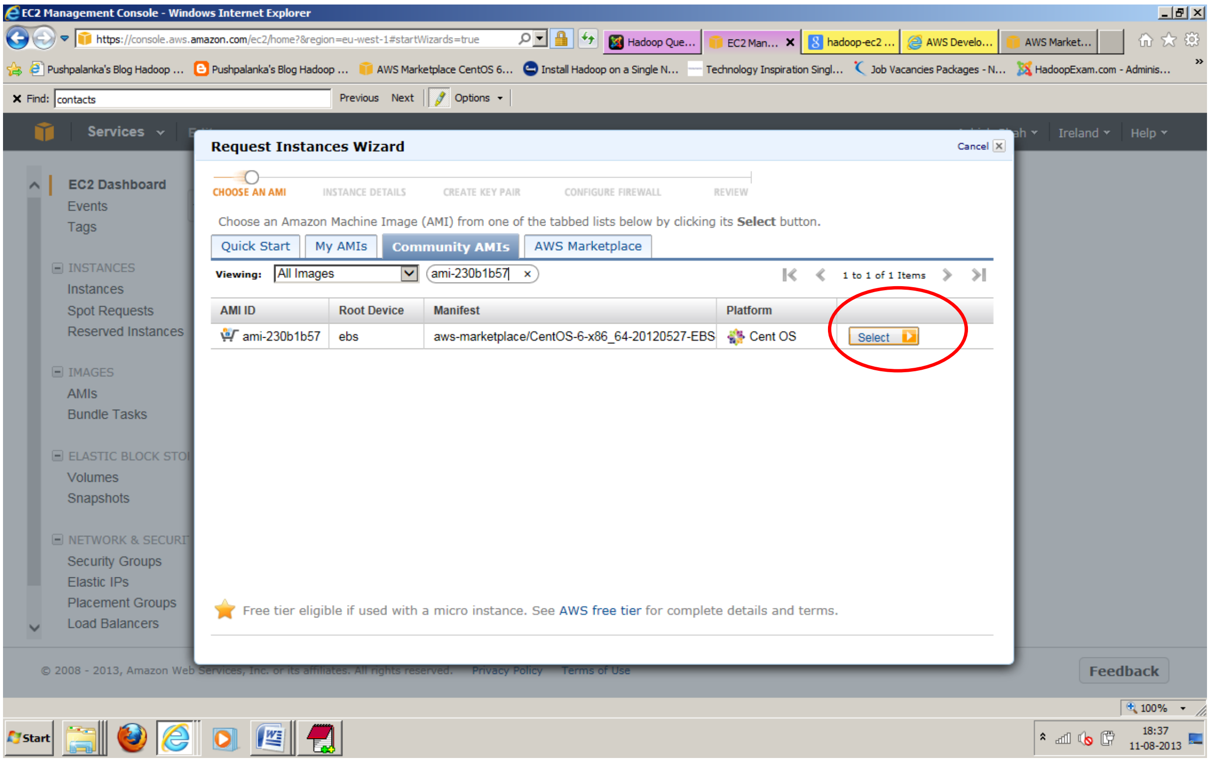
Viewing: All Images ami-230b1b57 No Items

AMI ID	Root Device	Manifest	Platform
No records found.			

Free tier eligible if used with a micro instance. See AWS free tier for complete details and terms.

© 2008 - 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

6. Select the AMI, it means you are configuring a virtual machine which will have CentOs linux installed.



Request Instances Wizard

Choose an Amazon Machine Image (AMI) from one of the tabbed lists below by clicking its **Select** button.

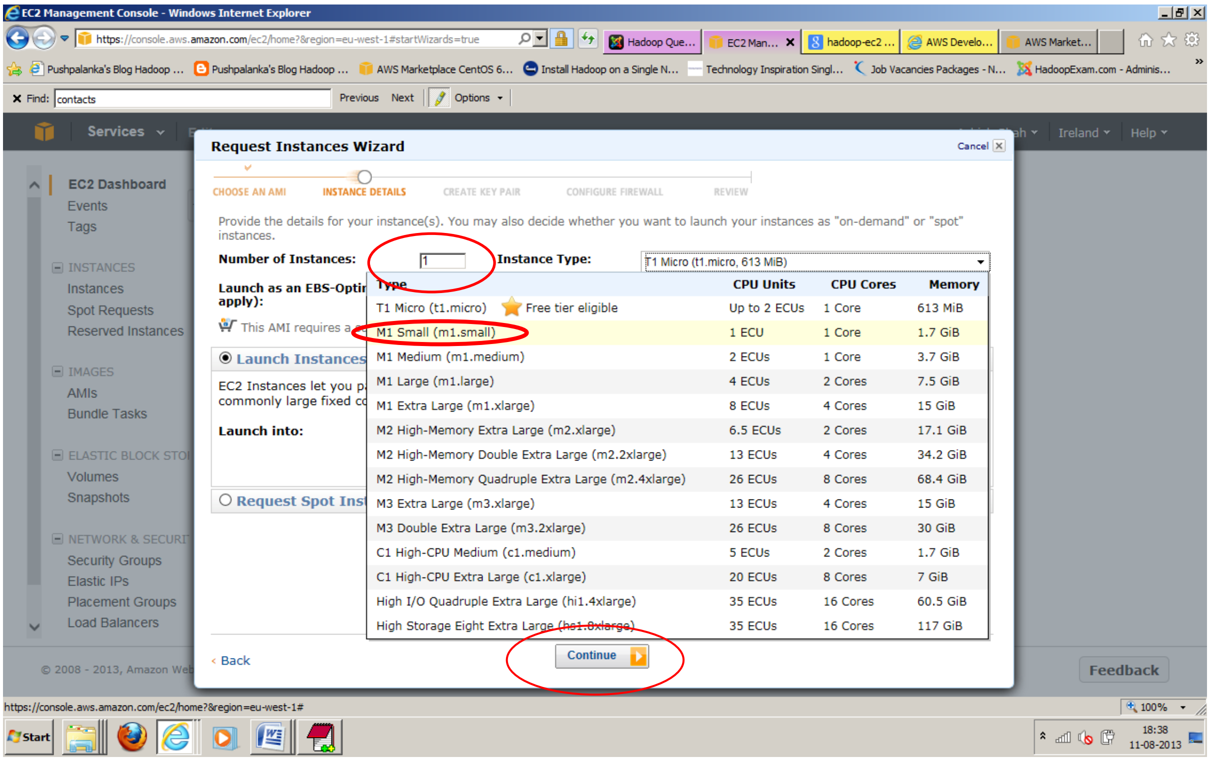
Viewing: All Images ami-230b1b57 1 to 1 of 1 Items

AMI ID	Root Device	Manifest	Platform	
ami-230b1b57	ebs	aws-marketplace/CentOS-6-x86_64-20120527-EBS	Cent OS	Select

Free tier eligible if used with a micro instance. See AWS free tier for complete details and terms.

© 2008 - 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

7. Now in this step we will decide how many instance of this virtual machine and type of the instance. We are going to create single node cluster hence select only 1 instance and choose Small Instance type which at least required for running Hadoop mapreduce example. You can choose micro instance which is completely free for 750 Hrs in a month, but that is not enough to run mapred example. However, if you are new to EC2 we suggest you try with micro instance first, so you would not incur any cost while configuring Hadoop cluster. And once you become confident with the configuration then you can start using the Small Instance for real practice. However, cost is very small approx.06$(Check Amazon for price) per Hour per Instance. And now click continue.


EC2 Management Console - Windows Internet Explorer
https://console.aws.amazon.com/ec2/home?&region=eu-west-1#startWizards=true
Hadoop Que...
EC2 Man...
hadoop-ec2 ...
AWS Develo...
AWS Market...
Pushpalanka's Blog Hadoop ...
Pushpalanka's Blog Hadoop ...
AWS Marketplace CentOS 6...
Install Hadoop on a Single N...
Technology Inspiration Singl...
Job Vacancies Packages - N...
HadoopExam.com - Adminis...
Find: contacts
Previous
Next
Options
Services
Ireland
Help
EC2 Dashboard
Events
Tags
INSTANCES
Instances
Spot Requests
Reserved Instances
IMAGES
AMIs
Bundle Tasks
ELASTIC BLOCK STOI
Volumes
Snapshots
NETWORK & SECURIT
Security Groups
Elastic IPs
Placement Groups
Load Balancers
© 2008 - 2013, Amazon Web
Feedback
https://console.aws.amazon.com/ec2/home?&region=eu-west-1#
100%
Start
18:38
11-08-2013

Request Instances Wizard

Cancel

CHOOSE AN AMI · INSTANCE DETAILS · CREATE KEY PAIR · CONFIGURE FIREWALL · REVIEW

Provide the details for your instance(s). You may also decide whether you want to launch your instances as "on-demand" or "spot" instances.

Number of Instances: 1

Instance Type: T1 Micro (t1.micro, 613 MiB)

Launch as an EBS-Optir apply):

This AMI requires a

● Launch Instances

EC2 Instances let you p commonly large fixed c

Launch into:

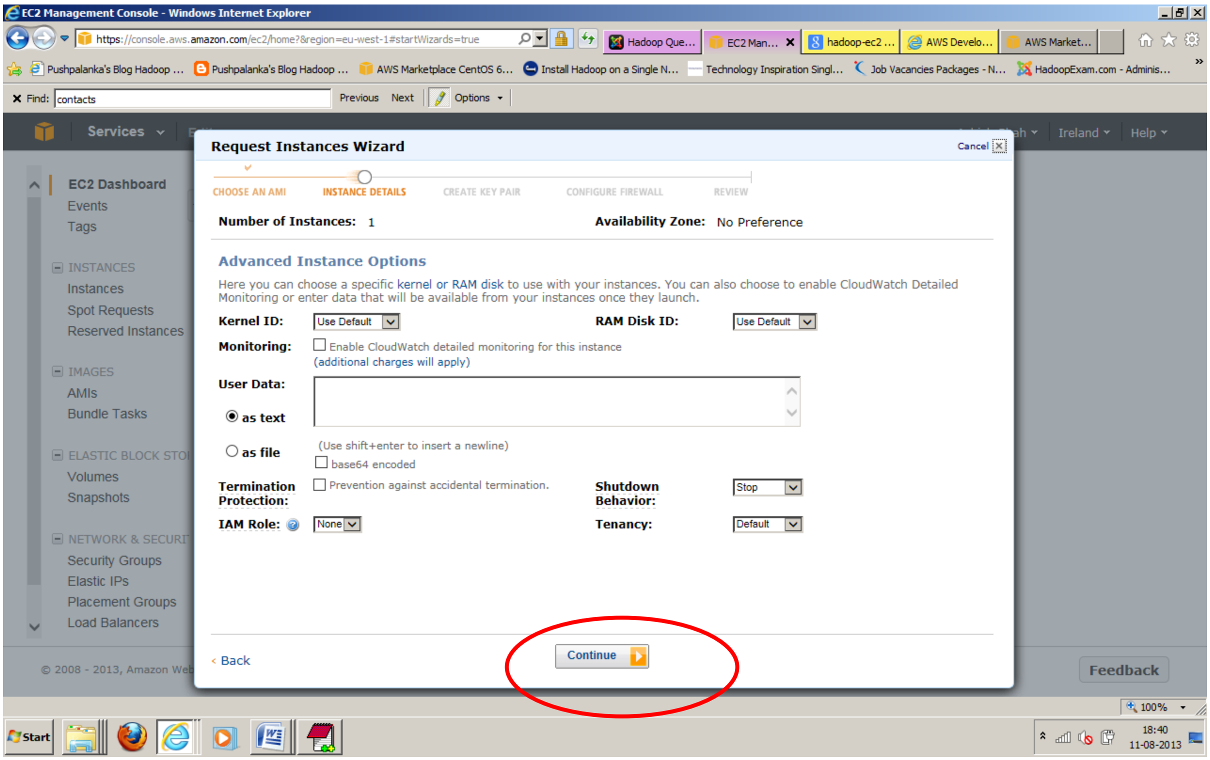
○ Request Spot Inst

Type	CPU Units	CPU Cores	Memory
T1 Micro (t1.micro) Free tier eligible	Up to 2 ECUs	1 Core	613 MiB
M1 Small (m1.small)	1 ECU	1 Core	1.7 GiB
M1 Medium (m1.medium)	2 ECUs	1 Core	3.7 GiB
M1 Large (m1.large)	4 ECUs	2 Cores	7.5 GiB
M1 Extra Large (m1.xlarge)	8 ECUs	4 Cores	15 GiB
M2 High-Memory Extra Large (m2.xlarge)	6.5 ECUs	2 Cores	17.1 GiB
M2 High-Memory Double Extra Large (m2.2xlarge)	13 ECUs	4 Cores	34.2 GiB
M2 High-Memory Quadruple Extra Large (m2.4xlarge)	26 ECUs	8 Cores	68.4 GiB
M3 Extra Large (m3.xlarge)	13 ECUs	4 Cores	15 GiB
M3 Double Extra Large (m3.2xlarge)	26 ECUs	8 Cores	30 GiB
C1 High-CPU Medium (c1.medium)	5 ECUs	2 Cores	1.7 GiB
C1 High-CPU Extra Large (c1.xlarge)	20 ECUs	8 Cores	7 GiB
High I/O Quadruple Extra Large (hi1.4xlarge)	35 ECUs	16 Cores	60.5 GiB
High Storage Eight Extra Large (hs1.8xlarge)	35 ECUs	16 Cores	117 GiB

‹ Back

Continue

8. Click Continue again, by keeping the default configuration.



Request Instances Wizard

Number of Instances: 1 Availability Zone: No Preference

Advanced Instance Options

Here you can choose a specific kernel or RAM disk to use with your instances. You can also choose to enable CloudWatch Detailed Monitoring or enter data that will be available from your instances once they launch.

Kernel ID: Use Default RAM Disk ID: Use Default

Monitoring: □ Enable CloudWatch detailed monitoring for this instance (additional charges will apply)

User Data:

● as text

○ as file

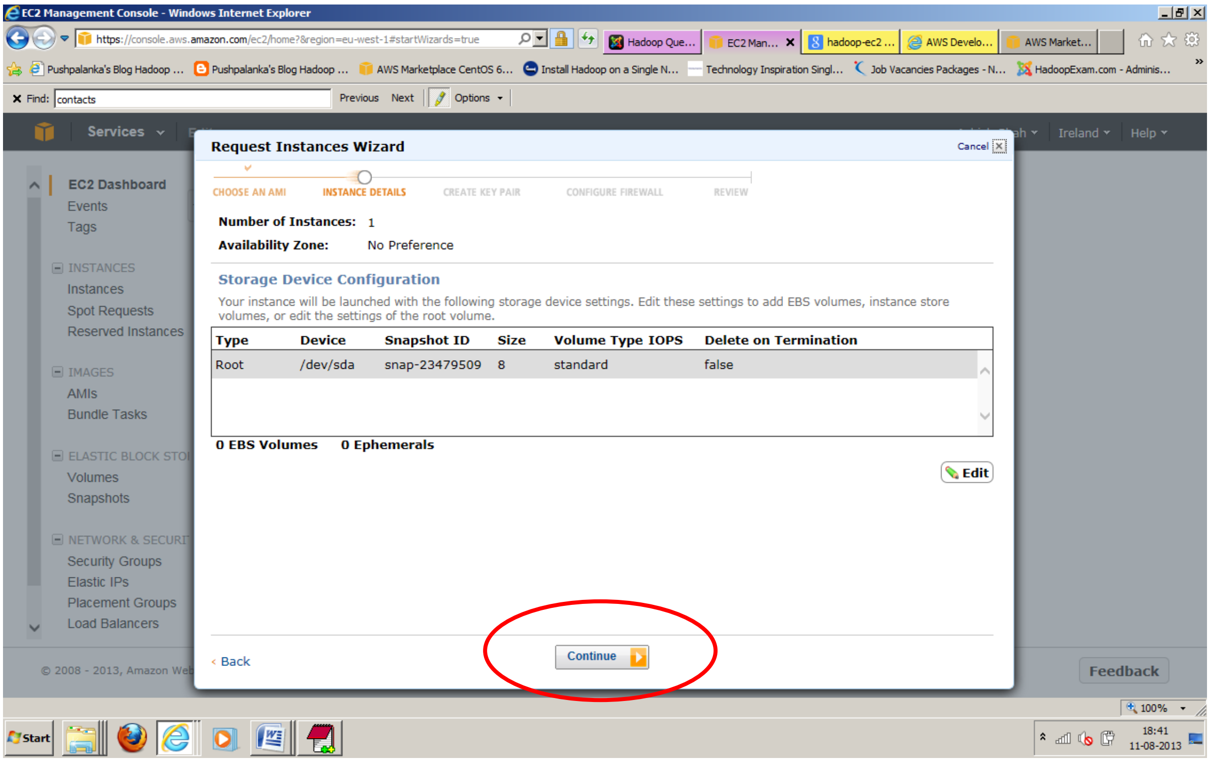
(Use shift+enter to insert a newline)

□ base64 encoded

Termination Protection: □ Prevention against accidental termination. Shutdown Behavior: Stop

IAM Role: None Tenancy: Default

9. Click Continue again, by keeping the default configuration



Request Instances Wizard

Number of Instances: 1

Availability Zone: No Preference

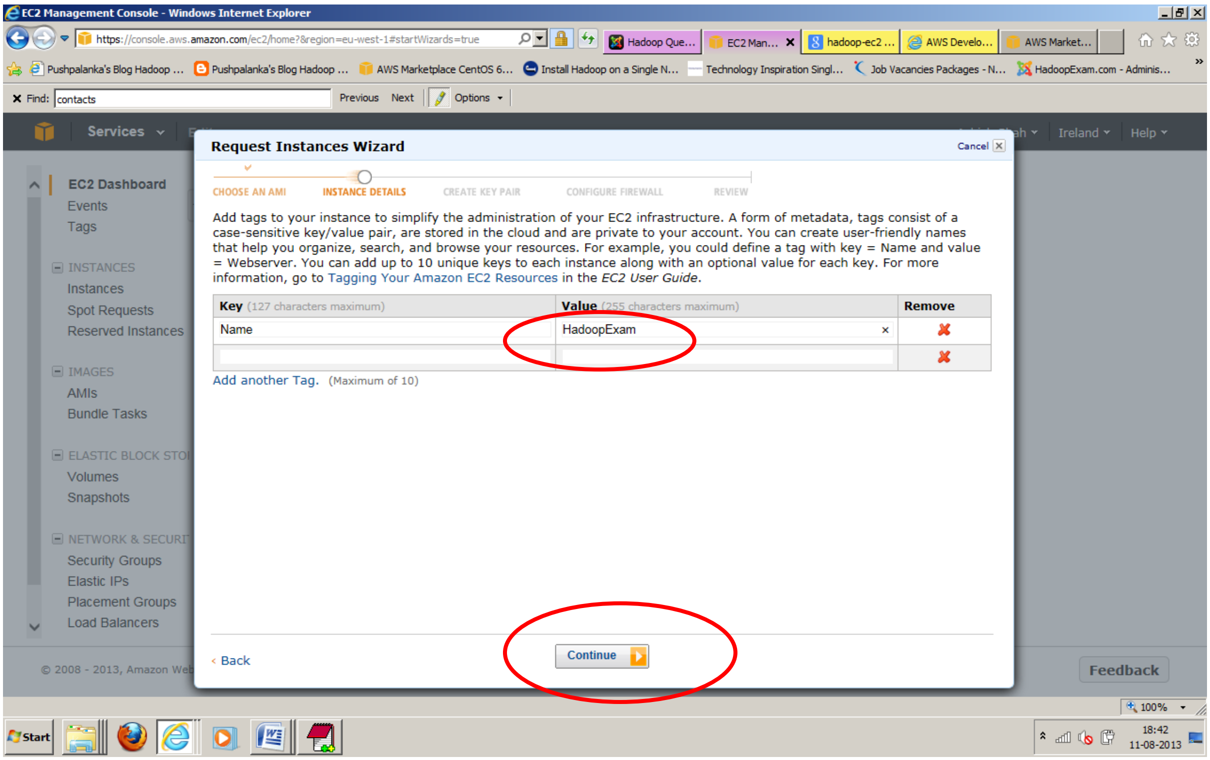
Storage Device Configuration

Your instance will be launched with the following storage device settings. Edit these settings to add EBS volumes, instance store volumes, or edit the settings of the root volume.

Type	Device	Snapshot ID	Size	Volume Type	IOPS	Delete on Termination
Root	/dev/sda	snap-23479509	8	standard		false

0 EBS Volumes 0 Ephemerals

10. Give Name to instance e.g. HadoopExam and click continue



Request Instances Wizard

Cancel

CHOOSE AN AMI · INSTANCE DETAILS · CREATE KEY PAIR · CONFIGURE FIREWALL · REVIEW

Add tags to your instance to simplify the administration of your EC2 infrastructure. A form of metadata, tags consist of a case-sensitive key/value pair, are stored in the cloud and are private to your account. You can create user-friendly names that help you organize, search, and browse your resources. For example, you could define a tag with key = Name and value = Webserver. You can add up to 10 unique keys to each instance along with an optional value for each key. For more information, go to Tagging Your Amazon EC2 Resources in the *EC2 User Guide*.

Key (127 characters maximum)	**Value** (255 characters maximum)	**Remove**
Name	HadoopExam	✗
		✗

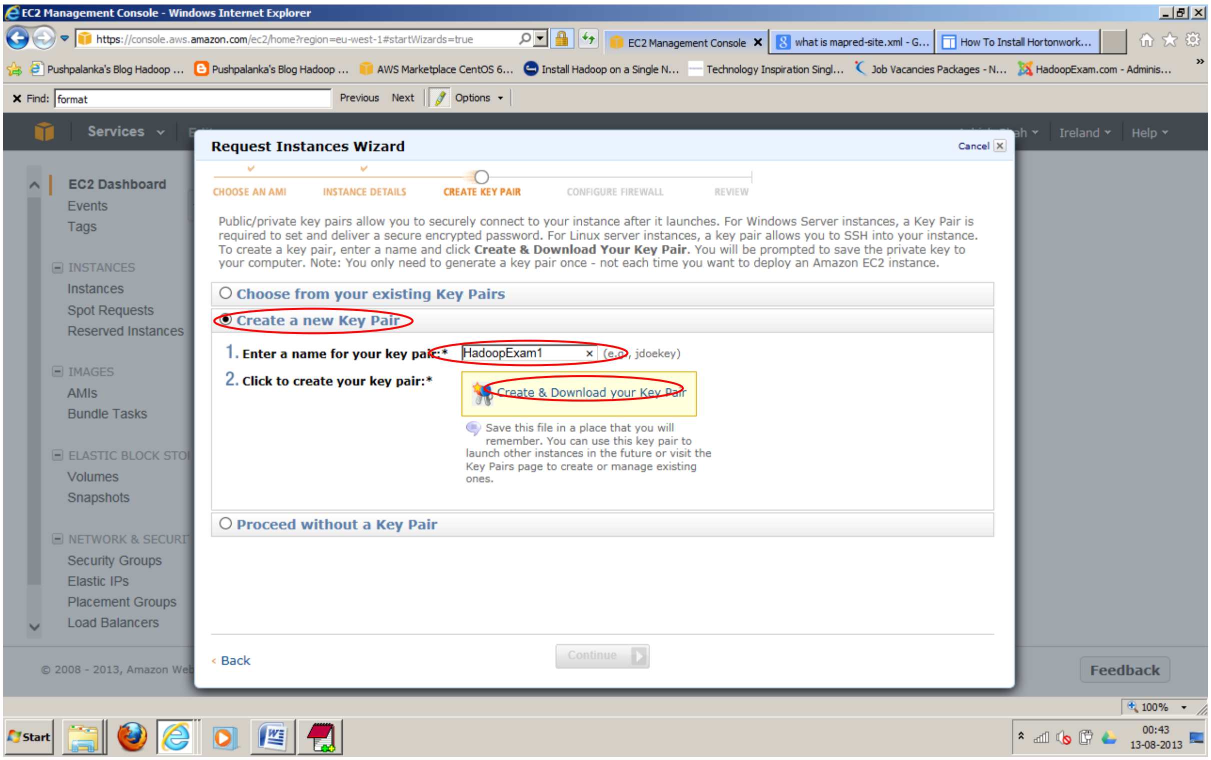
Add another Tag. (Maximum of 10)

‹ Back   Continue

11. Add the steps for Key Generation : Create New Key Pair : this key pair used for all instances named: **hadoopexam1.pem**
Download **hadoopexam1.pem** (DO NOT LOSE THIS)

- DON’T FORGET: download .pem file to your local machine when creating a new key pair (hadoopexam1.pem). The .pem (private key) file allows your client machine to connect to the running Amazon EC2 instance through SSH.If you lose the .pem you will need to re-create the instance, Amazon doesn’t store this file because of security reasons.

However you can stop, snapshot, and re-create a new instance based on this one so you don’t lose your configuration



Request Instances Wizard

CHOOSE AN AMI — INSTANCE DETAILS — CREATE KEY PAIR — CONFIGURE FIREWALL — REVIEW

Public/private key pairs allow you to securely connect to your instance after it launches. For Windows Server instances, a Key Pair is required to set and deliver a secure encrypted password. For Linux server instances, a key pair allows you to SSH into your instance. To create a key pair, enter a name and click **Create & Download Your Key Pair**. You will be prompted to save the private key to your computer. Note: You only need to generate a key pair once - not each time you want to deploy an Amazon EC2 instance.

