

# Hadoop Single Node Setup

Sorayut Glomglome

$\pi$

# Content

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



# AWS Academy

π

# AWS Academy

aws ALLv1-22427

Home Modules Discussions

## AWS Academy Learner Lab [22427]



The 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.

View Course Stream  
View Course Calendar  
View Course Notifications

To Do  
Nothing for now

Recent Feedback  
Nothing for now

π

# Learner Lab - Vocareum

The screenshot shows the AWS Lambda console interface. At the top, the path 'ALLv1-22427 > Modules' is visible. On the left, a sidebar menu includes 'aws' (with the Amazon logo), 'Account', 'Dashboard', 'Courses', 'Calendar', 'Inbox', 'History', and 'Help'. The 'Modules' option is selected and highlighted with a red arrow. In the main content area, a dropdown menu for 'Learner Lab' is open, showing three items: 'Student Guide.pdf', 'Learner Lab' (which is also highlighted with a red arrow), and 'End of Course Feedback Survey'. A 'Collapse All' button is located in the top right corner.

The screenshot shows the 'Terms and Conditions' page from the Vocareum website. The URL is 'ALLv1-22427 > Modules > Learner Lab > Learner Lab'. The page header includes the Vocareum logo and navigation links for 'Home', 'My Classes', 'Help', and 'Watanyou'. The main content instructs users to read the terms and conditions and click 'I agree' at the bottom. Below this, the 'Terms and Conditions' section is titled 'Terms and Conditions'. It welcomes users to the Vocareum website and explains that it is a web-based education and learning platform. It states that by using the services, teachers can create, customize, and administer educational courses. It also mentions that the site and platform are collectively called 'Services'. The 'Terms and Conditions' section is followed by a '1. Agreement to Terms' section, which contains the legal text of the terms and conditions. The sidebar on the left is identical to the one in the top screenshot, showing the same AWS Lambda interface.

π

# Learner Lab - Vocareum

The screenshot displays the Learner Lab interface from Vocareum, showing three main panels:

- AWS Home Panel (Left):** Shows the AWS logo and navigation links: Account, Dashboard, Courses, Calendar, Inbox, History, and Help. A red arrow points to the "AWS" status indicator.
- AWS Modules Panel (Middle Left):** Shows the AWS logo and navigation links: Home, Modules, and Discussions. A red arrow points to the "AWS" status indicator.
- Learner Lab Details Panel (Right):** Shows the "Learner Lab" title, a terminal window with the command "ddd\_v1\_w\_ENB\_1317166@runweb58405:~\$", and various status indicators:
  - AWS status: Red dot (error).
  - Used \$0 of \$100 (red arrow).
  - 03:56 (red arrow).
  - Start Lab, End Lab, AWS Details, Readme, Reset buttons.
  - EN-US language dropdown.
  - Learner Lab section with links: Environment Overview, Environment Navigation, Access the AWS Management Console, Region restriction, Service usage and other restrictions, Using the terminal in the browser, Running AWS CLI commands, Using the AWS SDK for Python, Preserving your budget, Accessing EC2 Instances, SSH Access to EC2 Instances, SSH Access from Windows, and SSH Access from a Mac.

π

# AWS Console

S | Services | Search for services, features, blogs, docs, ar [Alt+S] | N. Virginia | vocabs/