○ Choose from your existing Key Pairs

● Create a new Key Pair

1. Enter a name for your key pair:* HadoopExam1 (e.g., jdoekey)
2. Click to create your key pair:* Create & Download your Key Pair

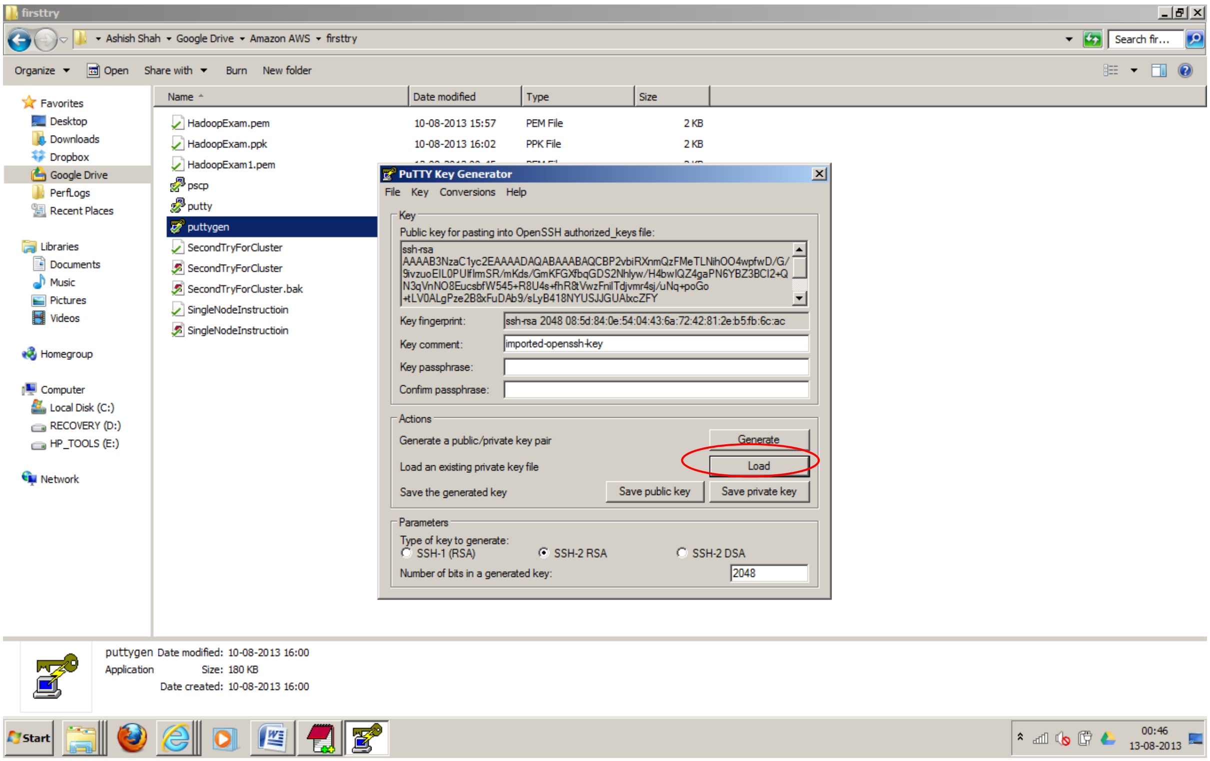
Save this file in a place that you will remember. You can use this key pair to launch other instances in the future or visit the Key Pairs page to create or manage existing ones.

○ Proceed without a Key Pair

‹ Back | Continue

12. Create .ppk file for Putty SSH client

§ Open PuttyGen, and Click Conversions > Import Key or load



Name	Date modified	Type	Size
HadoopExam.pem	10-08-2013 15:57	PEM File	2 KB
HadoopExam.ppk	10-08-2013 16:02	PPK File	2 KB
HadoopExam1.pem			
pscp			
putty			
puttygen			
SecondTryForCluster			
SecondTryForCluster			
SecondTryForCluster.bak			
SingleNodeInstructioin			
SingleNodeInstructioin			

PuTTY Key Generator

File Key Conversions Help

Key

Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQCBP2vbiRXnmQzFMeTLNihOO4wpfwD/G/
9ivzuoEIL0PUlflmSR/mKds/GmKFGXfbqGDS2Nhlyw/H4bwIQZ4gaPN6YBZ3BCI2+Q
N3qVnNO8EucsbfW545+R8U4s+fhR8tVwzFnilTdjvmr4sj/uNq+poGo
+tLV0ALgPze2B8xFuDAb9/sLyB418NYUSJJGUAlxcZFY
```

Key fingerprint: ssh-rsa 2048 08:5d:84:0e:54:04:43:6a:72:42:81:2e:b5:fb:6c:ac

Key comment: imported-openssh-key

Key passphrase:

Confirm passphrase:

Actions

Generate a public/private key pair — Generate

Load an existing private key file — Load

Save the generated key — Save public key | Save private key

Parameters

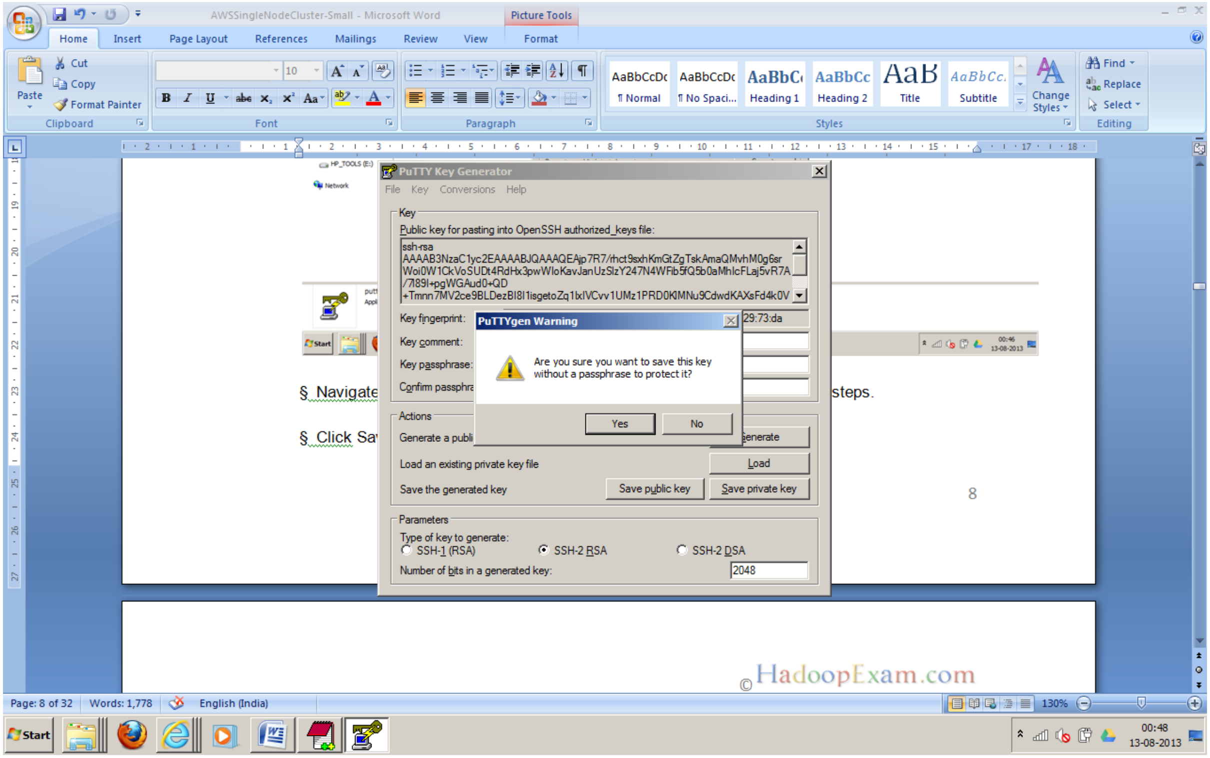
Type of key to generate: ○ SSH-1 (RSA) ● SSH-2 RSA ○ SSH-2 DSA

Number of bits in a generated key: 2048

puttygen Date modified: 10-08-2013 16:00
Application Size: 180 KB
Date created: 10-08-2013 16:00

§ Navigate and select **hadoopexam1.pem** that was created in previous steps.

§ Click Save, no passphrase, as: **hadoopexam1.ppk**



PuTTY Key Generator

Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAABJQAAAQEAjp7R7/rhct9sxhKmGtZgTskAmaQMvhM0g6sr
Woi0W1CkVoSUDt4RdHx3pwWloKavJanUzSlzY247N4WFib5fQ5b0aMhlcFLaj5vR7A
/7l89l+pgWGAud0+QD
+Tmnn7MV2ce9BLDezBI8l1isgetoZq1lxIVCvv1UMz1PRD0KlMNu9CdwdKAXsFd4k0V
```

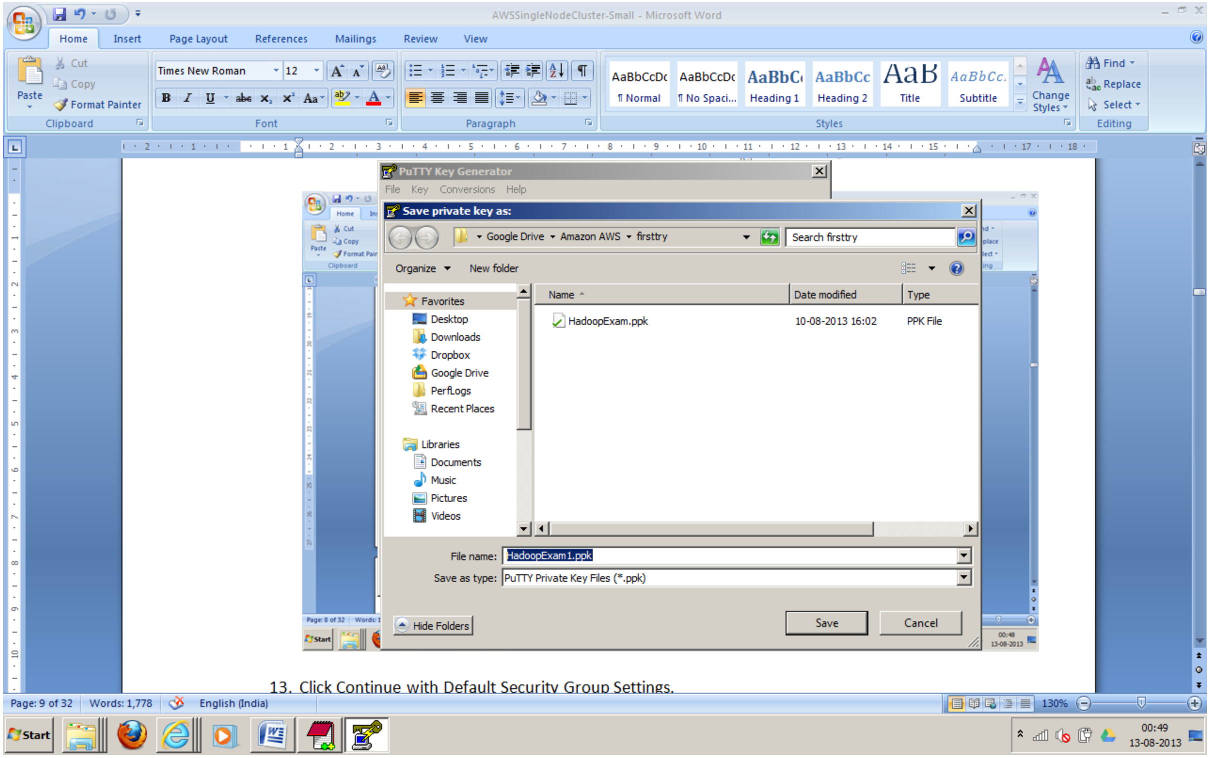
Key fingerprint: ...29:73:da

PuTTYgen Warning

Are you sure you want to save this key without a passphrase to protect it?

Type of key to generate: ○ SSH-1 (RSA) ● SSH-2 RSA ○ SSH-2 DSA

Number of bits in a generated key: 2048



Save private key as:

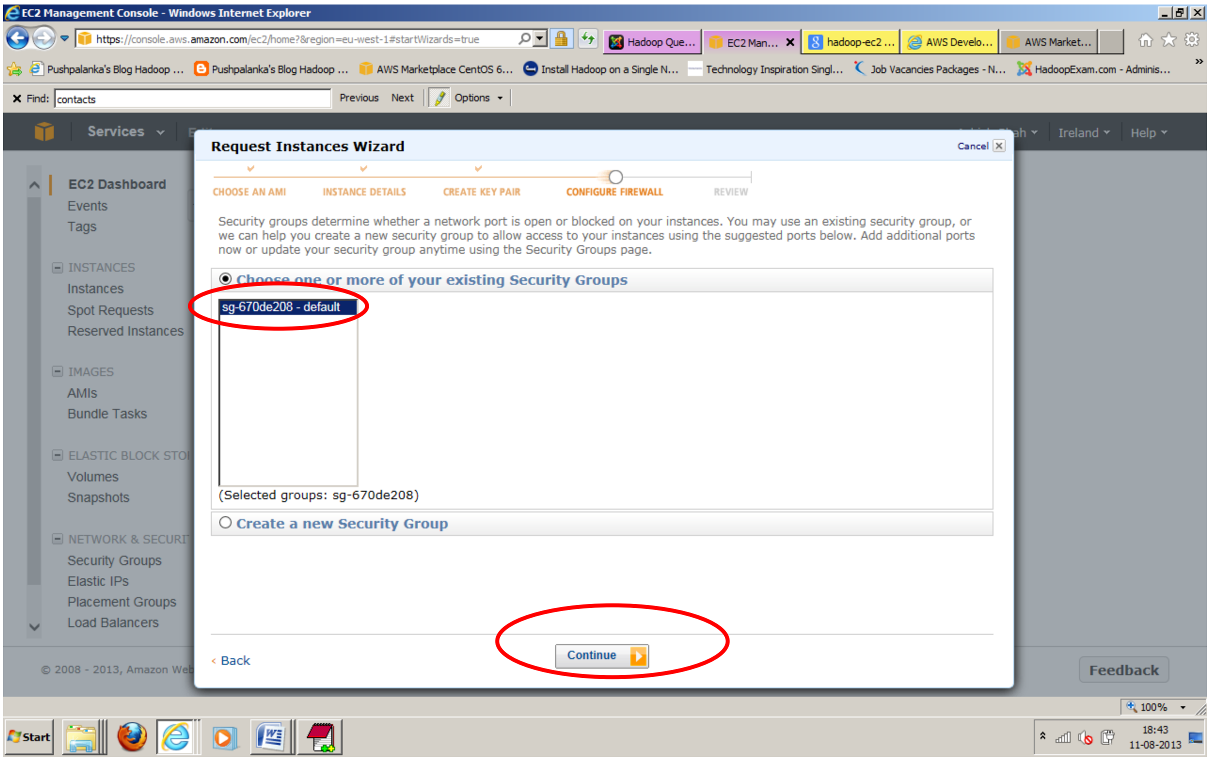
Google Drive ▸ Amazon AWS ▸ firsttry

Name	Date modified	Type
HadoopExam.ppk	10-08-2013 16:02	PPK File

File name: HadoopExam1.ppk

Save as type: PuTTY Private Key Files (*.ppk)

13. Click Continue with Default Security Group Settings.



Request Instances Wizard

Security groups determine whether a network port is open or blocked on your instances. You may use an existing security group, or we can help you create a new security group to allow access to your instances using the suggested ports below. Add additional ports now or update your security group anytime using the Security Groups page.

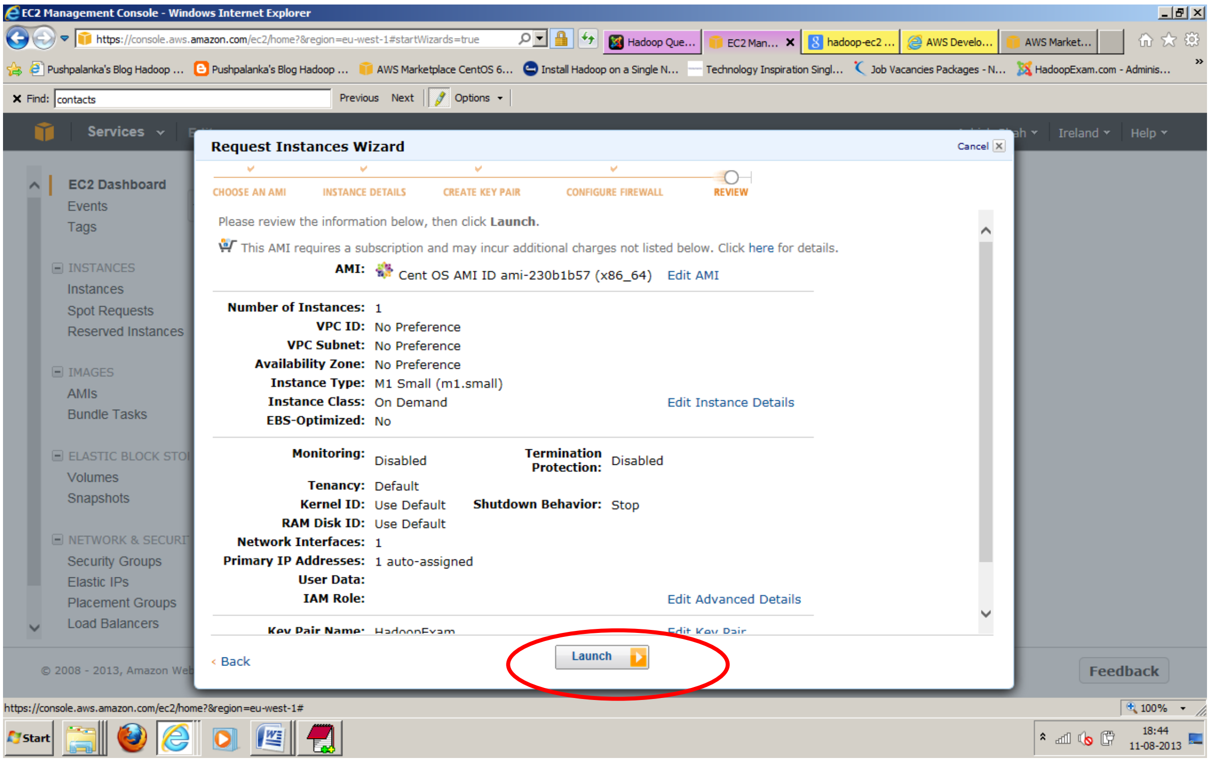
● Choose one or more of your existing Security Groups

sg-670de208 - default

(Selected groups: sg-670de208)

○ Create a new Security Group

14. Review and then click Launch, which will create the instance based on your configuration. This is the good place to verify your configuration.



Request Instances Wizard

Please review the information below, then click **Launch**.

This AMI requires a subscription and may incur additional charges not listed below. Click here for details.

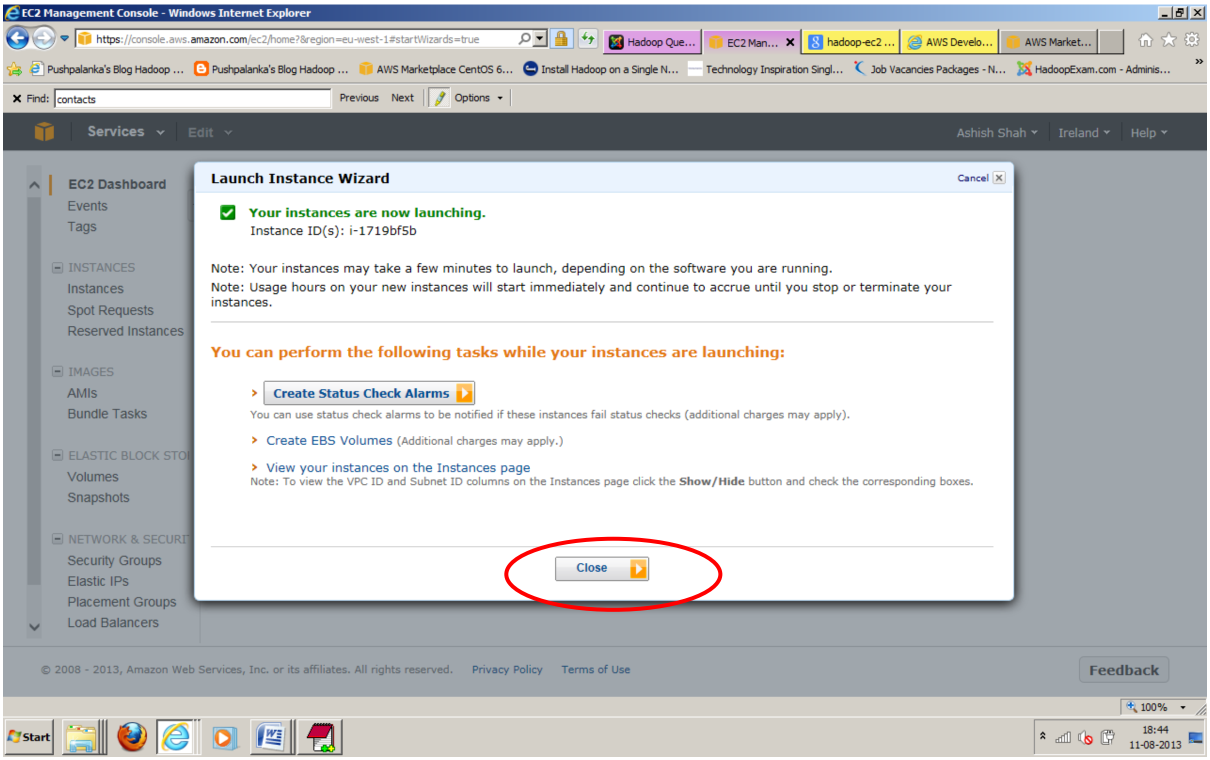
AMI: Cent OS AMI ID ami-230b1b57 (x86_64)

Number of Instances: 1
VPC ID: No Preference
VPC Subnet: No Preference
Availability Zone: No Preference
Instance Type: M1 Small (m1.small)
Instance Class: On Demand
EBS-Optimized: No

Monitoring: Disabled
Termination Protection: Disabled
Tenancy: Default
Kernel ID: Use Default
Shutdown Behavior: Stop
RAM Disk ID: Use Default
Network Interfaces: 1
Primary IP Addresses: 1 auto-assigned
User Data:
IAM Role:

Key Pair Name: HadoopExam

15. Click the close button



Launch Instance Wizard

Your instances are now launching.
Instance ID(s): i-1719bf5b

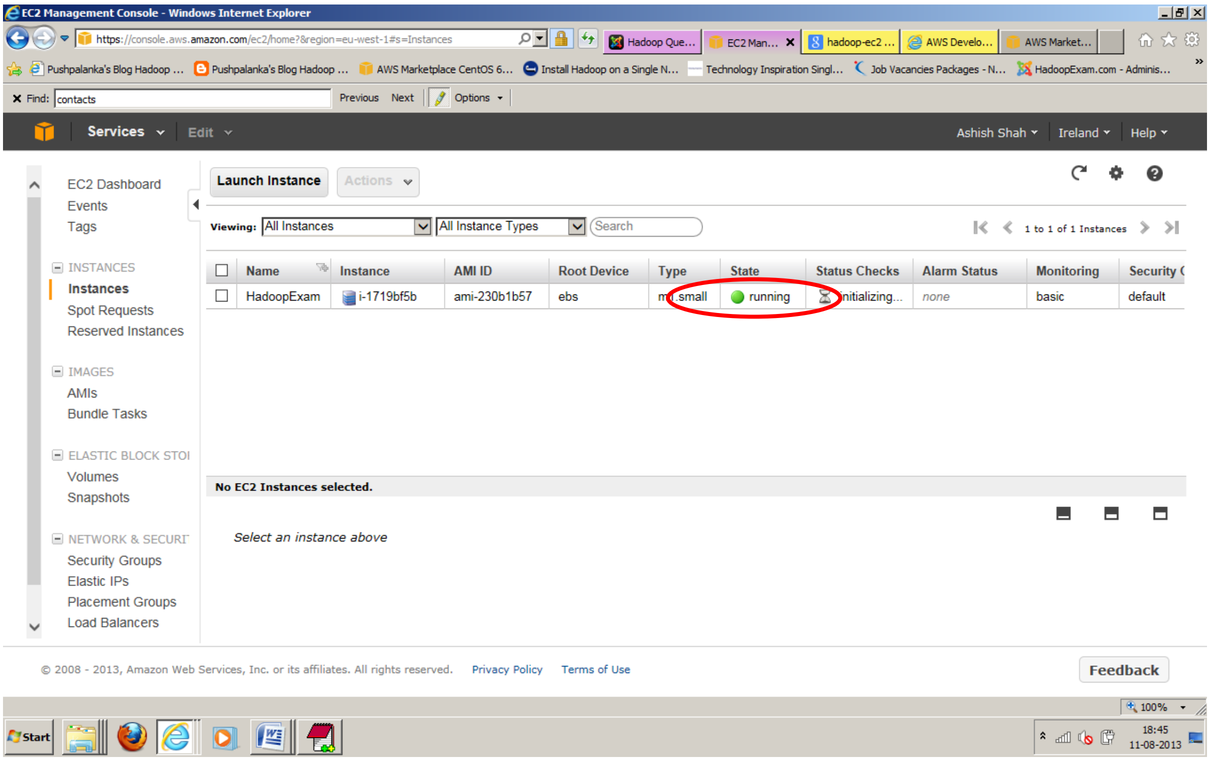
Note: Your instances may take a few minutes to launch, depending on the software you are running.
Note: Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

You can perform the following tasks while your instances are launching:

- Create Status Check Alarms
  You can use status check alarms to be notified if these instances fail status checks (additional charges may apply).
- Create EBS Volumes (Additional charges may apply.)
- View your instances on the Instances page
  Note: To view the VPC ID and Subnet ID columns on the Instances page click the Show/Hide button and check the corresponding boxes.

© 2008 - 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

16. Below is the Instance Detail and it shows the state that it is running.



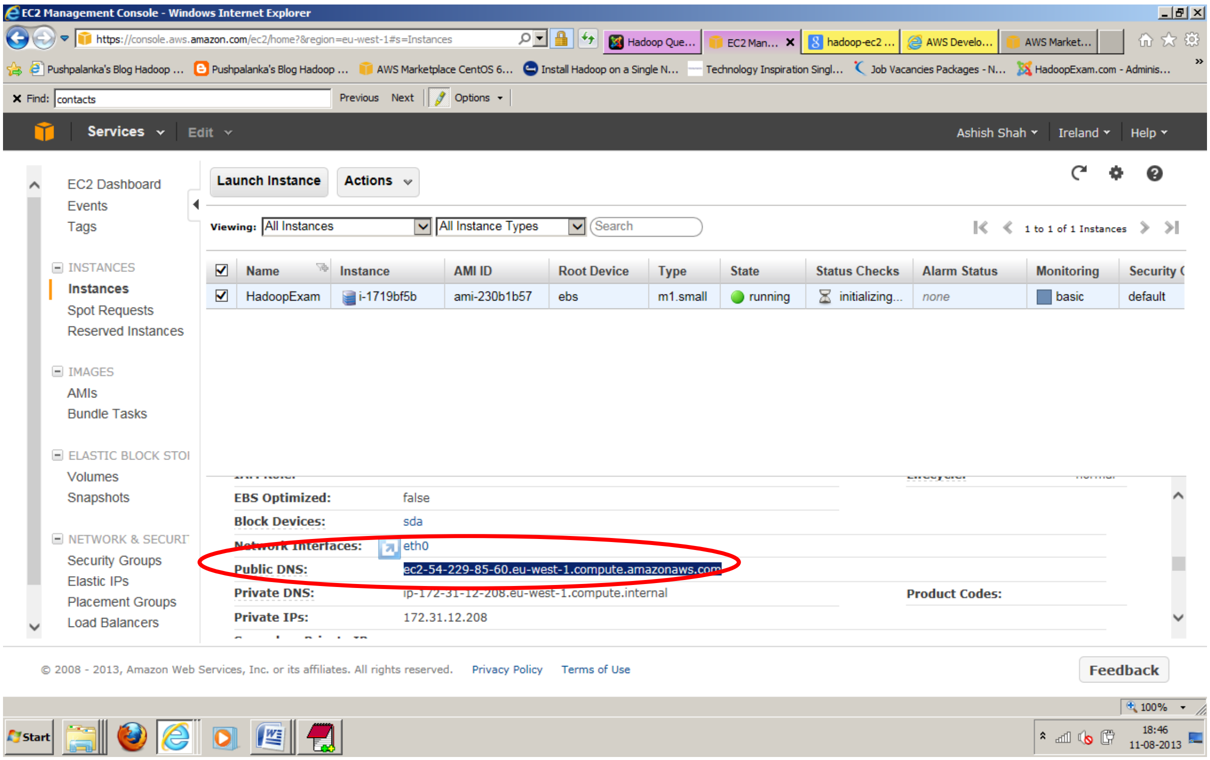
	Name	Instance	AMI ID	Root Device	Type	State	Status Checks	Alarm Status	Monitoring	Security (
□	HadoopExam	i-1719bf5b	ami-230b1b57	ebs	m1.small	● running	initializing...	none	basic	default

No EC2 Instances selected.

Select an instance above

© 2008 - 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

17. Copy the Public DNS somewhere in notepad for future use and this is URL by which you will access your instance, which you have created using the putty.



☑	Name	Instance	AMI ID	Root Device	Type	State	Status Checks	Alarm Status	Monitoring	Security (
☑	HadoopExam	i-1719bf5b	ami-230b1b57	ebs	m1.small	● running	initializing...	none	basic	default

EBS Optimized: false

Block Devices: sda

Network Interfaces: eth0

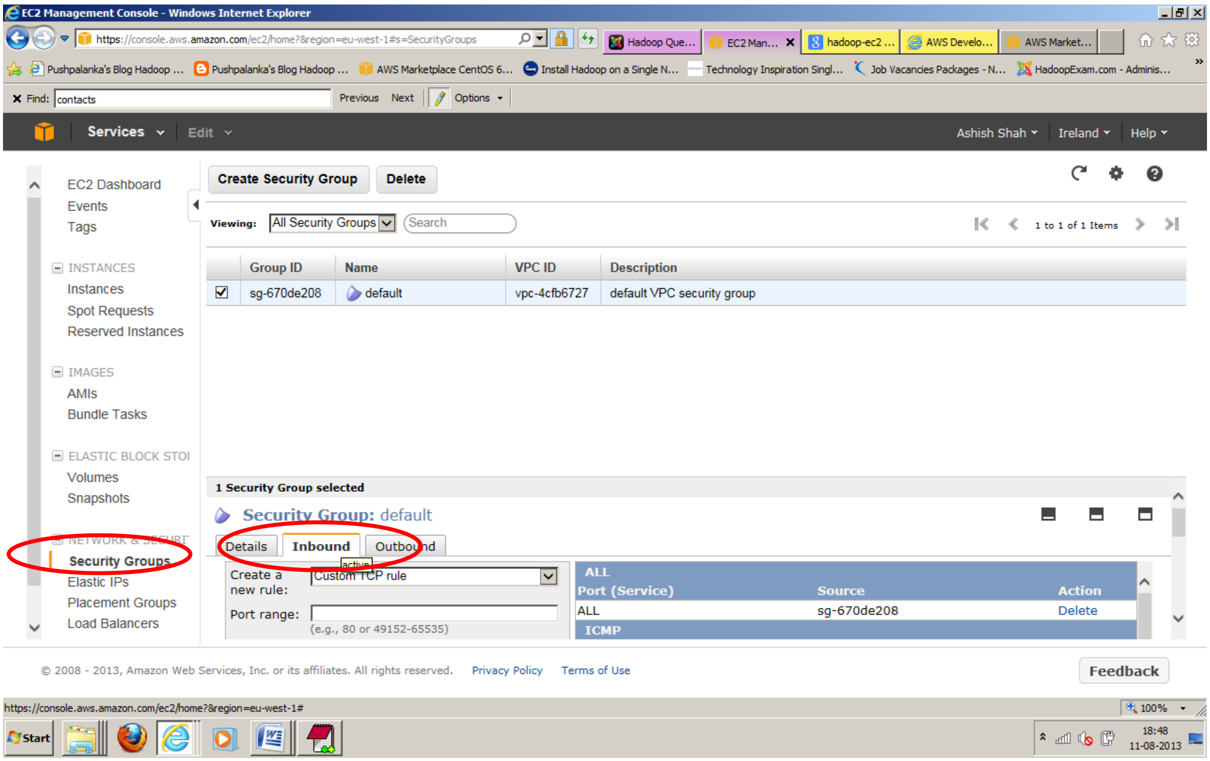
Public DNS: ec2-54-229-85-60.eu-west-1.compute.amazonaws.com

Private DNS: ip-172-31-12-208.eu-west-1.compute.internal

Product Codes:

Private IPs: 172.31.12.208

18. Under the Network and Security tab , click to “Security Groups” menu and then Select Inbound submenu. Here we configure some port, so this instance can be accessed over ssh (Secure Shell) and TCP. Follow the training how to apply the rule, its simple..



☑	Group ID	Name	VPC ID	Description
☑	sg-670de208	default	vpc-4cfb6727	default VPC security group

1 Security Group selected

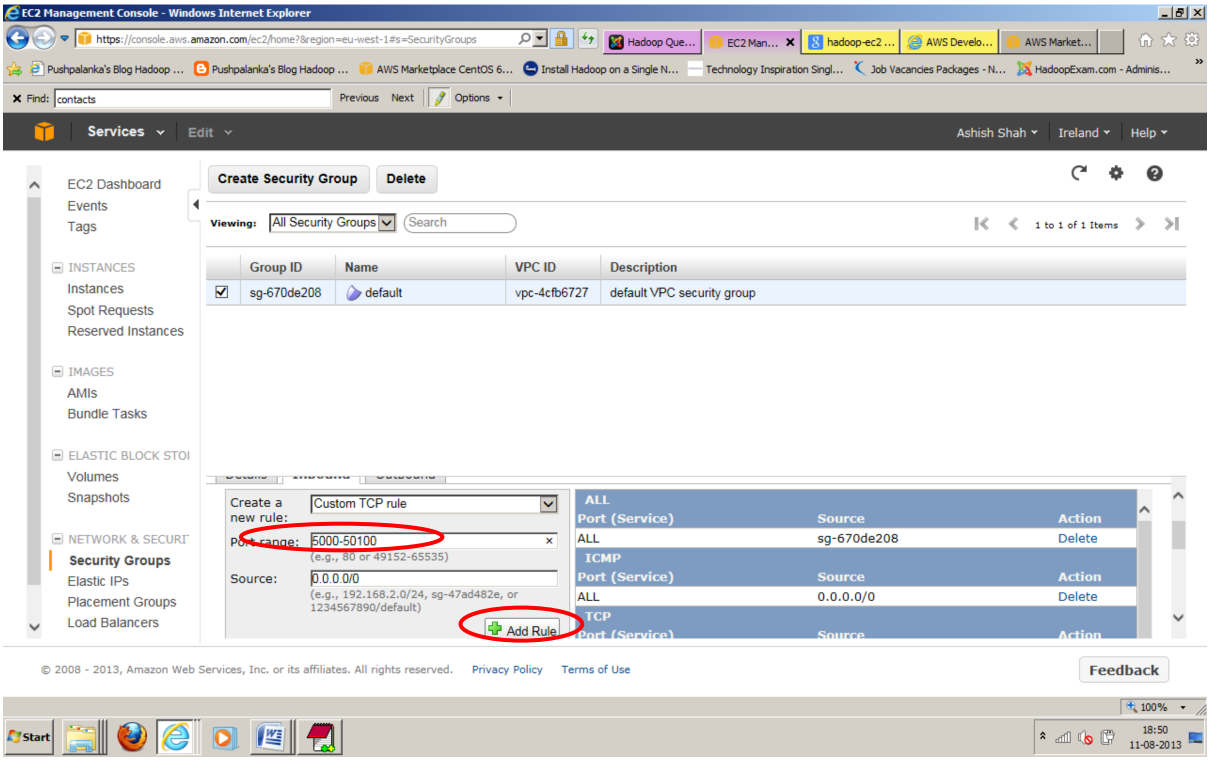
Security Group: default

Create a new rule: Custom TCP rule

Port range: (e.g., 80 or 49152-65535)

Port (Service)	Source	Action
ALL		
ALL	sg-670de208	Delete
ICMP		

19. Add Rule for 50000-50100 port first click on to Add Rule and Then Apply Rule Changes, similarly apply this rule for port 9000,9001,9100 etc.



Group ID	Name	VPC ID	Description
sg-670de208	default	vpc-4cfb6727	default VPC security group

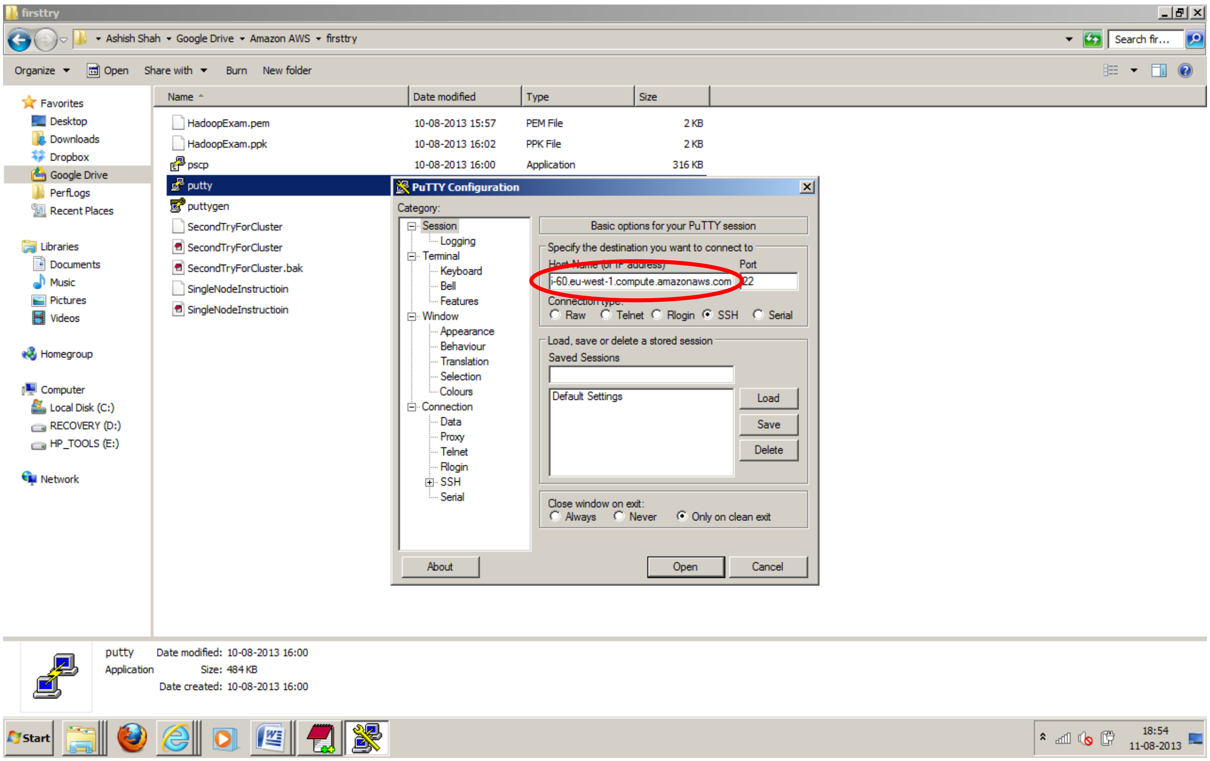
Create a new rule: Custom TCP rule

Port range: 5000-50100 (e.g., 80 or 49152-65535)

Source: 0.0.0.0/0 (e.g., 192.168.2.0/24, sg-47ad482e, or 1234567890/default)

Port (Service)	Source	Action
ALL		
ALL	sg-670de208	Delete
ICMP		
ALL	0.0.0.0/0	Delete
TCP		

20. Now your instance is ready to be connected with putty and you can work on linux instance directly. Open the putty and add the HostName (Public DNS of your Instance)



Name	Date modified	Type	Size
HadoopExam.pem	10-08-2013 15:57	PEM File	2 KB
HadoopExam.ppk	10-08-2013 16:02	PPK File	2 KB
pscp	10-08-2013 16:00	Application	316 KB
putty			
puttygen			
SecondTryForCluster			
SecondTryForCluster			
SecondTryForCluster.bak			
SingleNodeInstructioin			
SingleNodeInstructioin			

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): -60.eu-west-1.compute.amazonaws.com Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

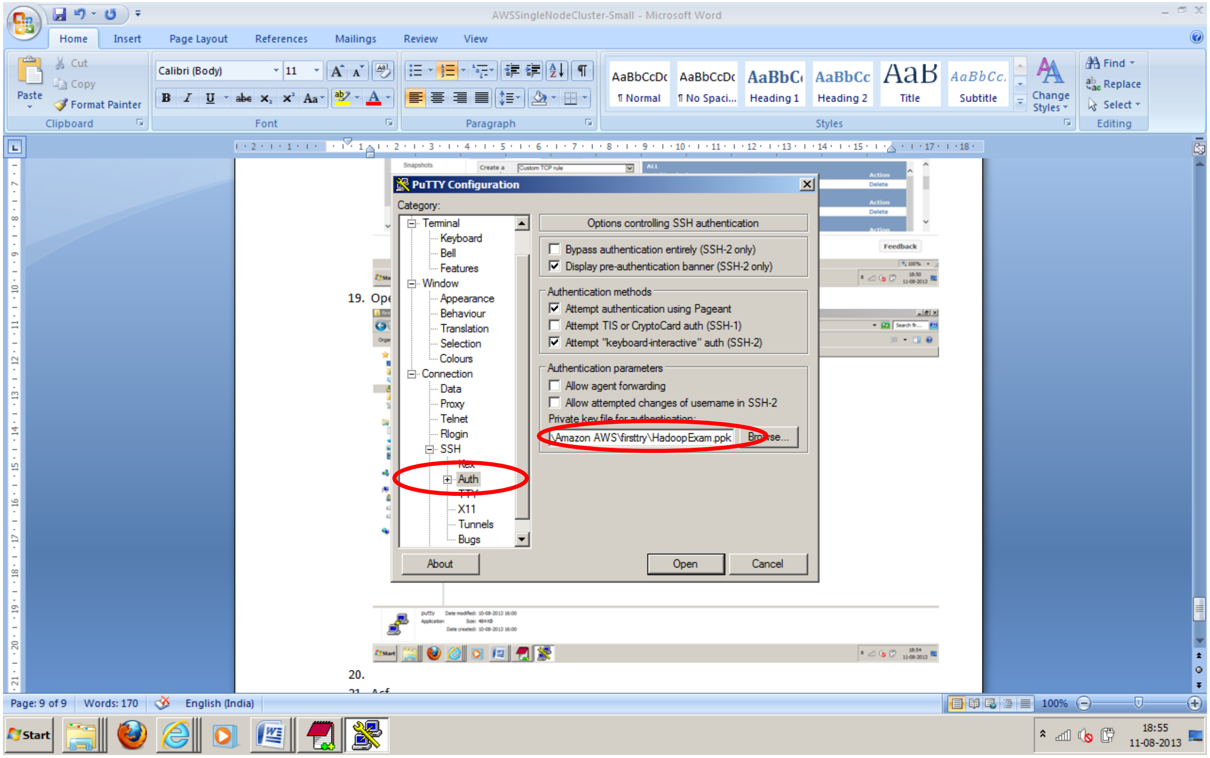
Load, save or delete a stored session

Saved Sessions: Default Settings

Close window on exit: ○ Always ○ Never ● Only on clean exit

putty Date modified: 10-08-2013 16:00
Application Size: 484 KB
Date created: 10-08-2013 16:00

21. Under the SSH Menu Select Auth and then add HadoopExam.ppk file, which we had already created in previous steps.



PuTTY Configuration

Category: Terminal, Keyboard, Bell, Features, Window, Appearance, Behaviour, Translation, Selection, Colours, Connection, Data, Proxy, Telnet, Rlogin, SSH, Kex, Auth, TTY, X11, Tunnels, Bugs

Options controlling SSH authentication

☐ Bypass authentication entirely (SSH-2 only)
☑ Display pre-authentication banner (SSH-2 only)

Authentication methods

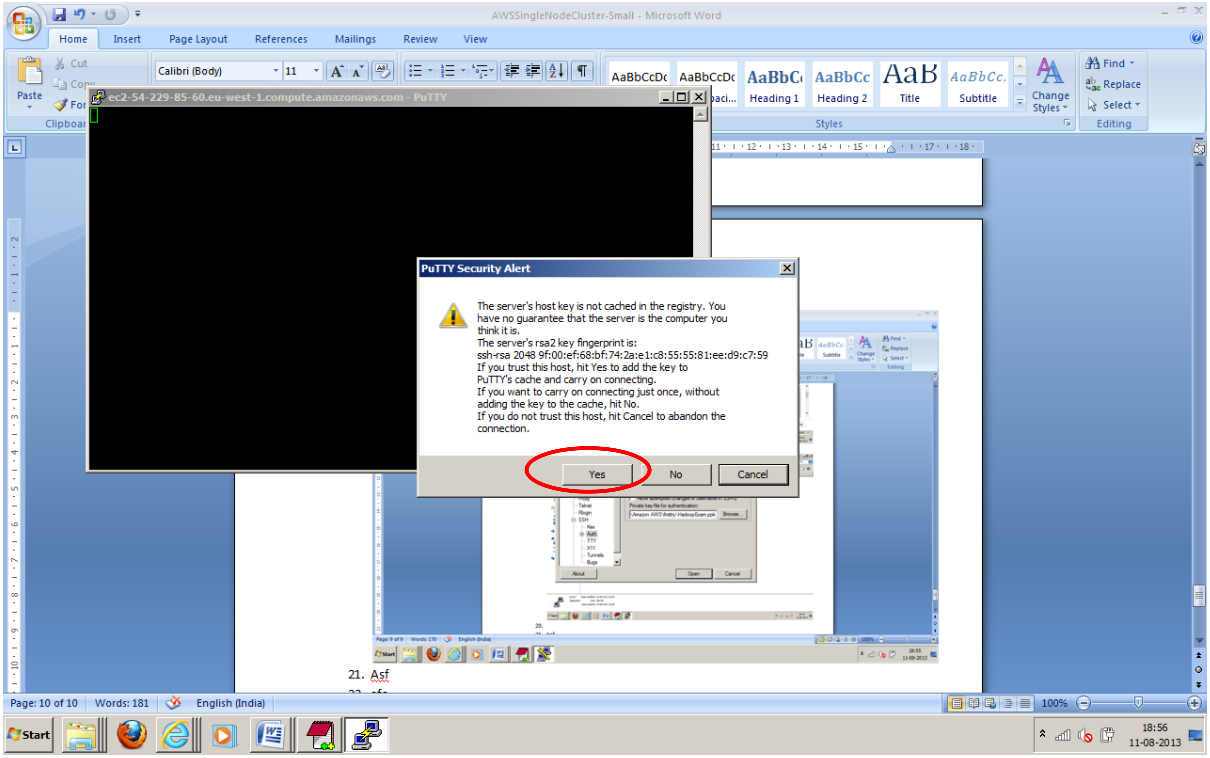
☑ Attempt authentication using Pageant
☐ Attempt TIS or CryptoCard auth (SSH-1)
☑ Attempt "keyboard-interactive" auth (SSH-2)

Authentication parameters

☐ Allow agent forwarding
☐ Allow attempted changes of username in SSH-2

Private key file for authentication: \Amazon AWS\firsttry\HadoopExam.ppk

22. Say Yes, it has be done only once.

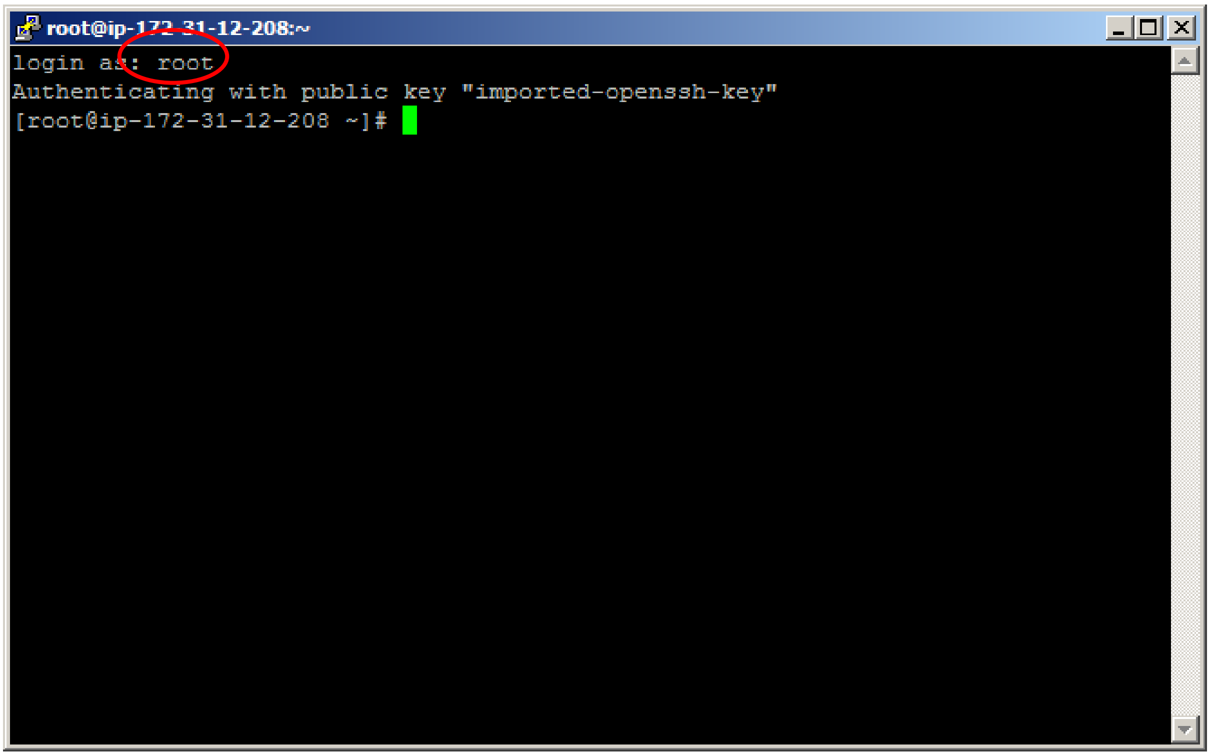


ec2-54-229-85-60.eu-west-1.compute.amazonaws.com - PuTTY

PuTTY Security Alert

The server's host key is not cached in the registry. You have no guarantee that the server is the computer you think it is.
The server's rsa2 key fingerprint is:
ssh-rsa 2048 9f:00:ef:68:bf:74:2a:e1:c8:55:55:81:ee:d9:c7:59
If you trust this host, hit Yes to add the key to PuTTY's cache and carry on connecting.
If you want to carry on connecting just once, without adding the key to the cache, hit No.
If you do not trust this host, hit Cancel to abandon the connection.