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

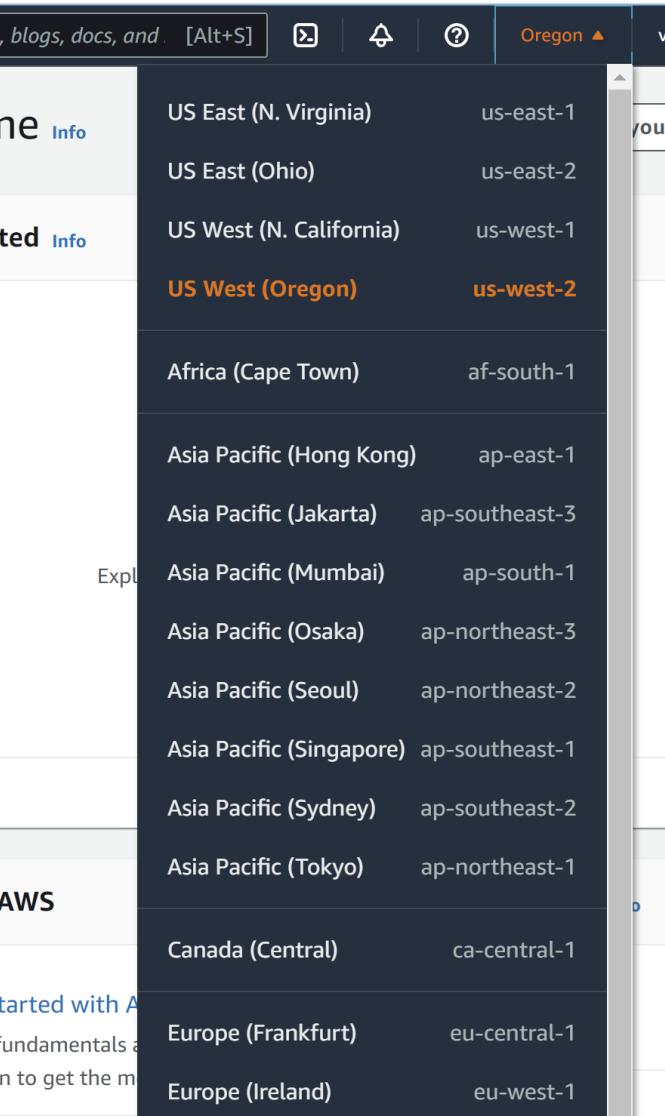
Scheduled changes 0 Upcoming and past 7 days

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

π

# AWS Regions



The screenshot shows a list of AWS Regions and their corresponding codes. The regions listed are:

| Region                   | Code             |
|--------------------------|------------------|
| US East (N. Virginia)    | us-east-1        |
| US East (Ohio)           | us-east-2        |
| US West (N. California)  | us-west-1        |
| <b>US West (Oregon)</b>  | <b>us-west-2</b> |
| Africa (Cape Town)       | af-south-1       |
| Asia Pacific (Hong Kong) | ap-east-1        |
| Asia Pacific (Jakarta)   | ap-southeast-3   |
| Explor                   | ap-south-1       |
| Asia Pacific (Mumbai)    | ap-northeast-3   |
| Asia Pacific (Osaka)     | ap-northeast-2   |
| Asia Pacific (Seoul)     | ap-southeast-1   |
| Asia Pacific (Singapore) | ap-southeast-2   |
| Asia Pacific (Sydney)    | ap-northeast-1   |
| Asia Pacific (Tokyo)     | ca-central-1     |
| Canada (Central)         | eu-central-1     |
| Europe (Frankfurt)       | eu-west-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~~

π

# Log in to Amazon Web Services

## Select a datacenter location

### Select EC2 service

The screenshot shows the AWS Management Console homepage. At the top, there's a navigation bar with the AWS logo, a 'Services' dropdown, a 'Resource Groups' dropdown, and a user profile section for 'Sorayut Glomglome' with a 'Oregon' location. A red arrow labeled '1' points to the 'Oregon' location. In the main content area, there's a 'Find Services' search bar with an example placeholder 'Example: Relational Database Service, database, RDS'. Below it, under 'Recently visited services', there are links for 'AWS Cost Explorer', 'Billing', and 'EC2'. A red circle labeled '2' with an arrow points to the 'Compute' section under 'All services', which contains a link to 'EC2'. The page is divided into several sections: 'AWS services' (with 'Compute' highlighted), 'Access resources on the go' (with a mobile phone icon), 'Explore AWS' (with sections for 'Amazon Redshift', 'Run Serverless Containers with AWS Fargate', 'Scalable, Durable, Secure Backup & Restore with Amazon S3', and 'AWS Marketplace'), and a footer with a 'Footer' link.

π

# Select Launch Instance

The screenshot shows the AWS EC2 Dashboard. On the left sidebar, under the 'INSTANCES' section, the 'Launch Instances' option is selected. The main content area displays the 'Resources' summary for the US West (Oregon) region, showing 0 Running Instances, 0 Dedicated Hosts, 1 Volumes, 2 Key Pairs, and 0 Placement Groups. Below this is a 'Create Instance' section with a 'Launch Instance' button, which is circled with a red arrow and labeled with a red number 1. To the right of the main content are 'Account Attributes' (Supported Platforms: VPC; Default VPC: vpc-72e14516), 'Additional Information' (Getting Started Guide, Documentation, All EC2 Resources, Forums, Pricing, Contact Us), and an 'AWS Marketplace' section.