23. Default user name for CentOs Linux is root; hence login with the 'root' user. And we will work with this user only.

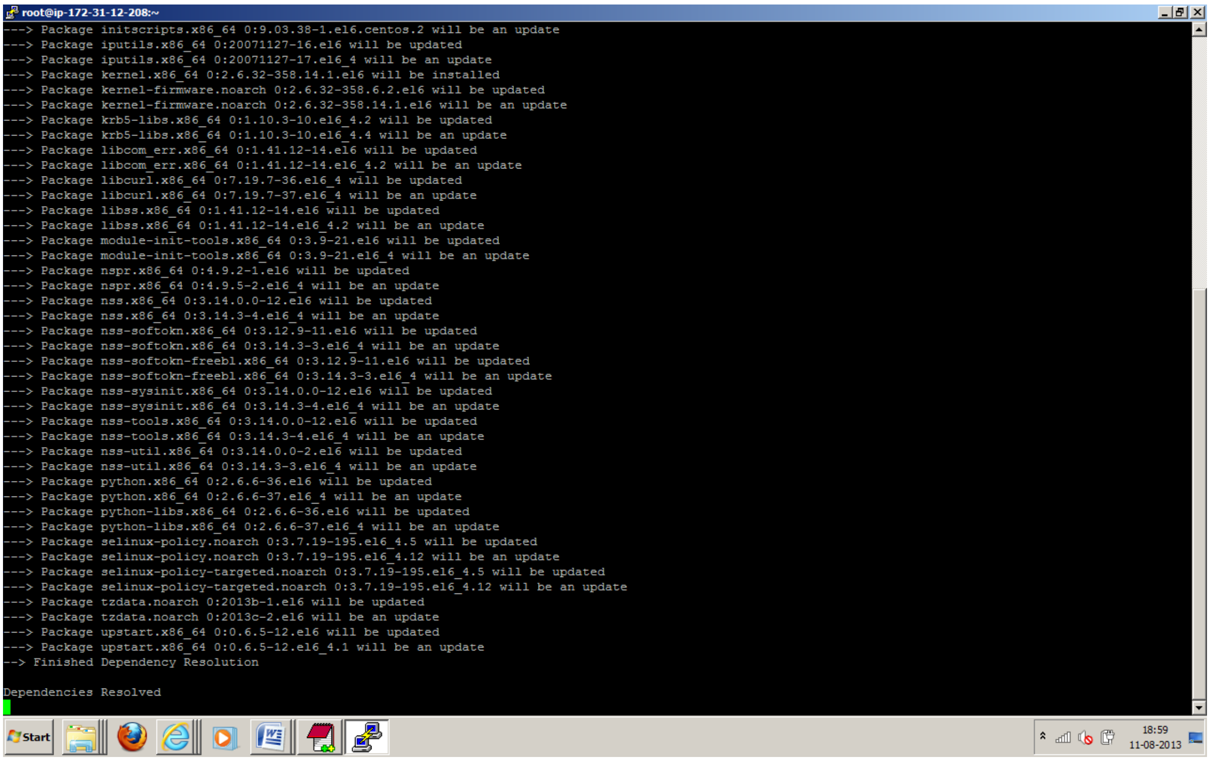


```
login as: root
Authenticating with public key "imported-openssh-key"
[root@ip-172-31-12-208 ~]#
```

24. Type following Command (Optional). And this will update all the available packages from some repository hosted somewhere on internet.

```
yum update
```

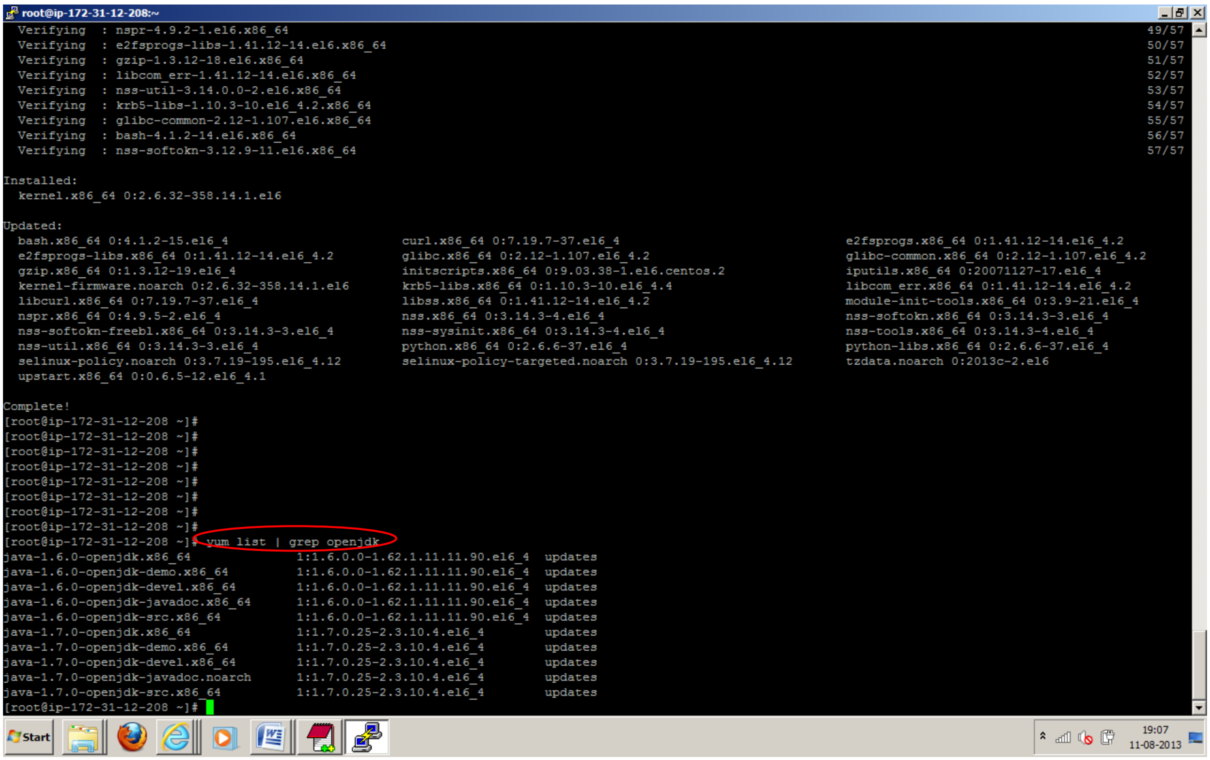
Keep typing y, when it asked and wait for some time.



```
---> Package initscripts.x86_64 0:9.03.38-1.el6.centos.2 will be an update
---> Package iputils.x86_64 0:20071127-16.el6 will be updated
---> Package iputils.x86_64 0:20071127-17.el6_4 will be an update
---> Package kernel.x86_64 0:2.6.32-358.14.1.el6 will be installed
---> Package kernel-firmware.noarch 0:2.6.32-358.6.2.el6 will be updated
---> Package kernel-firmware.noarch 0:2.6.32-358.14.1.el6 will be an update
---> Package krb5-libs.x86_64 0:1.10.3-10.el6_4.2 will be updated
---> Package krb5-libs.x86_64 0:1.10.3-10.el6_4.4 will be an update
---> Package libcom_err.x86_64 0:1.41.12-14.el6 will be updated
---> Package libcom_err.x86_64 0:1.41.12-14.el6_4.2 will be an update
---> Package libcurl.x86_64 0:7.19.7-36.el6_4 will be updated
---> Package libcurl.x86_64 0:7.19.7-37.el6_4 will be an update
---> Package libss.x86_64 0:1.41.12-14.el6 will be updated
---> Package libss.x86_64 0:1.41.12-14.el6_4.2 will be an update
---> Package module-init-tools.x86_64 0:3.9-21.el6 will be updated
---> Package module-init-tools.x86_64 0:3.9-21.el6_4 will be an update
---> Package nspr.x86_64 0:4.9.2-1.el6 will be updated
---> Package nspr.x86_64 0:4.9.5-2.el6_4 will be an update
---> Package nss.x86_64 0:3.14.0.0-12.el6 will be updated
---> Package nss.x86_64 0:3.14.3-4.el6_4 will be an update
---> Package nss-softokn.x86_64 0:3.12.9-11.el6 will be updated
---> Package nss-softokn.x86_64 0:3.14.3-3.el6_4 will be an update
---> Package nss-softokn-freebl.x86_64 0:3.12.9-11.el6 will be updated
---> Package nss-softokn-freebl.x86_64 0:3.14.3-3.el6_4 will be an update
---> Package nss-sysinit.x86_64 0:3.14.0.0-12.el6 will be updated
---> Package nss-sysinit.x86_64 0:3.14.3-4.el6_4 will be an update
---> Package nss-tools.x86_64 0:3.14.0.0-12.el6 will be updated
---> Package nss-tools.x86_64 0:3.14.3-4.el6_4 will be an update
---> Package nss-util.x86_64 0:3.14.0.0-2.el6 will be updated
---> Package nss-util.x86_64 0:3.14.3-3.el6_4 will be an update
---> Package python.x86_64 0:2.6.6-36.el6 will be updated
---> Package python.x86_64 0:2.6.6-37.el6_4 will be an update
---> Package python-libs.x86_64 0:2.6.6-36.el6 will be updated
---> Package python-libs.x86_64 0:2.6.6-37.el6_4 will be an update
---> Package selinux-policy.noarch 0:3.7.19-195.el6_4.5 will be updated
---> Package selinux-policy.noarch 0:3.7.19-195.el6_4.12 will be an update
---> Package selinux-policy-targeted.noarch 0:3.7.19-195.el6_4.5 will be updated
---> Package selinux-policy-targeted.noarch 0:3.7.19-195.el6_4.12 will be an update
---> Package tzdata.noarch 0:2013b-1.el6 will be updated
---> Package tzdata.noarch 0:2013c-2.el6 will be an update
---> Package upstart.x86_64 0:0.6.5-12.el6 will be updated
---> Package upstart.x86_64 0:0.6.5-12.el6_4.1 will be an update
--> Finished Dependency Resolution

Dependencies Resolved
```

25. Then Look for Open JDK with following command, we will be using OpenJDk for this example. To find any package in the linux with the help of yum utility type below command.

```
yum list | grep openjdk
```



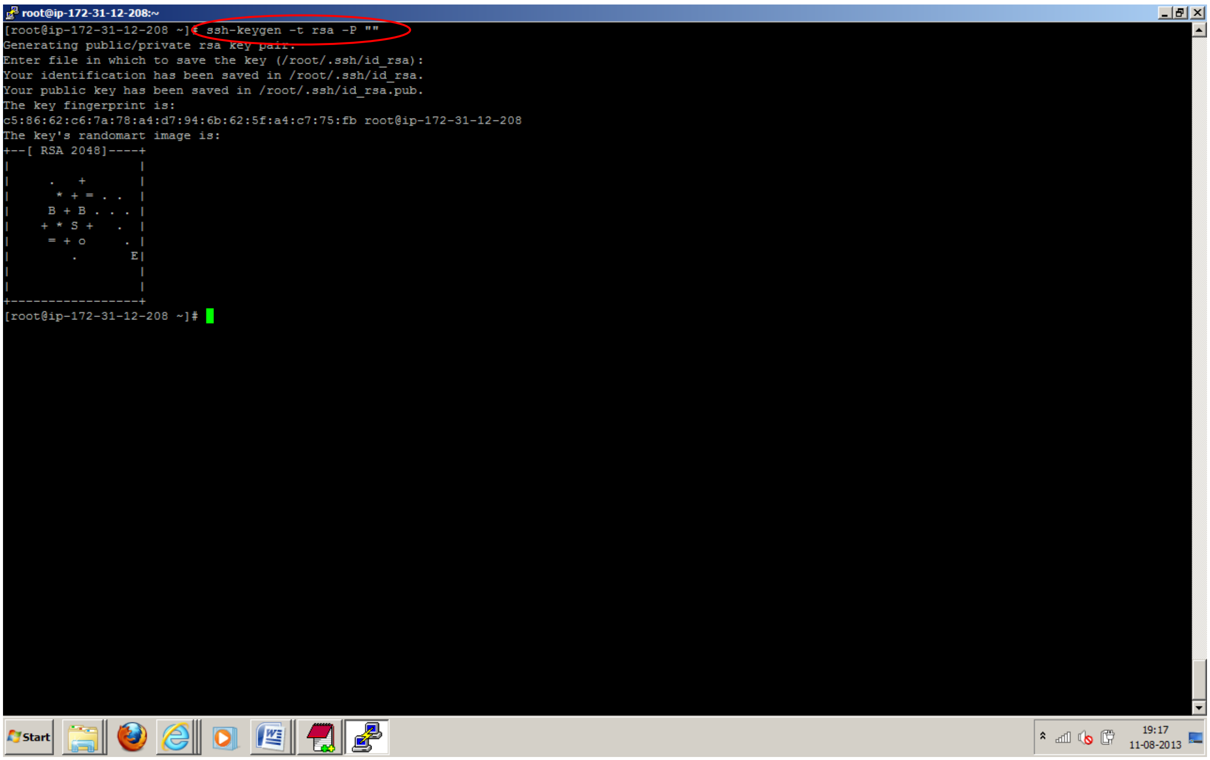
```
  Verifying  : nspr-4.9.2-1.el6.x86_64                                                                                       49/57
  Verifying  : e2fsprogs-libs-1.41.12-14.el6.x86_64                                                                          50/57
  Verifying  : gzip-1.3.12-18.el6.x86_64                                                                                     51/57
  Verifying  : libcom_err-1.41.12-14.el6.x86_64                                                                              52/57
  Verifying  : nss-util-3.14.0.0-2.el6.x86_64                                                                                53/57
  Verifying  : krb5-libs-1.10.3-10.el6_4.2.x86_64                                                                            54/57
  Verifying  : glibc-common-2.12-1.107.el6.x86_64                                                                            55/57
  Verifying  : bash-4.1.2-14.el6.x86_64                                                                                      56/57
  Verifying  : nss-softokn-3.12.9-11.el6.x86_64                                                                              57/57

Installed:
  kernel.x86_64 0:2.6.32-358.14.1.el6

Updated:
  bash.x86_64 0:4.1.2-15.el6_4                       curl.x86_64 0:7.19.7-37.el6_4                          e2fsprogs.x86_64 0:1.41.12-14.el6_4.2
  e2fsprogs-libs.x86_64 0:1.41.12-14.el6_4.2         glibc.x86_64 0:2.12-1.107.el6_4.2                      glibc-common.x86_64 0:2.12-1.107.el6_4.2
  gzip.x86_64 0:1.3.12-19.el6_4                      initscripts.x86_64 0:9.03.38-1.el6.centos.2            iputils.x86_64 0:20071127-17.el6_4
  kernel-firmware.noarch 0:2.6.32-358.14.1.el6       krb5-libs.x86_64 0:1.10.3-10.el6_4.4                   libcom_err.x86_64 0:1.41.12-14.el6_4.2
  libcurl.x86_64 0:7.19.7-37.el6_4                   libss.x86_64 0:1.41.12-14.el6_4.2                      module-init-tools.x86_64 0:3.9-21.el6_4
  nspr.x86_64 0:4.9.5-2.el6_4                        nss.x86_64 0:3.14.3-4.el6_4                            nss-softokn.x86_64 0:3.14.3-3.el6_4
  nss-softokn-freebl.x86_64 0:3.14.3-3.el6_4         nss-sysinit.x86_64 0:3.14.3-4.el6_4                    nss-tools.x86_64 0:3.14.3-4.el6_4
  nss-util.x86_64 0:3.14.3-3.el6_4                   python.x86_64 0:2.6.6-37.el6_4                         python-libs.x86_64 0:2.6.6-37.el6_4
  selinux-policy.noarch 0:3.7.19-195.el6_4.12        selinux-policy-targeted.noarch 0:3.7.19-195.el6_4.12   tzdata.noarch 0:2013c-2.el6
  upstart.x86_64 0:0.6.5-12.el6_4.1

Complete!
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]# yum list | grep openjdk
java-1.6.0-openjdk.x86_64              1:1.6.0.0-1.62.1.11.11.90.el6_4  updates
java-1.6.0-openjdk-demo.x86_64         1:1.6.0.0-1.62.1.11.11.90.el6_4  updates
java-1.6.0-openjdk-devel.x86_64        1:1.6.0.0-1.62.1.11.11.90.el6_4  updates
java-1.6.0-openjdk-javadoc.x86_64      1:1.6.0.0-1.62.1.11.11.90.el6_4  updates
java-1.6.0-openjdk-src.x86_64          1:1.6.0.0-1.62.1.11.11.90.el6_4  updates
java-1.7.0-openjdk.x86_64              1:1.7.0.25-2.3.10.4.el6_4        updates
java-1.7.0-openjdk-demo.x86_64         1:1.7.0.25-2.3.10.4.el6_4        updates
java-1.7.0-openjdk-devel.x86_64        1:1.7.0.25-2.3.10.4.el6_4        updates
java-1.7.0-openjdk-javadoc.noarch      1:1.7.0.25-2.3.10.4.el6_4        updates
java-1.7.0-openjdk-src.x86_64          1:1.7.0.25-2.3.10.4.el6_4        updates
[root@ip-172-31-12-208 ~]#
```

26. Now install openjdk using following command

```
yum install java-1.6.0-openjdk.x86_64
```

27. Enable ssh access and avoid password creation with the following command.

```
ssh-keygen -t rsa -P "" and then keep pressing enter for default location
```

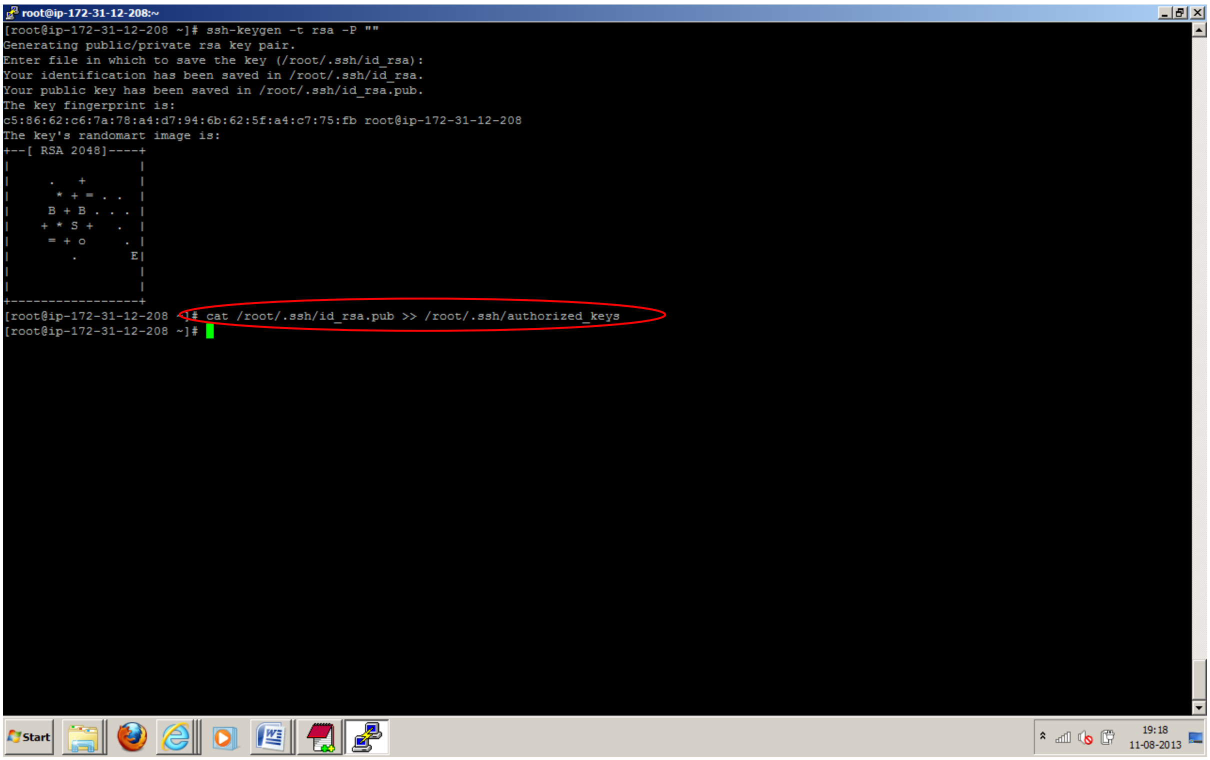


```
[root@ip-172-31-12-208 ~]# ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
c5:86:62:c6:7a:78:a4:d7:94:6b:62:5f:a4:c7:75:fb root@ip-172-31-12-208
The key's randomart image is:
+--[ RSA 2048]----+
|                 |
|     .   +       |
|      * + = . .  |
|     B + B . . . |
|    + * S +   .  |
|     = + o     . |
|        .       E|
|                 |
|                 |
+-----------------+
[root@ip-172-31-12-208 ~]#
```

28. And copy this key to enable SSH access from your local machine with newly created key, by applying following key.
    Provide the access to all keys.

```
chmod 700 /root/.ssh ; chmod 640 /root/.ssh/authorized_keys ; chmod 600 /root/.ssh/id_rsa
```

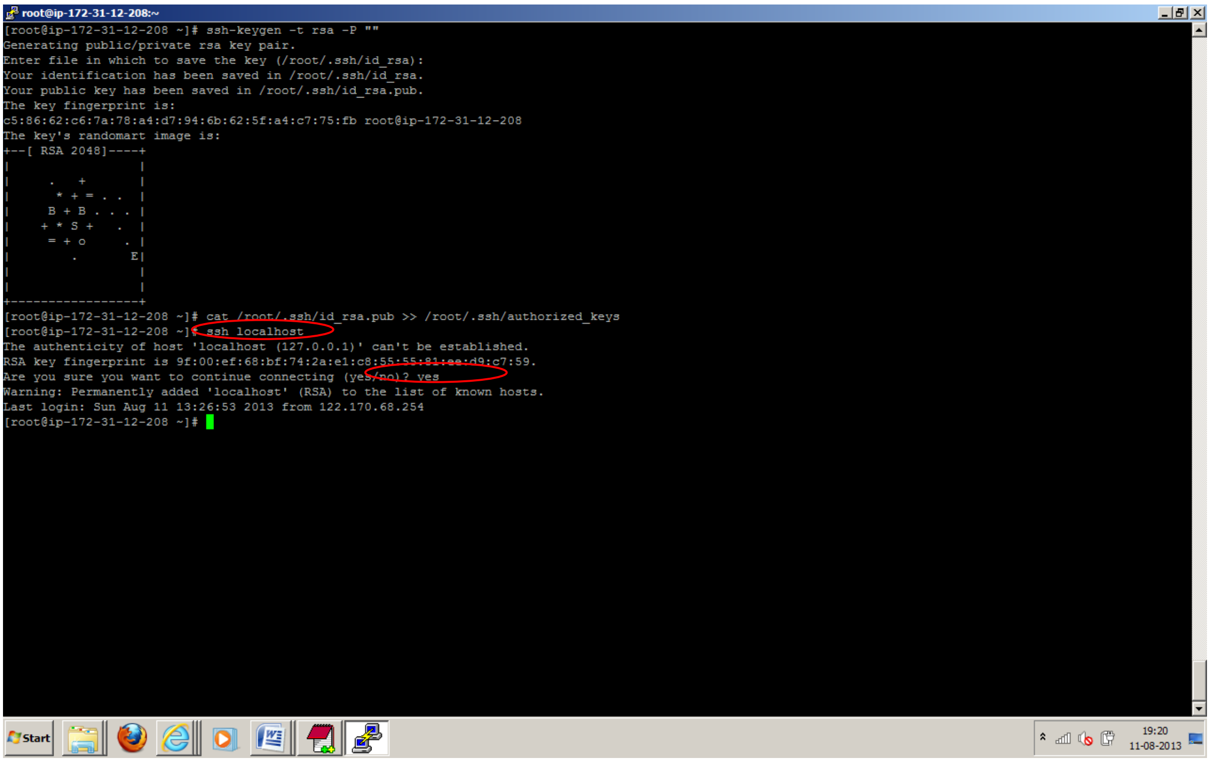
```
cat /root/.ssh/id_rsa.pub >> /root/.ssh/authorized_keys
```



```
[root@ip-172-31-12-208 ~]# ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
c5:86:62:c6:7a:78:a4:d7:94:6b:62:5f:a4:c7:75:fb root@ip-172-31-12-208
The key's randomart image is:
+--[ RSA 2048]----+
|                 |
|      .   +      |
|       * + = . . |
|      B + B . . .|
|     + * S +   . |
|      = + o     .|
|         .      E|
|                 |
|                 |
+-----------------+
[root@ip-172-31-12-208 ~]# cat /root/.ssh/id_rsa.pub >> /root/.ssh/authorized_keys
[root@ip-172-31-12-208 ~]#
```

29. Test the SSH setup by connecting to your local machine. The step is also needed to save your local machines host key fingerprint to the root user's known_hosts file. By typing following command and press yes.

```
ssh localhost
```



```
[root@ip-172-31-12-208 ~]# ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
c5:86:62:c6:7a:78:a4:d7:94:6b:62:5f:a4:c7:75:fb root@ip-172-31-12-208
The key's randomart image is:
+--[ RSA 2048]----+
|                 |
|      .   +      |
|       * + = . . |
|      B + B . . .|
|     + * S +   . |
|      = + o     .|
|         .      E|
|                 |
|                 |
+-----------------+
[root@ip-172-31-12-208 ~]# cat /root/.ssh/id_rsa.pub >> /root/.ssh/authorized_keys
[root@ip-172-31-12-208 ~]# ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
RSA key fingerprint is 9f:00:ef:68:bf:74:2a:e1:c8:55:55:81:ee:d9:c7:59.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (RSA) to the list of known hosts.
Last login: Sun Aug 11 13:26:53 2013 from 122.170.68.254
[root@ip-172-31-12-208 ~]#
```

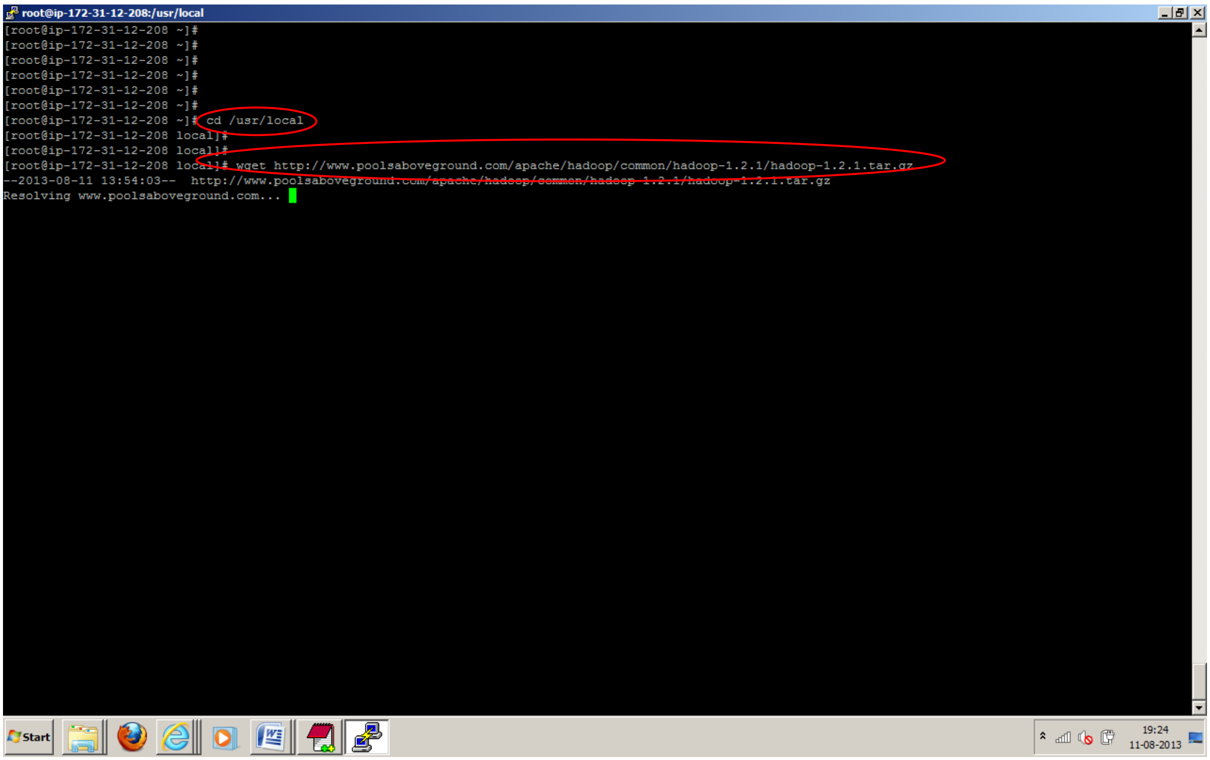
30. Install the **wget** tool with following command which will help us to download the software from internet which is available on http protocols.

```
yum -y install wget
```

31. Go to Directory **cd /usr/local** and download the Hadoop Latest Release version with following command

```
cd /usr/local
```

```
wget http://www.poolsaboveground.com/apache/hadoop/common/hadoop-1.2.1/hadoop-1.2.1.tar.gz
```

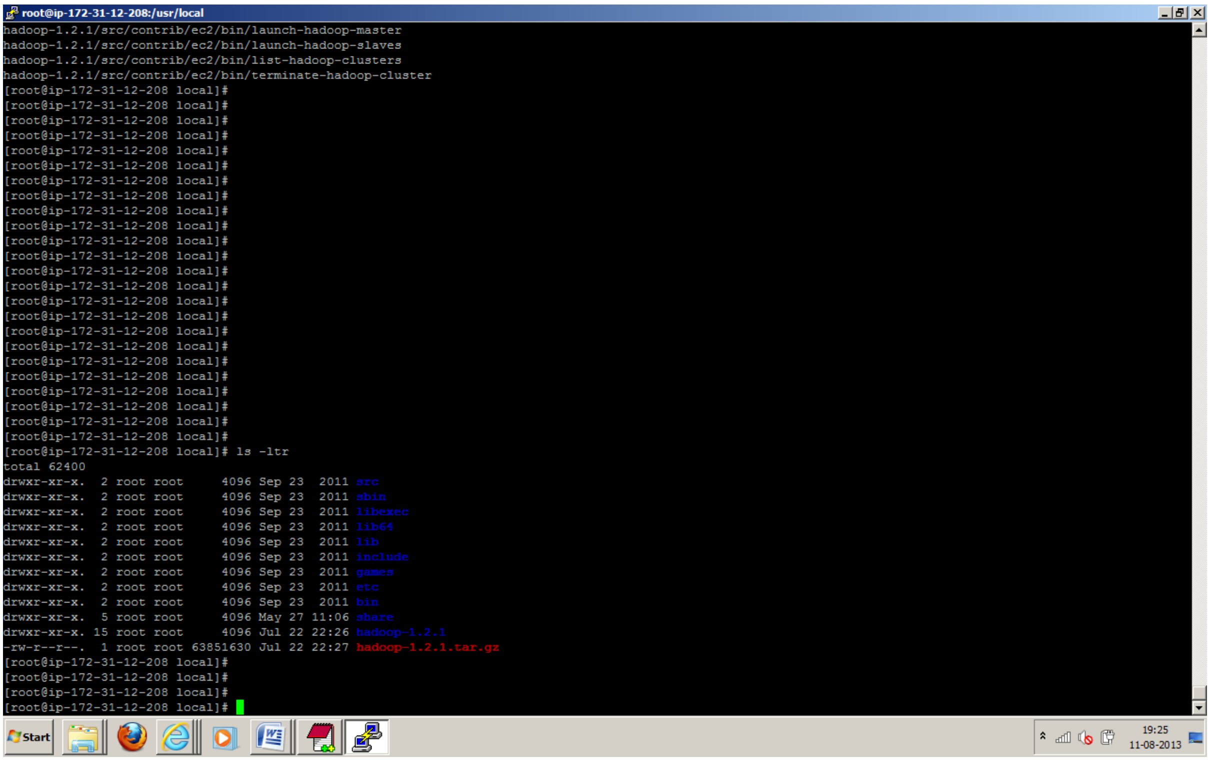


root@ip-172-31-12-208:/usr/local

```
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]#
[root@ip-172-31-12-208 ~]# cd /usr/local
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]# wget http://www.poolsaboveground.com/apache/hadoop/common/hadoop-1.2.1/hadoop-1.2.1.tar.gz
--2013-08-11 13:54:03--  http://www.poolsaboveground.com/apache/hadoop/common/hadoop-1.2.1/hadoop-1.2.1.tar.gz
Resolving www.poolsaboveground.com...
```

32. Now unzip/untar the downloaded Hadoop framework with command. And now Hadoop is installed on your amazon EC2 instance.

```
tar -zxvf hadoop-1.2.1.tar.gz
```

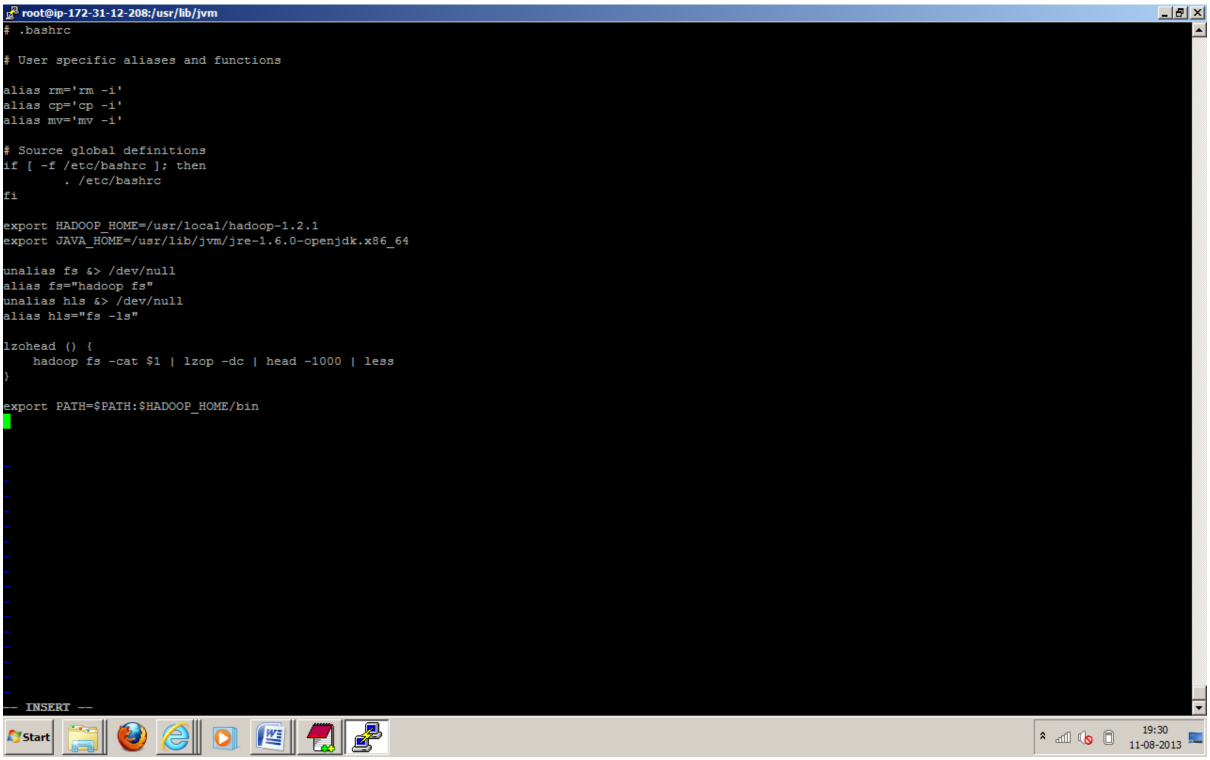


```
hadoop-1.2.1/src/contrib/ec2/bin/launch-hadoop-master
hadoop-1.2.1/src/contrib/ec2/bin/launch-hadoop-slaves
hadoop-1.2.1/src/contrib/ec2/bin/list-hadoop-clusters
hadoop-1.2.1/src/contrib/ec2/bin/terminate-hadoop-cluster
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]# ls -ltr
total 62400
drwxr-xr-x.  2 root root     4096 Sep 23  2011 src
drwxr-xr-x.  2 root root     4096 Sep 23  2011 sbin
drwxr-xr-x.  2 root root     4096 Sep 23  2011 libexec
drwxr-xr-x.  2 root root     4096 Sep 23  2011 lib64
drwxr-xr-x.  2 root root     4096 Sep 23  2011 lib
drwxr-xr-x.  2 root root     4096 Sep 23  2011 include
drwxr-xr-x.  2 root root     4096 Sep 23  2011 games
drwxr-xr-x.  2 root root     4096 Sep 23  2011 etc
drwxr-xr-x.  2 root root     4096 Sep 23  2011 bin
drwxr-xr-x.  5 root root     4096 May 27 11:06 share
drwxr-xr-x. 15 root root     4096 Jul 22 22:26 hadoop-1.2.1
-rw-r--r--.  1 root root 63851630 Jul 22 22:27 hadoop-1.2.1.tar.gz
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
[root@ip-172-31-12-208 local]#
```

33. Now set the JAVA_HOME and HADOOP_HOME in the root/.bashrc file, by copying the following content. As you know we have already installed Java and Hadoop in previous steps. Make sure you put proper path for java and Hadoop where it is installed. And save it by pressing, the way you save file in linux.
    esc:wq

```
vi /root/.bashrc
```

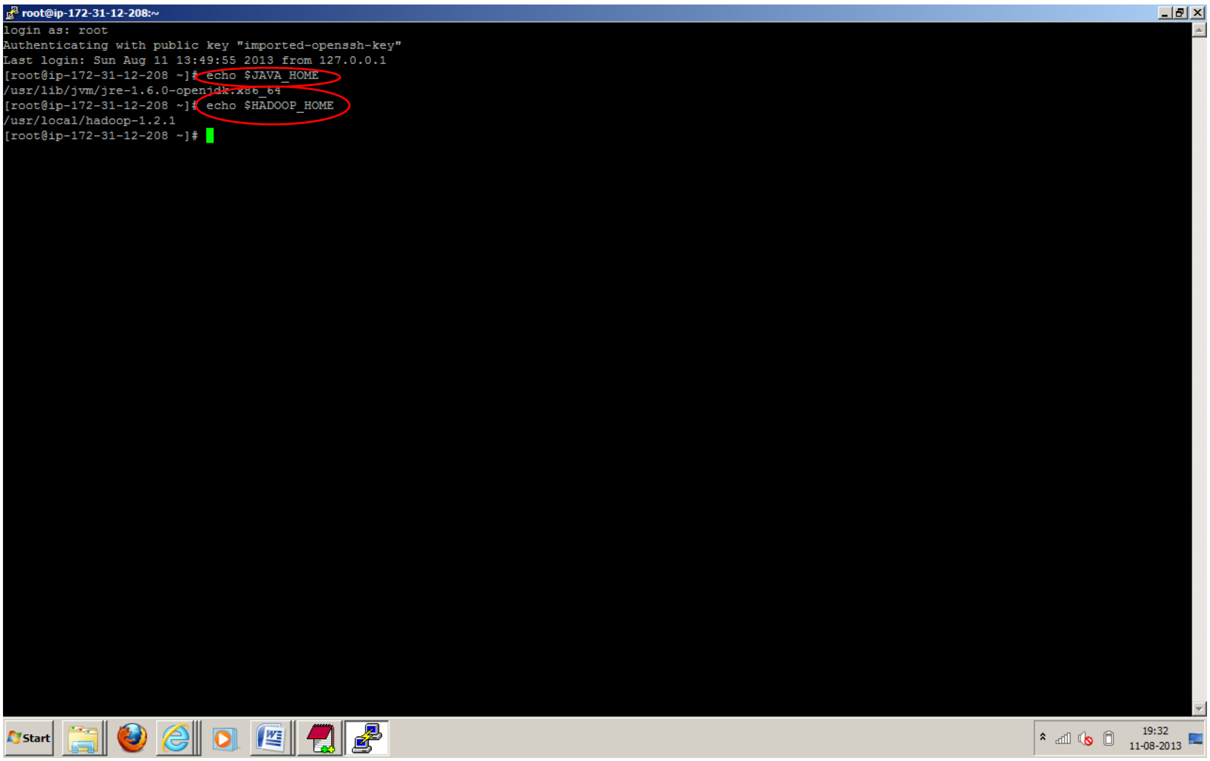
```
export HADOOP_HOME=/usr/local/hadoop-1.2.1
export JAVA_HOME=/usr/lib/jvm/jre-1.6.0-openjdk.x86_64

unalias fs &> /dev/null
alias fs="hadoop fs"
unalias hls &> /dev/null
alias hls="fs -ls"

lzohead () {
   hadoop fs -cat $1 | lzop -dc | head -1000 | less
}

export PATH=$PATH:$HADOOP_HOME/bin
```



```
# .bashrc

# User specific aliases and functions

alias rm='rm -i'
alias cp='cp -i'
alias mv='mv -i'

# Source global definitions
if [ -f /etc/bashrc ]; then
        . /etc/bashrc
fi

export HADOOP_HOME=/usr/local/hadoop-1.2.1
export JAVA_HOME=/usr/lib/jvm/jre-1.6.0-openjdk.x86_64

unalias fs &> /dev/null
alias fs="hadoop fs"
unalias hls &> /dev/null
alias hls="fs -ls"

lzohead () {
    hadoop fs -cat $1 | lzop -dc | head -1000 | less
}

export PATH=$PATH:$HADOOP_HOME/bin

-- INSERT --
```

34. Now restart putty shell to take effect this configuration and after restart JAVA_HOME and HADOOP_HOME should be available. And by typing following command you can make sure whether JAVA_HOME and HADOOP_HOME are pointing the installed location or not.

```
echo $JAVA_HOME
echo $HADOOP_HOME
```

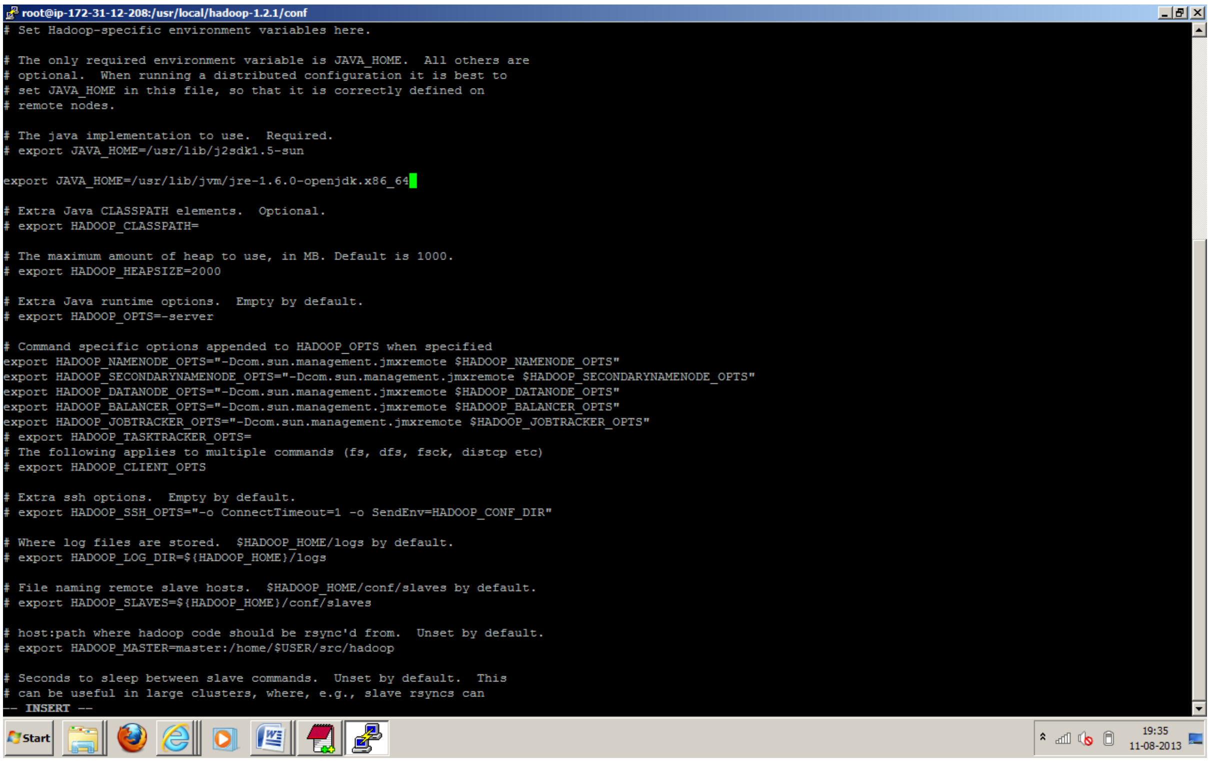


```
login as: root
Authenticating with public key "imported-openssh-key"
Last login: Sun Aug 11 13:49:55 2013 from 127.0.0.1
[root@ip-172-31-12-208 ~]# echo $JAVA_HOME
/usr/lib/jvm/jre-1.6.0-openjdk.x86_64
[root@ip-172-31-12-208 ~]# echo $HADOOP_HOME
/usr/local/hadoop-1.2.1
[root@ip-172-31-12-208 ~]#
```

35. Create temp directory for Hadoop Data storage. So here your all data will be stored, which you will be storing in hdfs file sytem

```
mkdir -p /tmp/hadoop/data
```

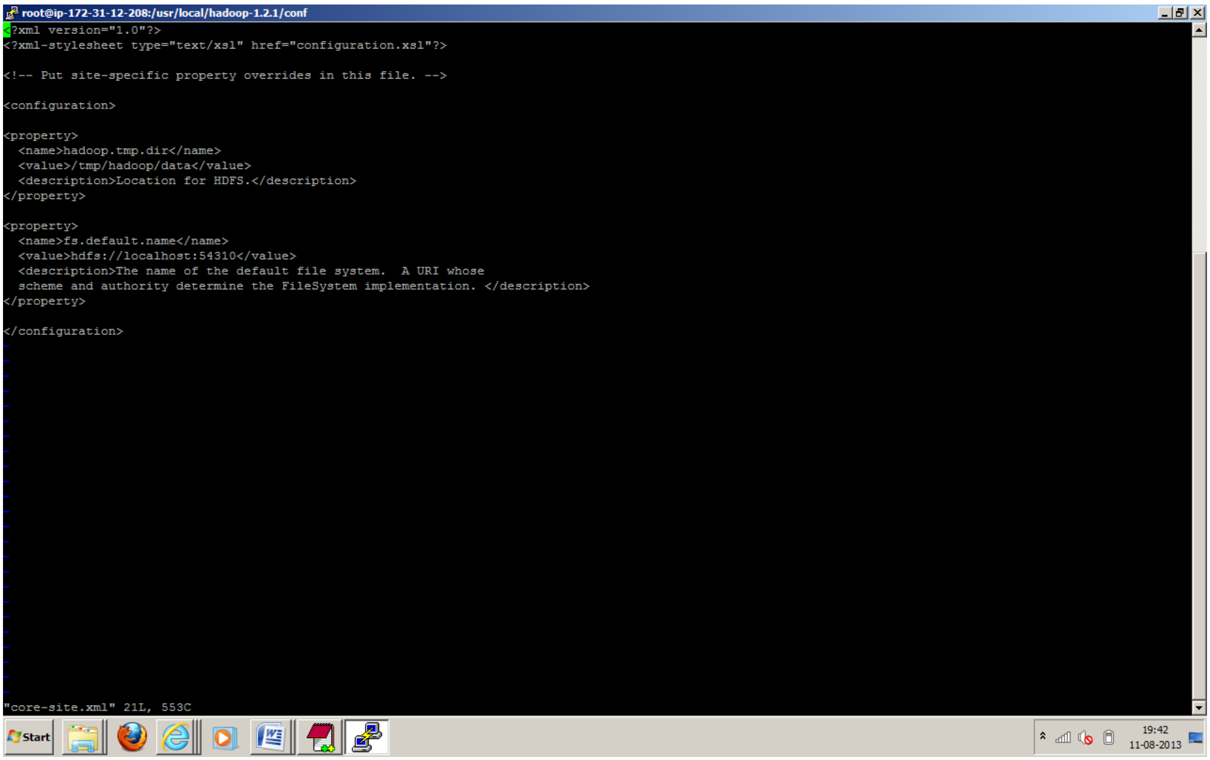
36. Set JAVA_HOME in /usr/local/hadoop-1.2.1/conf/hadoop-env.sh
Now, while starting the Hadoop Cluster it requires JAVA_HOME to be set in Hadoop-env.sh file. And as soon as you start the Hadoop it will use this file to read all Hadoop related configuration.



```
# Set Hadoop-specific environment variables here.

# The only required environment variable is JAVA_HOME.  All others are
# optional.  When running a distributed configuration it is best to
# set JAVA_HOME in this file, so that it is correctly defined on
# remote nodes.

# The java implementation to use.  Required.
# export JAVA_HOME=/usr/lib/j2sdk1.5-sun

export JAVA_HOME=/usr/lib/jvm/jre-1.6.0-openjdk.x86_64

# Extra Java CLASSPATH elements.  Optional.
# export HADOOP_CLASSPATH=

# The maximum amount of heap to use, in MB. Default is 1000.
# export HADOOP_HEAPSIZE=2000

# Extra Java runtime options.  Empty by default.
# export HADOOP_OPTS=-server

# Command specific options appended to HADOOP_OPTS when specified
export HADOOP_NAMENODE_OPTS="-Dcom.sun.management.jmxremote $HADOOP_NAMENODE_OPTS"
export HADOOP_SECONDARYNAMENODE_OPTS="-Dcom.sun.management.jmxremote $HADOOP_SECONDARYNAMENODE_OPTS"
export HADOOP_DATANODE_OPTS="-Dcom.sun.management.jmxremote $HADOOP_DATANODE_OPTS"
export HADOOP_BALANCER_OPTS="-Dcom.sun.management.jmxremote $HADOOP_BALANCER_OPTS"
export HADOOP_JOBTRACKER_OPTS="-Dcom.sun.management.jmxremote $HADOOP_JOBTRACKER_OPTS"
# export HADOOP_TASKTRACKER_OPTS=
# The following applies to multiple commands (fs, dfs, fsck, distcp etc)
# export HADOOP_CLIENT_OPTS

# Extra ssh options.  Empty by default.
# export HADOOP_SSH_OPTS="-o ConnectTimeout=1 -o SendEnv=HADOOP_CONF_DIR"

# Where log files are stored.  $HADOOP_HOME/logs by default.
# export HADOOP_LOG_DIR=${HADOOP_HOME}/logs

# File naming remote slave hosts.  $HADOOP_HOME/conf/slaves by default.
# export HADOOP_SLAVES=${HADOOP_HOME}/conf/slaves

# host:path where hadoop code should be rsync'd from.  Unset by default.
# export HADOOP_MASTER=master:/home/$USER/src/hadoop

# Seconds to sleep between slave commands.  Unset by default.  This
# can be useful in large clusters, where, e.g., slave rsyncs can
-- INSERT --
```

37. Now Configure the conf/core-site.xml with following content. It will set up the URI for namenode, in Hadoop cluster.

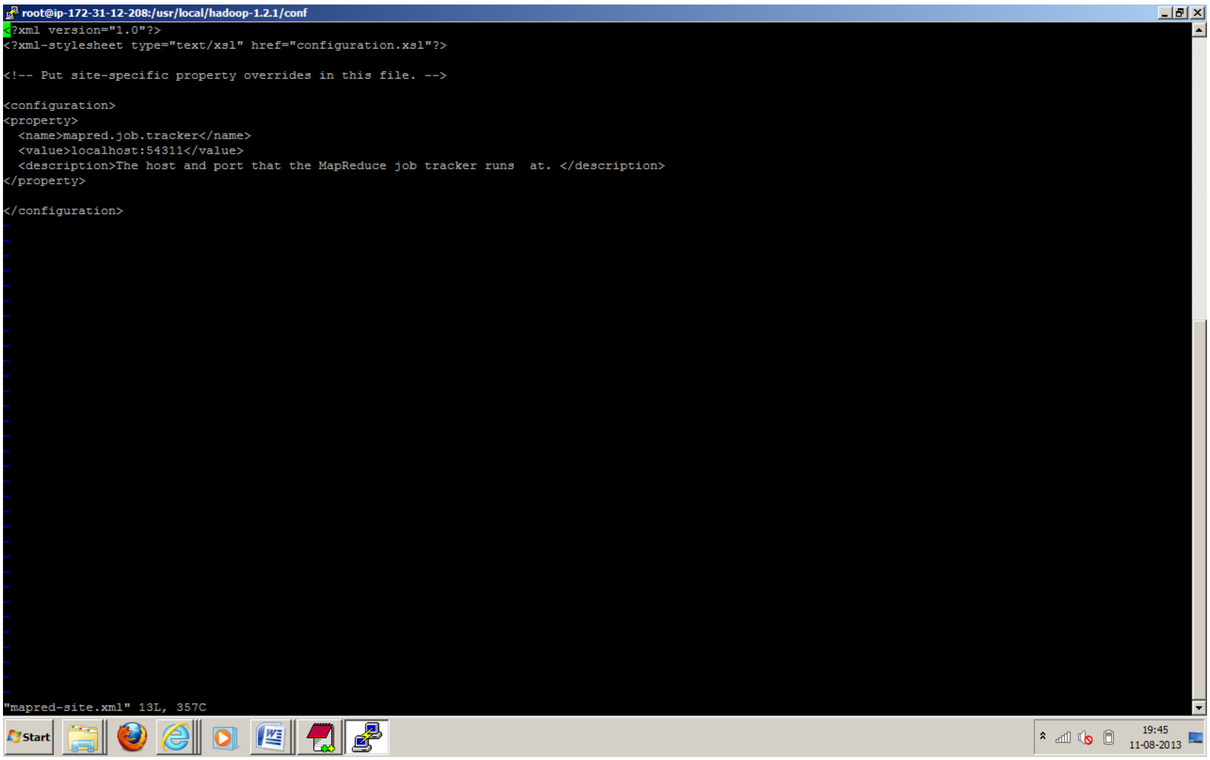
```
<configuration>
<property>
  <name>hadoop.tmp.dir</name>
  <value>/tmp/hadoop/data</value>
  <description>Location for HDFS.</description>
</property>

<property>
  <name>fs.default.name</name>
  <value>hdfs://localhost:54310</value>
  <description>The name of the default file system.  A URI whose
  scheme and authority determine the FileSystem implementation. </description>
</property>
</configuration>
```



```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!-- Put site-specific property overrides in this file. -->

<configuration>

<property>
  <name>hadoop.tmp.dir</name>
  <value>/tmp/hadoop/data</value>
  <description>Location for HDFS.</description>
</property>

<property>
  <name>fs.default.name</name>
  <value>hdfs://localhost:54310</value>
  <description>The name of the default file system.  A URI whose
  scheme and authority determine the FileSystem implementation. </description>
</property>

</configuration>

"core-site.xml" 21L, 553C
```

38. Configure the conf/mapred-site.xml with following content. It is the configuration for JobTracker.

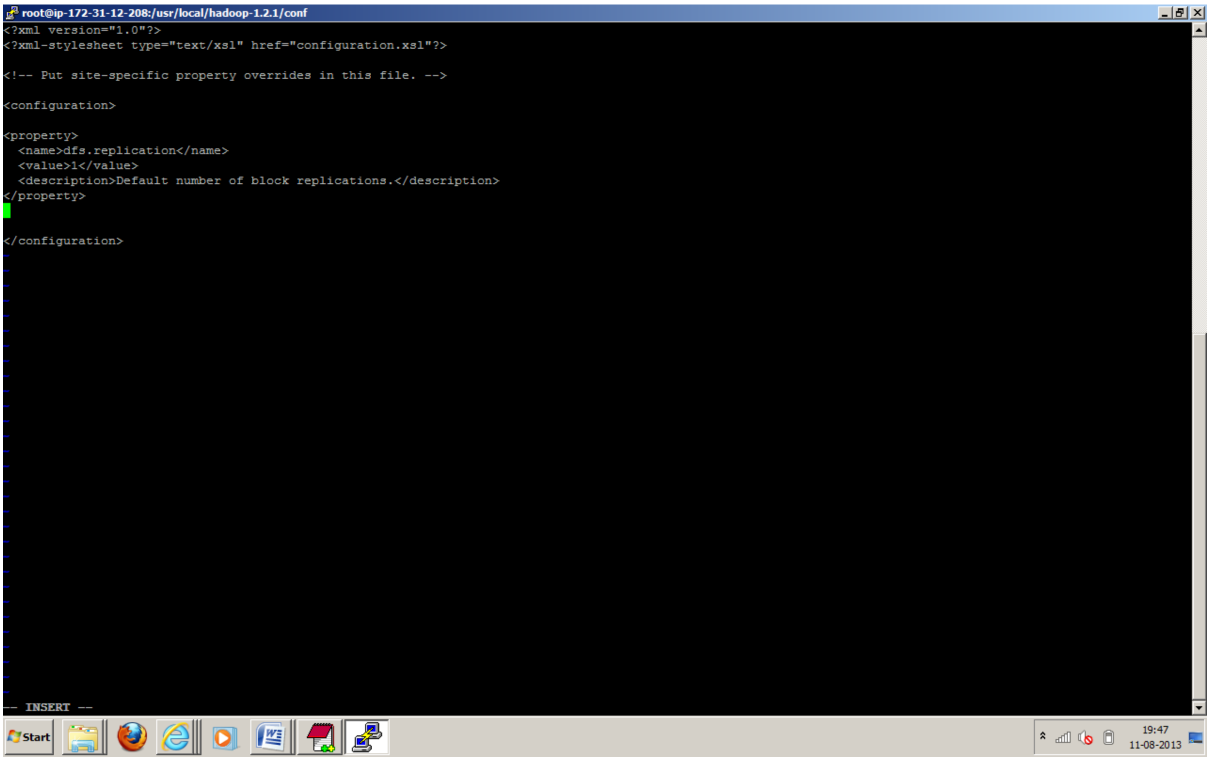
```
<configuration>
            <property>
             <name>mapred.job.tracker</name>
             <value>localhost:54311</value>
             <description>The host and port that the MapReduce job tracker runs  at.
            </description>
            </property>
</configuration>
```



```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
  <name>mapred.job.tracker</name>
  <value>localhost:54311</value>
  <description>The host and port that the MapReduce job tracker runs  at. </description>
</property>

</configuration>
"mapred-site.xml" 13L, 357C
```

39. Now configure conf/hdfs-site.xml. Replication factor configuration for the HDFS blocks.

```
<configuration>
            <property>
              <name>dfs.replication</name>
              <value>1</value>
              <description>Default number of block replications.</description>
            </property>
</configuration>
```

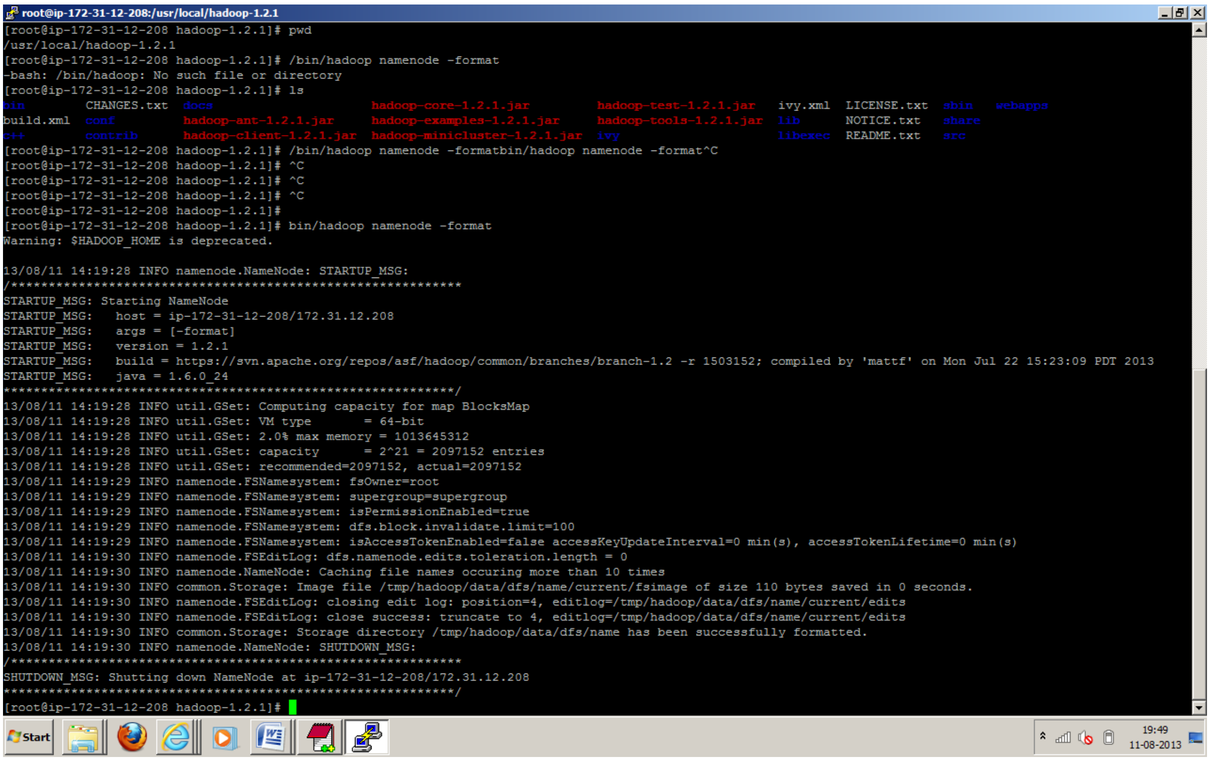


root@ip-172-31-12-208:/usr/local/hadoop-1.2.1/conf

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!-- Put site-specific property overrides in this file. -->

<configuration>

<property>
  <name>dfs.replication</name>
  <value>1</value>
  <description>Default number of block replications.</description>
</property>


</configuration>
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --
```

40. Format the hdfs with following command. Formatting the Hadoop filesystem, which is implemented on top of the local filesystems of your cluster, you need to do this the first time you set up a Hadoop installation. **Do not** format a running Hadoop filesystem, this will cause all your data to be erased.

```
bin/hadoop namenode -format
```

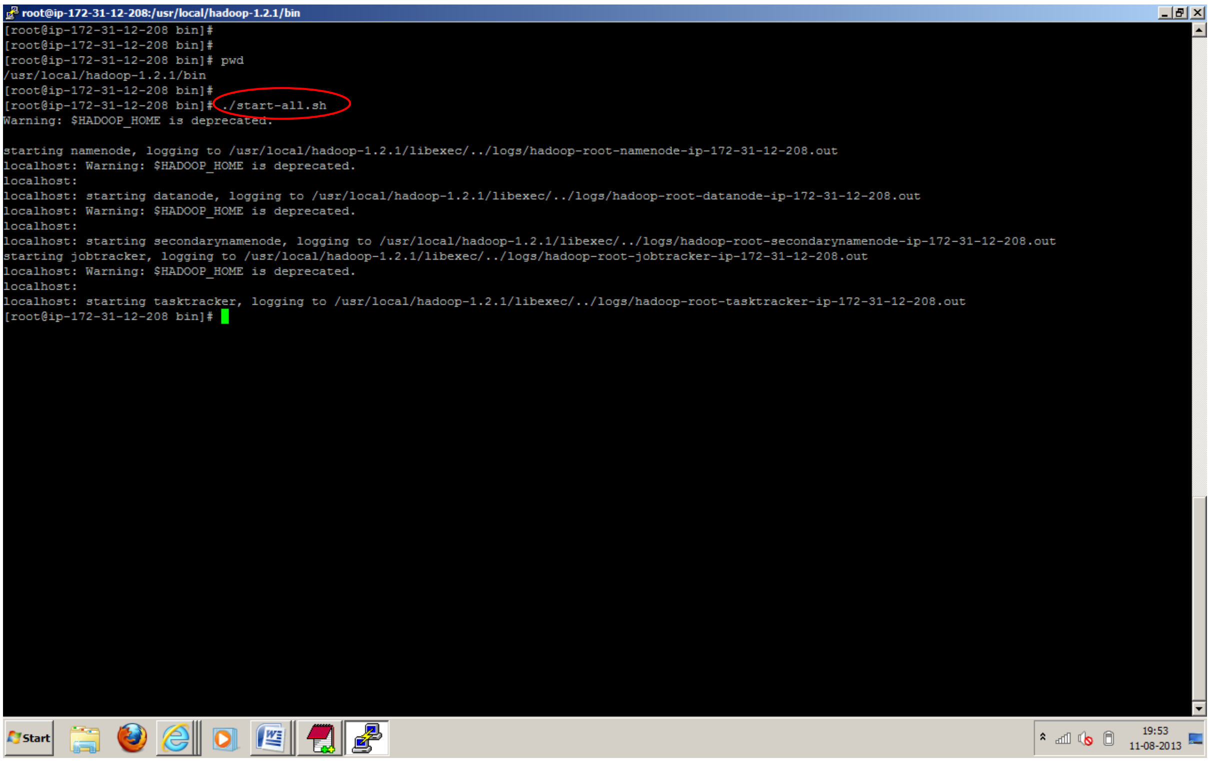


root@ip-172-31-12-208:/usr/local/hadoop-1.2.1

```
[root@ip-172-31-12-208 hadoop-1.2.1]# pwd
/usr/local/hadoop-1.2.1
[root@ip-172-31-12-208 hadoop-1.2.1]# /bin/hadoop namenode -format
-bash: /bin/hadoop: No such file or directory
[root@ip-172-31-12-208 hadoop-1.2.1]# ls
bin        CHANGES.txt  docs                      hadoop-core-1.2.1.jar         hadoop-test-1.2.1.jar   ivy.xml  LICENSE.txt  sbin   webapps
build.xml  conf         hadoop-ant-1.2.1.jar      hadoop-examples-1.2.1.jar     hadoop-tools-1.2.1.jar  lib      NOTICE.txt   share
c++        contrib      hadoop-client-1.2.1.jar   hadoop-minicluster-1.2.1.jar  ivy                     libexec  README.txt   src
[root@ip-172-31-12-208 hadoop-1.2.1]# /bin/hadoop namenode -formatbin/hadoop namenode -format^C
[root@ip-172-31-12-208 hadoop-1.2.1]# ^C
[root@ip-172-31-12-208 hadoop-1.2.1]# ^C
[root@ip-172-31-12-208 hadoop-1.2.1]# ^C
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]# bin/hadoop namenode -format
Warning: $HADOOP_HOME is deprecated.

13/08/11 14:19:28 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = ip-172-31-12-208/172.31.12.208
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 1.2.1
STARTUP_MSG:   build = https://svn.apache.org/repos/asf/hadoop/common/branches/branch-1.2 -r 1503152; compiled by 'mattf' on Mon Jul 22 15:23:09 PDT 2013
STARTUP_MSG:   java = 1.6.0_24
************************************************************/
13/08/11 14:19:28 INFO util.GSet: Computing capacity for map BlocksMap
13/08/11 14:19:28 INFO util.GSet: VM type       = 64-bit
13/08/11 14:19:28 INFO util.GSet: 2.0% max memory = 1013645312
13/08/11 14:19:28 INFO util.GSet: capacity      = 2^21 = 2097152 entries
13/08/11 14:19:28 INFO util.GSet: recommended=2097152, actual=2097152
13/08/11 14:19:29 INFO namenode.FSNamesystem: fsOwner=root
13/08/11 14:19:29 INFO namenode.FSNamesystem: supergroup=supergroup
13/08/11 14:19:29 INFO namenode.FSNamesystem: isPermissionEnabled=true
13/08/11 14:19:29 INFO namenode.FSNamesystem: dfs.block.invalidate.limit=100
13/08/11 14:19:29 INFO namenode.FSNamesystem: isAccessTokenEnabled=false accessKeyUpdateInterval=0 min(s), accessTokenLifetime=0 min(s)
13/08/11 14:19:30 INFO namenode.FSEditLog: dfs.namenode.edits.toleration.length = 0
13/08/11 14:19:30 INFO namenode.NameNode: Caching file names occuring more than 10 times
13/08/11 14:19:30 INFO common.Storage: Image file /tmp/hadoop/data/dfs/name/current/fsimage of size 110 bytes saved in 0 seconds.
13/08/11 14:19:30 INFO namenode.FSEditLog: closing edit log: position=4, editlog=/tmp/hadoop/data/dfs/name/current/edits
13/08/11 14:19:30 INFO namenode.FSEditLog: close success: truncate to 4, editlog=/tmp/hadoop/data/dfs/name/current/edits
13/08/11 14:19:30 INFO common.Storage: Storage directory /tmp/hadoop/data/dfs/name has been successfully formatted.
13/08/11 14:19:30 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at ip-172-31-12-208/172.31.12.208
************************************************************/
[root@ip-172-31-12-208 hadoop-1.2.1]#
```

41. Now it's time to start your Hadoop Single Node Cluster with following command

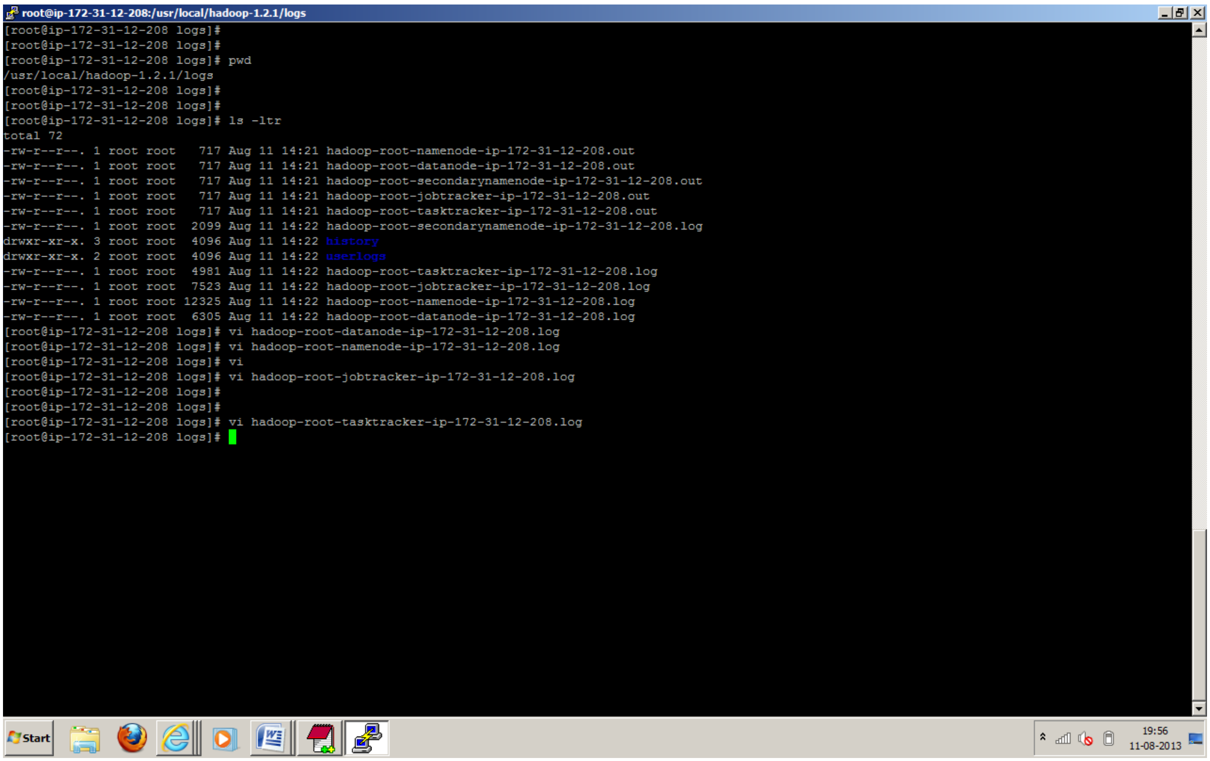
```
./start-all.sh
```



root@ip-172-31-12-208:/usr/local/hadoop-1.2.1/bin

```
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]# pwd
/usr/local/hadoop-1.2.1/bin
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]# ./start-all.sh
Warning: $HADOOP_HOME is deprecated.

starting namenode, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-namenode-ip-172-31-12-208.out
localhost: Warning: $HADOOP_HOME is deprecated.
localhost:
localhost: starting datanode, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-datanode-ip-172-31-12-208.out
localhost: Warning: $HADOOP_HOME is deprecated.
localhost:
localhost: starting secondarynamenode, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-secondarynamenode-ip-172-31-12-208.out
starting jobtracker, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-jobtracker-ip-172-31-12-208.out
localhost: Warning: $HADOOP_HOME is deprecated.
localhost:
localhost: starting tasktracker, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-tasktracker-ip-172-31-12-208.out
[root@ip-172-31-12-208 bin]#
```

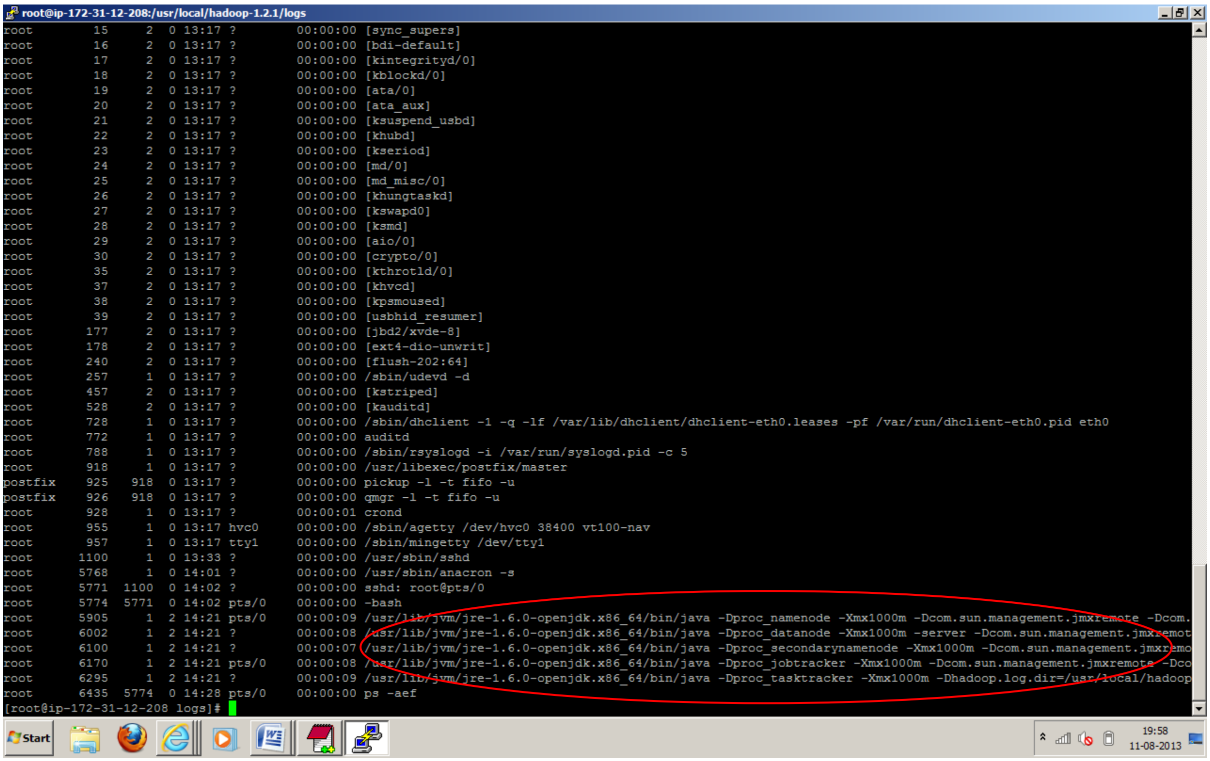
42. Check the files under log directory to check whether everything started properly, there should not be exception.



```
[root@ip-172-31-12-208 logs]#
[root@ip-172-31-12-208 logs]#
[root@ip-172-31-12-208 logs]# pwd
/usr/local/hadoop-1.2.1/logs
[root@ip-172-31-12-208 logs]#
[root@ip-172-31-12-208 logs]#
[root@ip-172-31-12-208 logs]# ls -ltr
total 72
-rw-r--r--. 1 root root   717 Aug 11 14:21 hadoop-root-namenode-ip-172-31-12-208.out
-rw-r--r--. 1 root root   717 Aug 11 14:21 hadoop-root-datanode-ip-172-31-12-208.out
-rw-r--r--. 1 root root   717 Aug 11 14:21 hadoop-root-secondarynamenode-ip-172-31-12-208.out
-rw-r--r--. 1 root root   717 Aug 11 14:21 hadoop-root-jobtracker-ip-172-31-12-208.out
-rw-r--r--. 1 root root   717 Aug 11 14:21 hadoop-root-tasktracker-ip-172-31-12-208.out
-rw-r--r--. 1 root root  2099 Aug 11 14:22 hadoop-root-secondarynamenode-ip-172-31-12-208.log
drwxr-xr-x. 3 root root  4096 Aug 11 14:22 history
drwxr-xr-x. 2 root root  4096 Aug 11 14:22 userlogs
-rw-r--r--. 1 root root  4981 Aug 11 14:22 hadoop-root-tasktracker-ip-172-31-12-208.log
-rw-r--r--. 1 root root  7523 Aug 11 14:22 hadoop-root-jobtracker-ip-172-31-12-208.log
-rw-r--r--. 1 root root 12325 Aug 11 14:22 hadoop-root-namenode-ip-172-31-12-208.log
-rw-r--r--. 1 root root  6305 Aug 11 14:22 hadoop-root-datanode-ip-172-31-12-208.log
[root@ip-172-31-12-208 logs]# vi hadoop-root-datanode-ip-172-31-12-208.log
[root@ip-172-31-12-208 logs]# vi hadoop-root-namenode-ip-172-31-12-208.log
[root@ip-172-31-12-208 logs]# vi
[root@ip-172-31-12-208 logs]# vi hadoop-root-jobtracker-ip-172-31-12-208.log
[root@ip-172-31-12-208 logs]#
[root@ip-172-31-12-208 logs]#
[root@ip-172-31-12-208 logs]# vi hadoop-root-tasktracker-ip-172-31-12-208.log
[root@ip-172-31-12-208 logs]#
```

43. Using following command will help you to get the all running Hadoop Daemon process

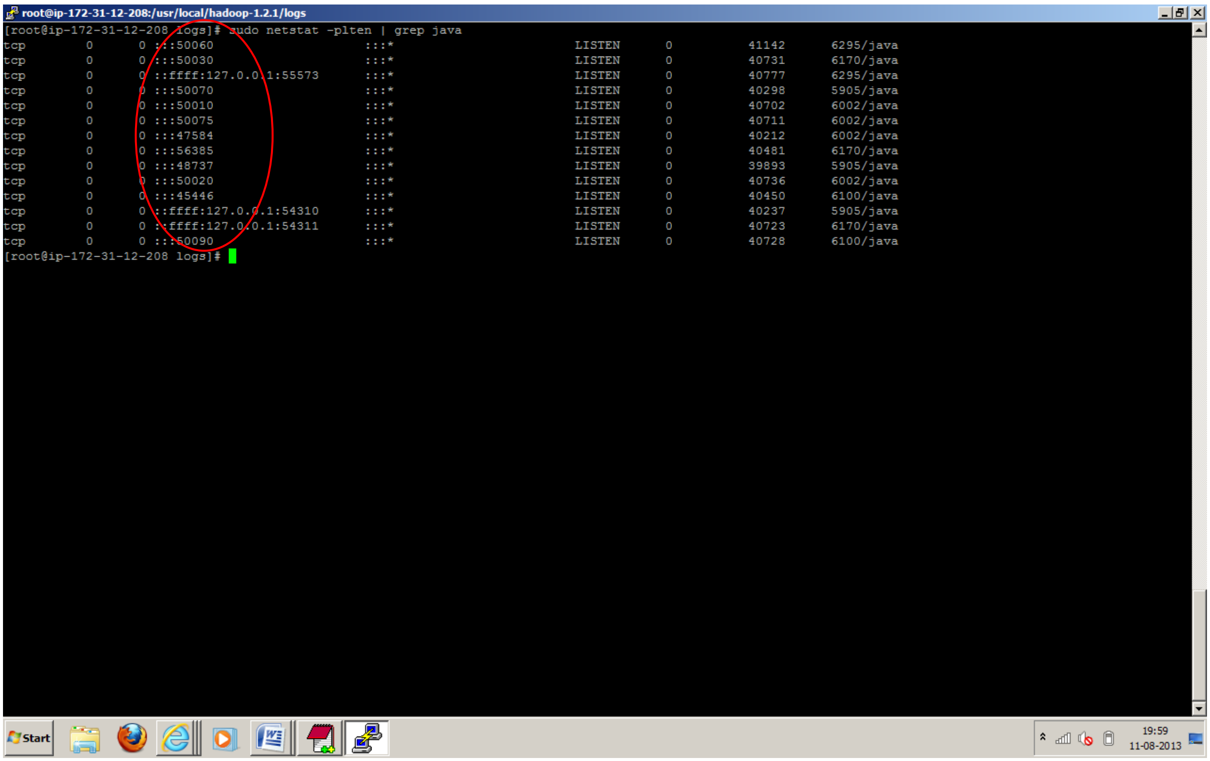
```
ps -aef
```



```
root        15     2  0 13:17 ?        00:00:00 [sync_supers]
root        16     2  0 13:17 ?        00:00:00 [bdi-default]
root        17     2  0 13:17 ?        00:00:00 [kintegrityd/0]
root        18     2  0 13:17 ?        00:00:00 [kblockd/0]
root        19     2  0 13:17 ?        00:00:00 [ata/0]
root        20     2  0 13:17 ?        00:00:00 [ata_aux]
root        21     2  0 13:17 ?        00:00:00 [ksuspend_usbd]
root        22     2  0 13:17 ?        00:00:00 [khubd]
root        23     2  0 13:17 ?        00:00:00 [kseriod]
root        24     2  0 13:17 ?        00:00:00 [md/0]
root        25     2  0 13:17 ?        00:00:00 [md_misc/0]
root        26     2  0 13:17 ?        00:00:00 [khungtaskd]
root        27     2  0 13:17 ?        00:00:00 [kswapd0]
root        28     2  0 13:17 ?        00:00:00 [ksmd]
root        29     2  0 13:17 ?        00:00:00 [aio/0]
root        30     2  0 13:17 ?        00:00:00 [crypto/0]
root        35     2  0 13:17 ?        00:00:00 [kthrotld/0]
root        37     2  0 13:17 ?        00:00:00 [khvcd]
root        38     2  0 13:17 ?        00:00:00 [kpsmoused]
root        39     2  0 13:17 ?        00:00:00 [usbhid_resumer]
root       177     2  0 13:17 ?        00:00:00 [jbd2/xvde-8]
root       178     2  0 13:17 ?        00:00:00 [ext4-dio-unwrit]
root       240     2  0 13:17 ?        00:00:00 [flush-202:64]
root       257     1  0 13:17 ?        00:00:00 /sbin/udevd -d
root       457     2  0 13:17 ?        00:00:00 [kstriped]
root       528     2  0 13:17 ?        00:00:00 [kauditd]
root       728     1  0 13:17 ?        00:00:00 /sbin/dhclient -1 -q -lf /var/lib/dhclient/dhclient-eth0.leases -pf /var/run/dhclient-eth0.pid eth0
root       772     1  0 13:17 ?        00:00:00 auditd
root       788     1  0 13:17 ?        00:00:00 /sbin/rsyslogd -i /var/run/syslogd.pid -c 5
root       918     1  0 13:17 ?        00:00:00 /usr/libexec/postfix/master
postfix    925   918  0 13:17 ?        00:00:00 pickup -l -t fifo -u
postfix    926   918  0 13:17 ?        00:00:00 qmgr -l -t fifo -u
root       928     1  0 13:17 ?        00:00:01 crond
root       955     1  0 13:17 hvc0     00:00:00 /sbin/agetty /dev/hvc0 38400 vt100-nav
root       957     1  0 13:17 tty1     00:00:00 /sbin/mingetty /dev/tty1
root      1100     1  0 13:33 ?        00:00:00 /usr/sbin/sshd
root      5768     1  0 14:01 ?        00:00:00 /usr/sbin/anacron -s
root      5771  1100  0 14:02 ?        00:00:00 sshd: root@pts/0
root      5774  5771  0 14:02 pts/0    00:00:00 -bash
root      5905     1  2 14:21 pts/0    00:00:09 /usr/lib/jvm/jre-1.6.0-openjdk.x86_64/bin/java -Dproc_namenode -Xmx1000m -Dcom.sun.management.jmxremote -Dcom.
root      6002     1  2 14:21 ?        00:00:08 /usr/lib/jvm/jre-1.6.0-openjdk.x86_64/bin/java -Dproc_datanode -Xmx1000m -server -Dcom.sun.management.jmxremot
root      6100     1  2 14:21 ?        00:00:07 /usr/lib/jvm/jre-1.6.0-openjdk.x86_64/bin/java -Dproc_secondarynamenode -Xmx1000m -Dcom.sun.management.jmxremo
root      6170     1  2 14:21 pts/0    00:00:08 /usr/lib/jvm/jre-1.6.0-openjdk.x86_64/bin/java -Dproc_jobtracker -Xmx1000m -Dcom.sun.management.jmxremote -Dco
root      6295     1  2 14:21 ?        00:00:09 /usr/lib/jvm/jre-1.6.0-openjdk.x86_64/bin/java -Dproc_tasktracker -Xmx1000m -Dhadoop.log.dir=/usr/local/hadoop
root      6435  5774  0 14:28 pts/0    00:00:00 ps -aef
[root@ip-172-31-12-208 logs]#
```

44. Check all the port which are being used with following command.

```
sudo netstat -plten | grep java
```

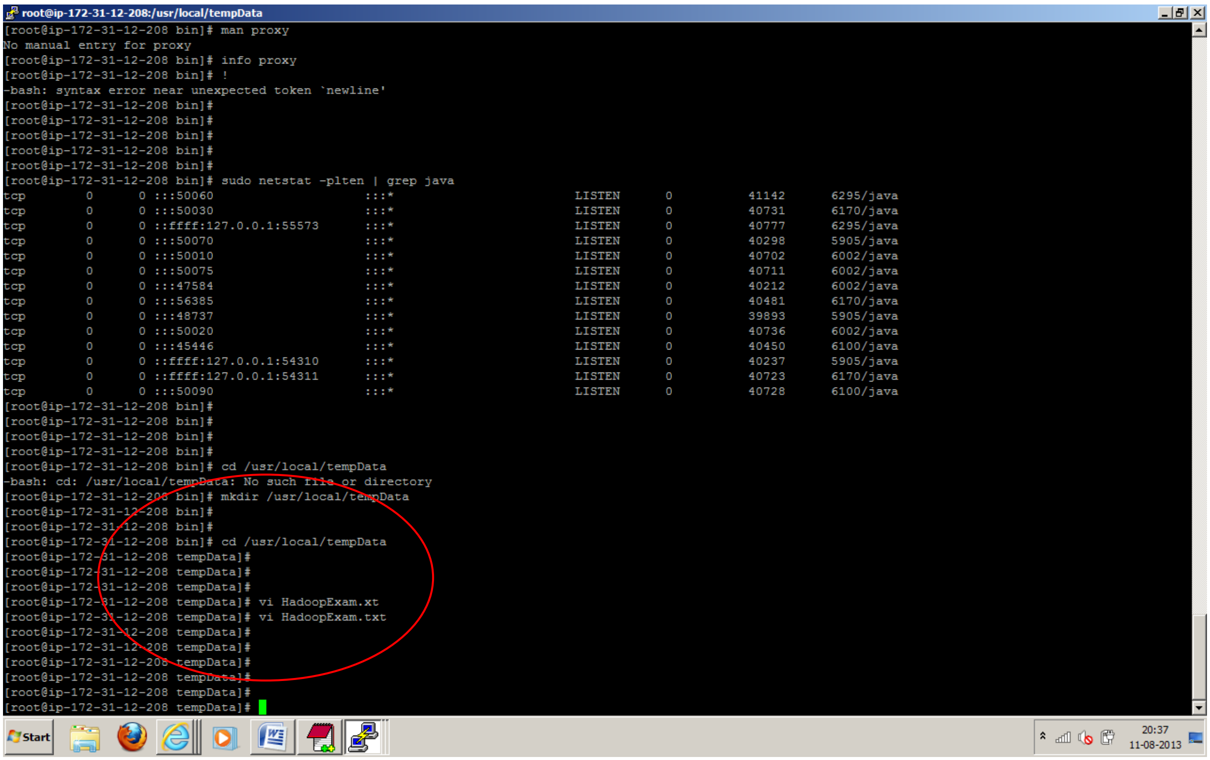


root@ip-172-31-12-208:/usr/local/hadoop-1.2.1/logs

```
[root@ip-172-31-12-208 logs]# sudo netstat -plten | grep java
tcp        0      0 :::50060                    :::*                        LISTEN      0          41142      6295/java
tcp        0      0 :::50030                    :::*                        LISTEN      0          40731      6170/java
tcp        0      0 ::ffff:127.0.0.1:55573      :::*                        LISTEN      0          40777      6295/java
tcp        0      0 :::50070                    :::*                        LISTEN      0          40298      5905/java
tcp        0      0 :::50010                    :::*                        LISTEN      0          40702      6002/java
tcp        0      0 :::50075                    :::*                        LISTEN      0          40711      6002/java
tcp        0      0 :::47584                    :::*                        LISTEN      0          40212      6002/java
tcp        0      0 :::56385                    :::*                        LISTEN      0          40481      6170/java
tcp        0      0 :::48737                    :::*                        LISTEN      0          39893      5905/java
tcp        0      0 :::50020                    :::*                        LISTEN      0          40736      6002/java
tcp        0      0 :::45446                    :::*                        LISTEN      0          40450      6100/java
tcp        0      0 ::ffff:127.0.0.1:54310      :::*                        LISTEN      0          40237      5905/java
tcp        0      0 ::ffff:127.0.0.1:54311      :::*                        LISTEN      0          40723      6170/java
tcp        0      0 :::50090                    :::*                        LISTEN      0          40728      6100/java
[root@ip-172-31-12-208 logs]#
```

45. Now it's time to run Hadoop Word Count Example which is comes with the Apache Hadoop Installer. Create a Dummy file called HadoopExam.txt under some /usr/local/tempData directory. With lot of words in it.

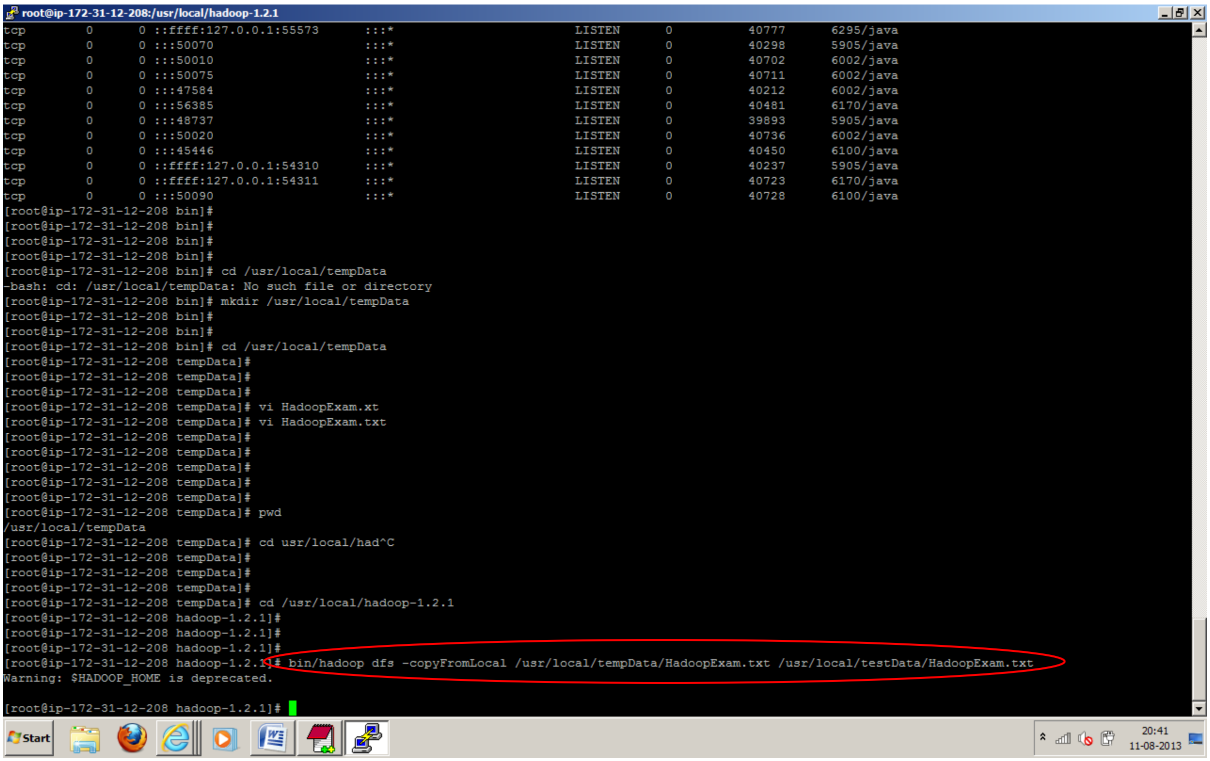
```
vi /usr/local/tempData/HadoopExam.txt
```



```
[root@ip-172-31-12-208 bin]# man proxy
No manual entry for proxy
[root@ip-172-31-12-208 bin]# info proxy
[root@ip-172-31-12-208 bin]# !
-bash: syntax error near unexpected token `newline'
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]# sudo netstat -plten | grep java
tcp        0      0 :::50060                    :::*                        LISTEN      0          41142      6295/java
tcp        0      0 :::50030                    :::*                        LISTEN      0          40731      6170/java
tcp        0      0 ::ffff:127.0.0.1:55573      :::*                        LISTEN      0          40777      6295/java
tcp        0      0 :::50070                    :::*                        LISTEN      0          40298      5905/java
tcp        0      0 :::50010                    :::*                        LISTEN      0          40702      6002/java
tcp        0      0 :::50075                    :::*                        LISTEN      0          40711      6002/java
tcp        0      0 :::47584                    :::*                        LISTEN      0          40212      6002/java
tcp        0      0 :::56385                    :::*                        LISTEN      0          40481      6170/java
tcp        0      0 :::48737                    :::*                        LISTEN      0          39893      5905/java
tcp        0      0 :::50020                    :::*                        LISTEN      0          40736      6002/java
tcp        0      0 :::45446                    :::*                        LISTEN      0          40450      6100/java
tcp        0      0 ::ffff:127.0.0.1:54310      :::*                        LISTEN      0          40237      5905/java
tcp        0      0 ::ffff:127.0.0.1:54311      :::*                        LISTEN      0          40723      6170/java
tcp        0      0 :::50090                    :::*                        LISTEN      0          40728      6100/java
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]# cd /usr/local/tempData
-bash: cd: /usr/local/tempData: No such file or directory
[root@ip-172-31-12-208 bin]# mkdir /usr/local/tempData
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]# cd /usr/local/tempData
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]# vi HadoopExam.xt
[root@ip-172-31-12-208 tempData]# vi HadoopExam.txt
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
```

46. You have created this file in your local instance, now we need to copy this file in the hdfs filesystem, so Hadoop mapred framework can read that file for counting the words in the file. Use following command to do that..

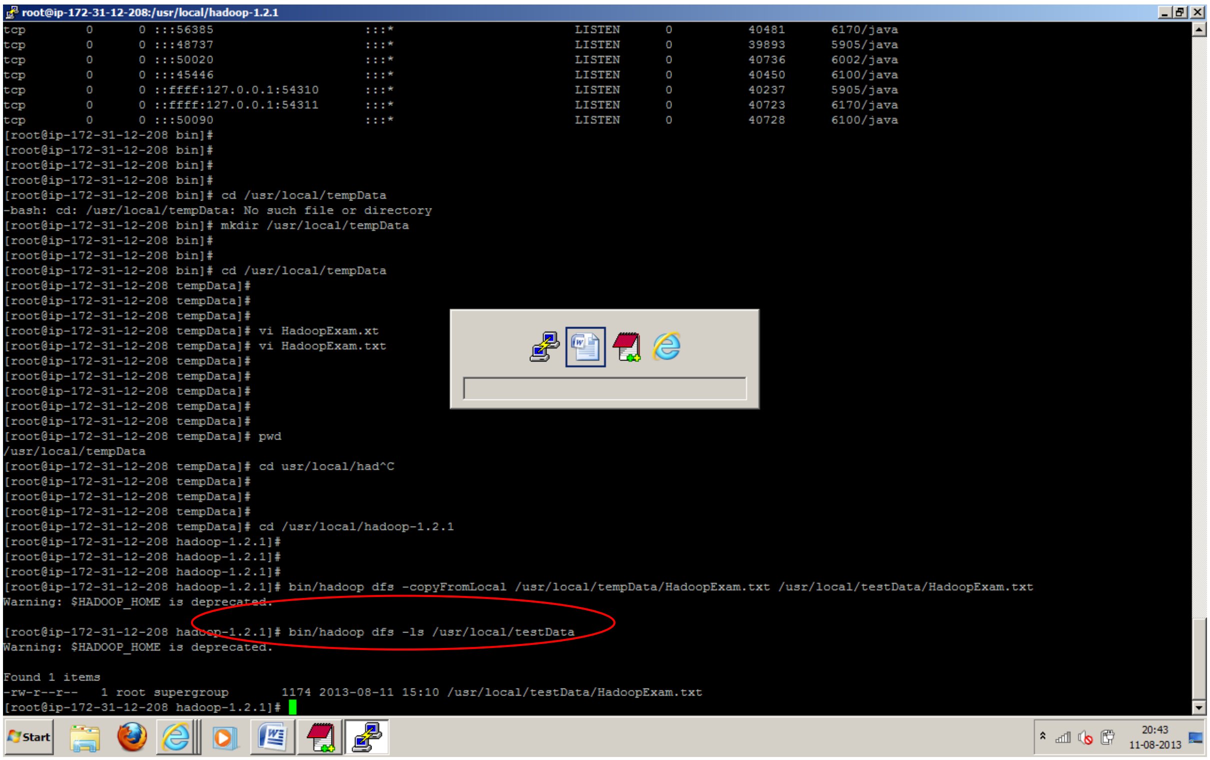
```
bin/hadoop dfs -copyFromLocal /usr/local/tempData/HadoopExam.txt
/usr/local/testData/HadoopExam.txt
```



```
tcp        0      0 ::ffff:127.0.0.1:55573      :::*                        LISTEN      0          40777      6295/java
tcp        0      0 :::50070                    :::*                        LISTEN      0          40298      5905/java
tcp        0      0 :::50010                    :::*                        LISTEN      0          40702      6002/java
tcp        0      0 :::50075                    :::*                        LISTEN      0          40711      6002/java
tcp        0      0 :::47584                    :::*                        LISTEN      0          40212      6002/java
tcp        0      0 :::56385                    :::*                        LISTEN      0          40481      6170/java
tcp        0      0 :::48737                    :::*                        LISTEN      0          39893      5905/java
tcp        0      0 :::50020                    :::*                        LISTEN      0          40736      6002/java
tcp        0      0 :::45446                    :::*                        LISTEN      0          40450      6100/java
tcp        0      0 ::ffff:127.0.0.1:54310      :::*                        LISTEN      0          40237      5905/java
tcp        0      0 ::ffff:127.0.0.1:54311      :::*                        LISTEN      0          40723      6170/java
tcp        0      0 :::50090                    :::*                        LISTEN      0          40728      6100/java
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]# cd /usr/local/tempData
-bash: cd: /usr/local/tempData: No such file or directory
[root@ip-172-31-12-208 bin]# mkdir /usr/local/tempData
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]# cd /usr/local/tempData
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]# vi HadoopExam.xt
[root@ip-172-31-12-208 tempData]# vi HadoopExam.txt
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]# pwd
/usr/local/tempData
[root@ip-172-31-12-208 tempData]# cd usr/local/had^C
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]# cd /usr/local/hadoop-1.2.1
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]# bin/hadoop dfs -copyFromLocal /usr/local/tempData/HadoopExam.txt /usr/local/testData/HadoopExam.txt
Warning: $HADOOP_HOME is deprecated.

[root@ip-172-31-12-208 hadoop-1.2.1]#
```

47. Now check whether the file copied in the Hadoop cluster with following commands.

```
bin/hadoop dfs -ls /usr/local/testData
```

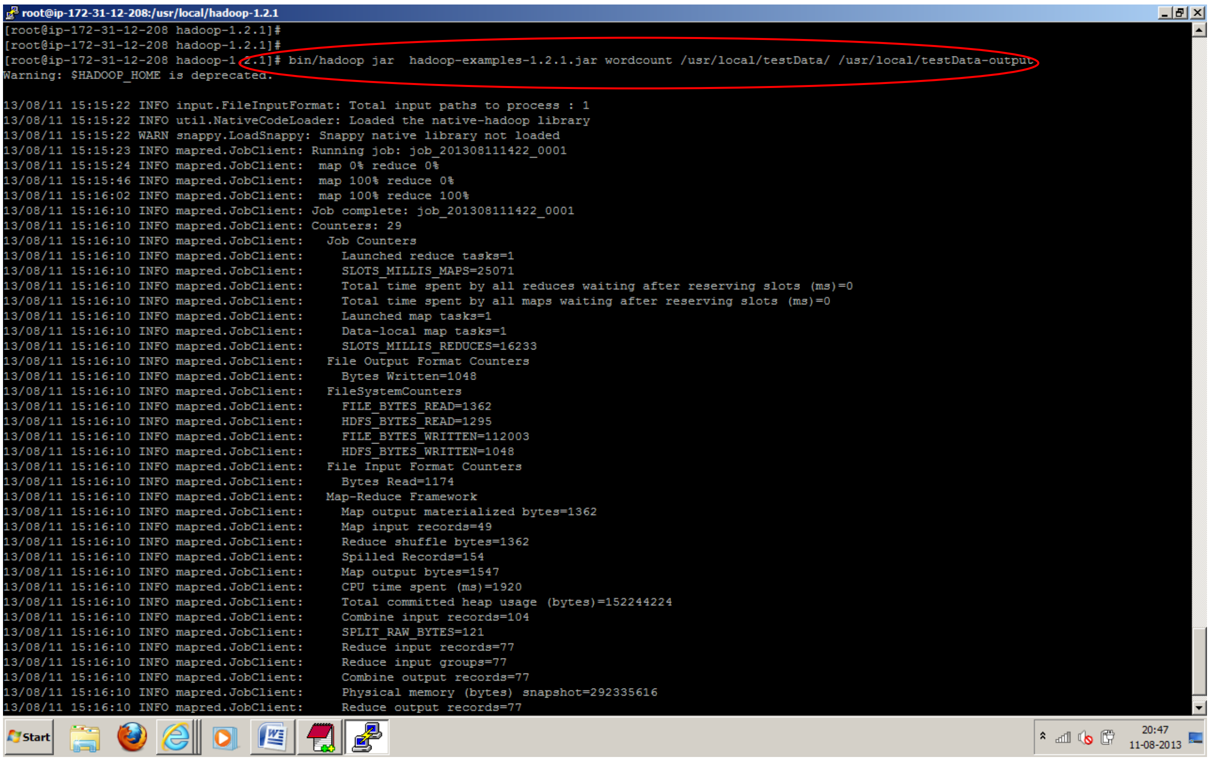


root@ip-172-31-12-208:/usr/local/hadoop-1.2.1

```
tcp        0      0 :::56385                    :::*                        LISTEN      0          40481      6170/java
tcp        0      0 :::48737                    :::*                        LISTEN      0          39893      5905/java
tcp        0      0 :::50020                    :::*                        LISTEN      0          40736      6002/java
tcp        0      0 :::45446                    :::*                        LISTEN      0          40450      6100/java
tcp        0      0 ::ffff:127.0.0.1:54310      :::*                        LISTEN      0          40237      5905/java
tcp        0      0 ::ffff:127.0.0.1:54311      :::*                        LISTEN      0          40723      6170/java
tcp        0      0 :::50090                    :::*                        LISTEN      0          40728      6100/java
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]# cd /usr/local/tempData
-bash: cd: /usr/local/tempData: No such file or directory
[root@ip-172-31-12-208 bin]# mkdir /usr/local/tempData
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]#
[root@ip-172-31-12-208 bin]# cd /usr/local/tempData
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]# vi HadoopExam.xt
[root@ip-172-31-12-208 tempData]# vi HadoopExam.txt
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]# pwd
/usr/local/tempData
[root@ip-172-31-12-208 tempData]# cd usr/local/had^C
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]#
[root@ip-172-31-12-208 tempData]# cd /usr/local/hadoop-1.2.1
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]# bin/hadoop dfs -copyFromLocal /usr/local/tempData/HadoopExam.txt /usr/local/testData/HadoopExam.txt
Warning: $HADOOP_HOME is deprecated.

[root@ip-172-31-12-208 hadoop-1.2.1]# bin/hadoop dfs -ls /usr/local/testData
Warning: $HADOOP_HOME is deprecated.

Found 1 items
-rw-r--r--   1 root supergroup       1174 2013-08-11 15:10 /usr/local/testData/HadoopExam.txt
[root@ip-172-31-12-208 hadoop-1.2.1]#
```

48. Now run the MapReduce word count example, with following command which will launch map and reduce task and finally will write the output.

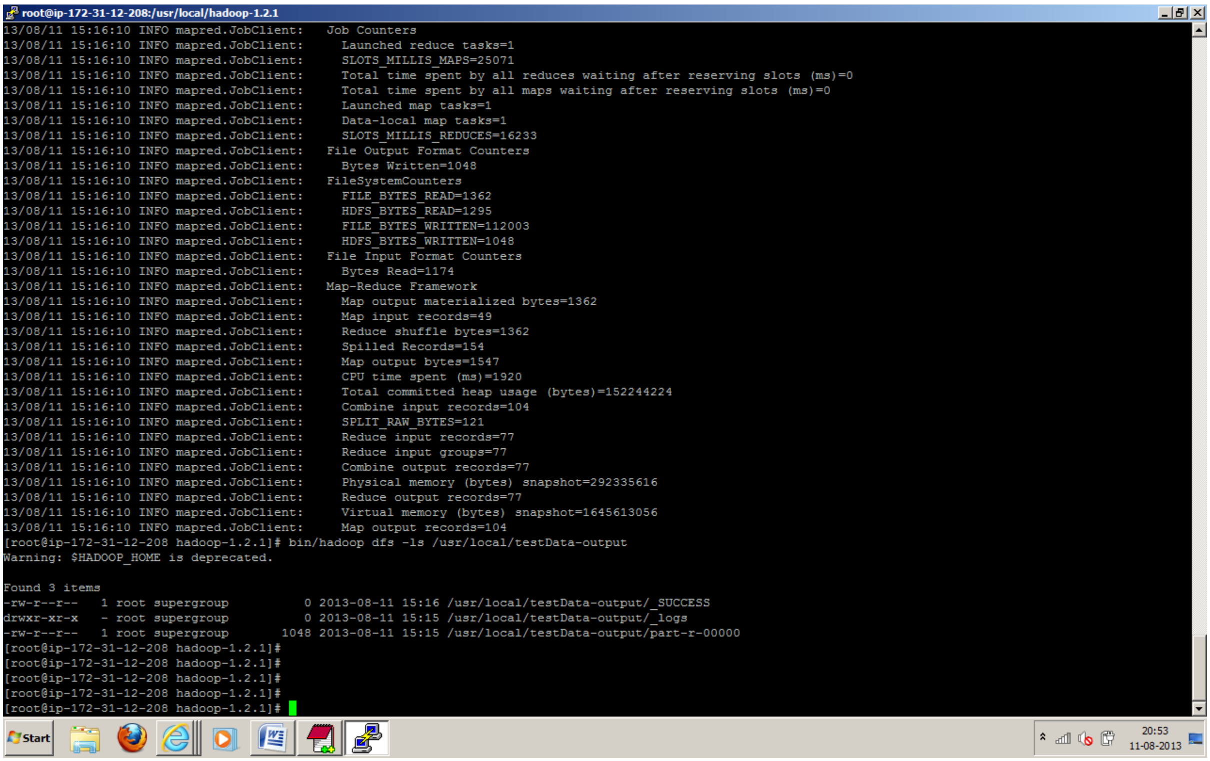
```
bin/hadoop jar  hadoop-examples-1.2.1.jar wordcount /usr/local/testData/
/usr/local/testData-output
```



```
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]# bin/hadoop jar  hadoop-examples-1.2.1.jar wordcount /usr/local/testData/ /usr/local/testData-output
Warning: $HADOOP_HOME is deprecated.

13/08/11 15:15:22 INFO input.FileInputFormat: Total input paths to process : 1
13/08/11 15:15:22 INFO util.NativeCodeLoader: Loaded the native-hadoop library
13/08/11 15:15:22 WARN snappy.LoadSnappy: Snappy native library not loaded
13/08/11 15:15:23 INFO mapred.JobClient: Running job: job_201308111422_0001
13/08/11 15:15:24 INFO mapred.JobClient:  map 0% reduce 0%
13/08/11 15:15:46 INFO mapred.JobClient:  map 100% reduce 0%
13/08/11 15:16:02 INFO mapred.JobClient:  map 100% reduce 100%
13/08/11 15:16:10 INFO mapred.JobClient: Job complete: job_201308111422_0001
13/08/11 15:16:10 INFO mapred.JobClient: Counters: 29
13/08/11 15:16:10 INFO mapred.JobClient:   Job Counters
13/08/11 15:16:10 INFO mapred.JobClient:     Launched reduce tasks=1
13/08/11 15:16:10 INFO mapred.JobClient:     SLOTS_MILLIS_MAPS=25071
13/08/11 15:16:10 INFO mapred.JobClient:     Total time spent by all reduces waiting after reserving slots (ms)=0
13/08/11 15:16:10 INFO mapred.JobClient:     Total time spent by all maps waiting after reserving slots (ms)=0
13/08/11 15:16:10 INFO mapred.JobClient:     Launched map tasks=1
13/08/11 15:16:10 INFO mapred.JobClient:     Data-local map tasks=1
13/08/11 15:16:10 INFO mapred.JobClient:     SLOTS_MILLIS_REDUCES=16233
13/08/11 15:16:10 INFO mapred.JobClient:   File Output Format Counters
13/08/11 15:16:10 INFO mapred.JobClient:     Bytes Written=1048
13/08/11 15:16:10 INFO mapred.JobClient:   FileSystemCounters
13/08/11 15:16:10 INFO mapred.JobClient:     FILE_BYTES_READ=1362
13/08/11 15:16:10 INFO mapred.JobClient:     HDFS_BYTES_READ=1295
13/08/11 15:16:10 INFO mapred.JobClient:     FILE_BYTES_WRITTEN=112003
13/08/11 15:16:10 INFO mapred.JobClient:     HDFS_BYTES_WRITTEN=1048
13/08/11 15:16:10 INFO mapred.JobClient:   File Input Format Counters
13/08/11 15:16:10 INFO mapred.JobClient:     Bytes Read=1174
13/08/11 15:16:10 INFO mapred.JobClient:   Map-Reduce Framework
13/08/11 15:16:10 INFO mapred.JobClient:     Map output materialized bytes=1362
13/08/11 15:16:10 INFO mapred.JobClient:     Map input records=49
13/08/11 15:16:10 INFO mapred.JobClient:     Reduce shuffle bytes=1362
13/08/11 15:16:10 INFO mapred.JobClient:     Spilled Records=154
13/08/11 15:16:10 INFO mapred.JobClient:     Map output bytes=1547
13/08/11 15:16:10 INFO mapred.JobClient:     CPU time spent (ms)=1920
13/08/11 15:16:10 INFO mapred.JobClient:     Total committed heap usage (bytes)=152244224
13/08/11 15:16:10 INFO mapred.JobClient:     Combine input records=104
13/08/11 15:16:10 INFO mapred.JobClient:     SPLIT_RAW_BYTES=121
13/08/11 15:16:10 INFO mapred.JobClient:     Reduce input records=77
13/08/11 15:16:10 INFO mapred.JobClient:     Reduce input groups=77
13/08/11 15:16:10 INFO mapred.JobClient:     Combine output records=77
13/08/11 15:16:10 INFO mapred.JobClient:     Physical memory (bytes) snapshot=292335616
13/08/11 15:16:10 INFO mapred.JobClient:     Reduce output records=77
```

49. Check the output generated

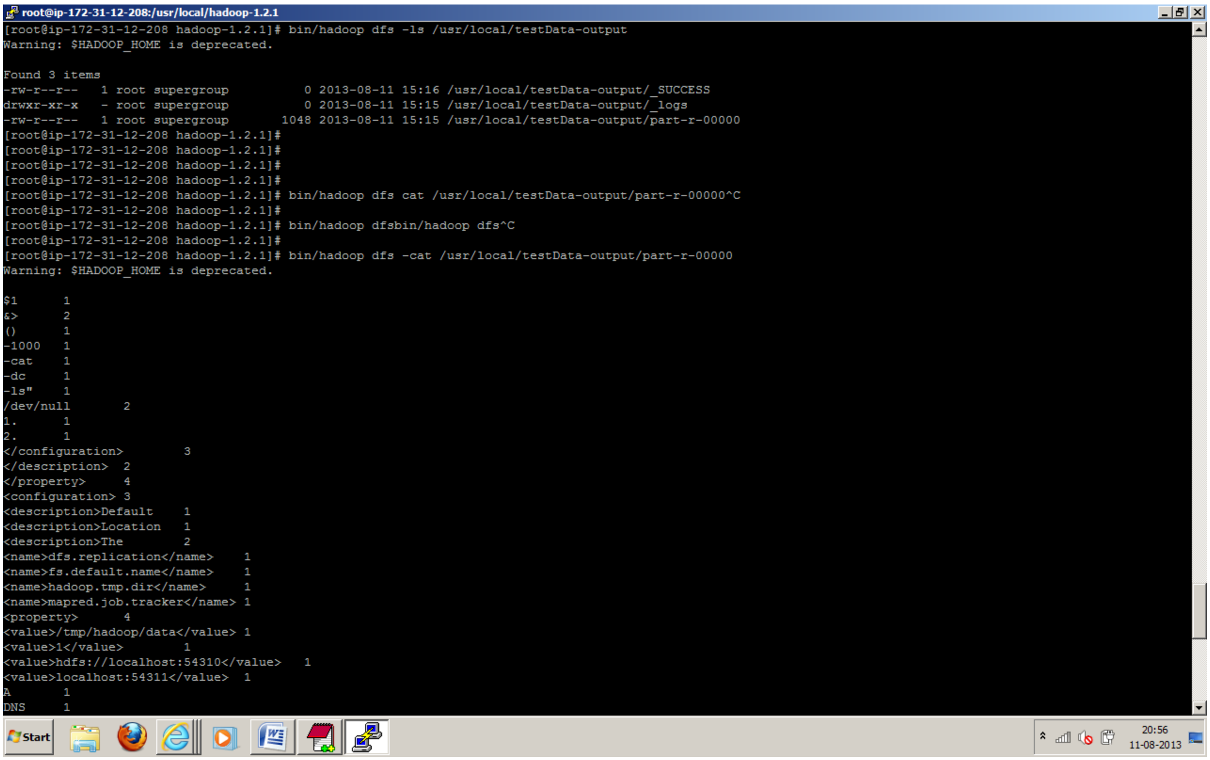
```
bin/hadoop dfs -ls /usr/local/testData-output
```



```
13/08/11 15:16:10 INFO mapred.JobClient:   Job Counters
13/08/11 15:16:10 INFO mapred.JobClient:     Launched reduce tasks=1
13/08/11 15:16:10 INFO mapred.JobClient:     SLOTS_MILLIS_MAPS=25071
13/08/11 15:16:10 INFO mapred.JobClient:     Total time spent by all reduces waiting after reserving slots (ms)=0
13/08/11 15:16:10 INFO mapred.JobClient:     Total time spent by all maps waiting after reserving slots (ms)=0
13/08/11 15:16:10 INFO mapred.JobClient:     Launched map tasks=1
13/08/11 15:16:10 INFO mapred.JobClient:     Data-local map tasks=1
13/08/11 15:16:10 INFO mapred.JobClient:     SLOTS_MILLIS_REDUCES=16233
13/08/11 15:16:10 INFO mapred.JobClient:   File Output Format Counters
13/08/11 15:16:10 INFO mapred.JobClient:     Bytes Written=1048
13/08/11 15:16:10 INFO mapred.JobClient:   FileSystemCounters
13/08/11 15:16:10 INFO mapred.JobClient:     FILE_BYTES_READ=1362
13/08/11 15:16:10 INFO mapred.JobClient:     HDFS_BYTES_READ=1295
13/08/11 15:16:10 INFO mapred.JobClient:     FILE_BYTES_WRITTEN=112003
13/08/11 15:16:10 INFO mapred.JobClient:     HDFS_BYTES_WRITTEN=1048
13/08/11 15:16:10 INFO mapred.JobClient:   File Input Format Counters
13/08/11 15:16:10 INFO mapred.JobClient:     Bytes Read=1174
13/08/11 15:16:10 INFO mapred.JobClient:   Map-Reduce Framework
13/08/11 15:16:10 INFO mapred.JobClient:     Map output materialized bytes=1362
13/08/11 15:16:10 INFO mapred.JobClient:     Map input records=49
13/08/11 15:16:10 INFO mapred.JobClient:     Reduce shuffle bytes=1362
13/08/11 15:16:10 INFO mapred.JobClient:     Spilled Records=154
13/08/11 15:16:10 INFO mapred.JobClient:     Map output bytes=1547
13/08/11 15:16:10 INFO mapred.JobClient:     CPU time spent (ms)=1920
13/08/11 15:16:10 INFO mapred.JobClient:     Total committed heap usage (bytes)=152244224
13/08/11 15:16:10 INFO mapred.JobClient:     Combine input records=104
13/08/11 15:16:10 INFO mapred.JobClient:     SPLIT_RAW_BYTES=121
13/08/11 15:16:10 INFO mapred.JobClient:     Reduce input records=77
13/08/11 15:16:10 INFO mapred.JobClient:     Reduce input groups=77
13/08/11 15:16:10 INFO mapred.JobClient:     Combine output records=77
13/08/11 15:16:10 INFO mapred.JobClient:     Physical memory (bytes) snapshot=292335616
13/08/11 15:16:10 INFO mapred.JobClient:     Reduce output records=77
13/08/11 15:16:10 INFO mapred.JobClient:     Virtual memory (bytes) snapshot=1645613056
13/08/11 15:16:10 INFO mapred.JobClient:     Map output records=104
[root@ip-172-31-12-208 hadoop-1.2.1]# bin/hadoop dfs -ls /usr/local/testData-output
Warning: $HADOOP_HOME is deprecated.

Found 3 items
-rw-r--r--   1 root supergroup          0 2013-08-11 15:16 /usr/local/testData-output/_SUCCESS
drwxr-xr-x   - root supergroup          0 2013-08-11 15:15 /usr/local/testData-output/_logs
-rw-r--r--   1 root supergroup       1048 2013-08-11 15:15 /usr/local/testData-output/part-r-00000
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]#
```

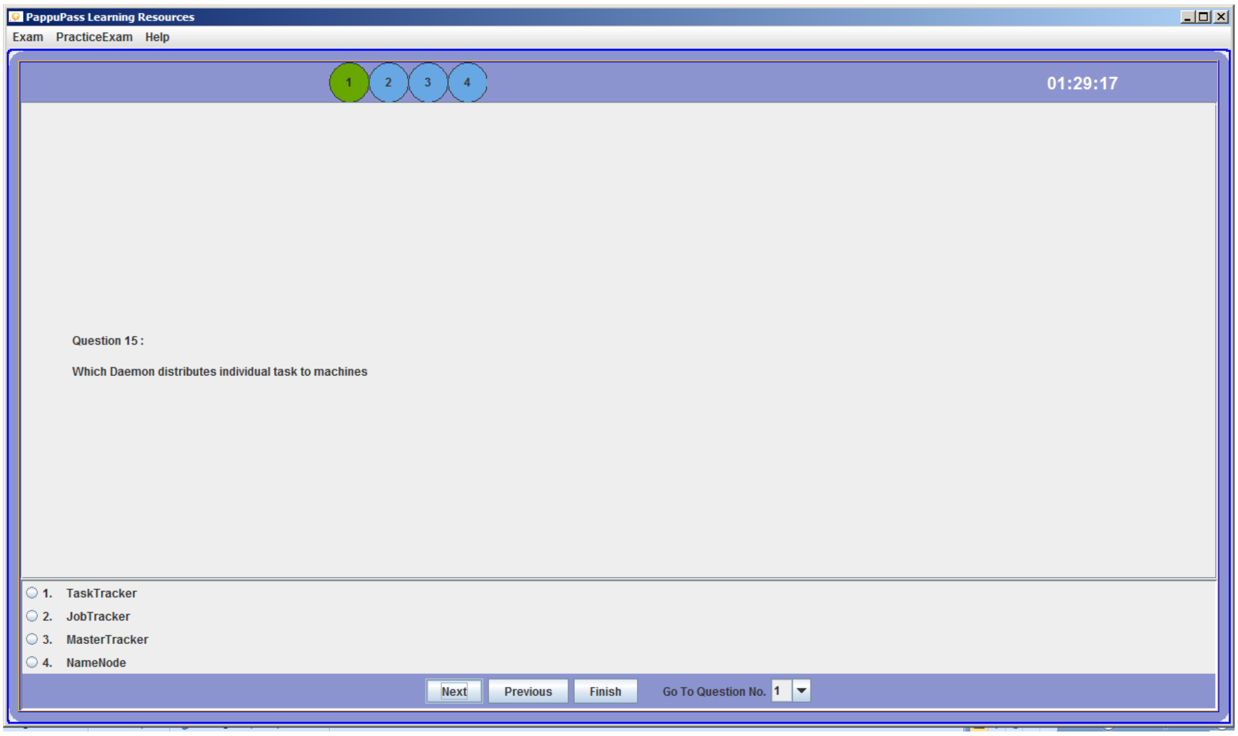
50. Now view the content in the ouput of word count program

```
bin/hadoop dfs -cat /usr/local/testData-output/part-r-00000
```



```
[root@ip-172-31-12-208 hadoop-1.2.1]# bin/hadoop dfs -ls /usr/local/testData-output
Warning: $HADOOP_HOME is deprecated.

Found 3 items
-rw-r--r--   1 root supergroup          0 2013-08-11 15:16 /usr/local/testData-output/_SUCCESS
drwxr-xr-x   - root supergroup          0 2013-08-11 15:15 /usr/local/testData-output/_logs
-rw-r--r--   1 root supergroup       1048 2013-08-11 15:15 /usr/local/testData-output/part-r-00000
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]# bin/hadoop dfs cat /usr/local/testData-output/part-r-00000^C
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]# bin/hadoop dfsbin/hadoop dfs^C
[root@ip-172-31-12-208 hadoop-1.2.1]#
[root@ip-172-31-12-208 hadoop-1.2.1]# bin/hadoop dfs -cat /usr/local/testData-output/part-r-00000
Warning: $HADOOP_HOME is deprecated.

$1      1
&>      2
()      1
-1000   1
-cat    1
-dc     1
-ls"    1
/dev/null       2
1.      1
2.      1
</configuration>        3
</description>  2
</property>     4
<configuration> 3
<description>Default    1
<description>Location   1
<description>The        2
<name>dfs.replication</name>    1
<name>fs.default.name</name>    1
<name>hadoop.tmp.dir</name>     1
<name>mapred.job.tracker</name> 1
<property>      4
<value>/tmp/hadoop/data</value> 1
<value>1</value>        1
<value>hdfs://localhost:54310</value>   1
<value>localhost:54311</value>  1
A       1
DNS     1
```

51. Now Terminate all the instances once you are done, else it would incur cost even you are not using instances. Simple step to terminate select the instance and under the action select for terminate.
52. Happy Hadoop Learning....
53. Send your suggestions to us admin@hadoopexam.com



Question 15 :

Which Daemon distributes individual task to machines

○ 1. TaskTracker
○ 2. JobTracker
○ 3. MasterTracker
○ 4. NameNode