You are using the following Amazon EC2 resources in the US West (Oregon) region:

|                     |                   |
|---------------------|-------------------|
| 0 Running Instances | 0 Elastic IPs     |
| 0 Dedicated Hosts   | 0 Snapshots       |
| 1 Volumes           | 0 Load Balancers  |
| 2 Key Pairs         | 2 Security Groups |
| 0 Placement Groups  |                   |

Learn more about the latest in AWS Compute from AWS re:Invent by viewing the [EC2 Videos](#).

**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 US West (Oregon) region

**Service Health**

**Service Status:**

- US West (Oregon):  US West (Oregon)

**Availability Zone Status:**

- us-west-2a: Availability zone is operating normally
- us-west-2b: Availability zone is operating normally

**Scheduled Events**

**US West (Oregon):**  
No events

**AWS Marketplace**

Find free software trial products in the AWS Marketplace from the [EC2 Launch Wizard](#). Or try these popular AMIs:

**Barracuda CloudGen Firewall for AWS - PAYG**

By Barracuda Networks, Inc.  
Rating  Starting from \$0.60/hr or from \$4,599/yr  
(12% savings) for software + AWS usage

Feedback English (US)

© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

# Select Ubuntu Server 16.04 LTS (HVM)

π

Step 1: Choose an Amazon Machine Image (AMI)

**Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-0bbe6b35405ecebdb (64-bit x86) / ami-0db180c518750ee4f (64-bit Arm)**

**Select**

64-bit (x86)  
 64-bit (Arm)

**Free tier eligible** Ubuntu Server 18.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

**Amazon RDS**

Are you launching a database instance? Try Amazon RDS.

Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale your database on AWS by automating time-consuming database management tasks. With RDS, you can easily deploy **Amazon Aurora**, **MariaDB**, **MySQL**, **Oracle**, **PostgreSQL**, and **SQL Server** databases on AWS. **Aurora** is a MySQL- and PostgreSQL-compatible, enterprise-class database at 1/10th the cost of commercial databases. [Learn more about RDS](#)

**Select**

**Ubuntu Server 16.04 LTS (HVM), SSD Volume Type - ami-076e276d85f524150 (64-bit x86) / ami-05e1b2aec3b47890f (64-bit Arm)**

**Select**

64-bit (x86)  
 64-bit (Arm)

**Free tier eligible** Ubuntu Server 16.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

**Microsoft Windows Server 2019 Base - ami-0902c15c8a8ebb717**

Microsoft Windows 2019 Datacenter edition. [English]

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

**Select**

64-bit (x86)

**Feedback** **English (US)**

© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)

π

# Select m5.large

The screenshot shows the AWS CloudFormation console interface for creating a new stack. The top navigation bar includes the AWS logo, Services dropdown, Resource Groups dropdown, and user information (Sorayut Glomglome, Oregon, Support). Below the navigation is a progress bar with steps 1 through 7. Step 2, "Choose an Instance Type," is currently active.

**Step 2: Choose an Instance Type**

|                                     | Category        | Instance Type   | Cores    | Memory (GiB) | Storage       | Networking | Monitoring       | Logs |
|-------------------------------------|-----------------|-----------------|----------|--------------|---------------|------------|------------------|------|
| <input type="checkbox"/>            | General purpose | m5a.24xlarge    | 96       | 384          | EBS only      | Yes        | 20 Gigabit       | Yes  |
| <input type="checkbox"/>            | General purpose | m5d.large       | 2        | 8            | 1 x 75 (SSD)  | Yes        | Up to 10 Gigabit | Yes  |
| <input type="checkbox"/>            | General purpose | m5d.xlarge      | 4        | 16           | 1 x 150 (SSD) | Yes        | Up to 10 Gigabit | Yes  |
| <input type="checkbox"/>            | General purpose | m5d.2xlarge     | 8        | 32           | 1 x 300 (SSD) | Yes        | Up to 10 Gigabit | Yes  |
| <input type="checkbox"/>            | General purpose | m5d.4xlarge     | 16       | 64           | 2 x 300 (SSD) | Yes        | Up to 10 Gigabit | Yes  |
| <input type="checkbox"/>            | General purpose | m5d.12xlarge    | 48       | 192          | 2 x 900 (SSD) | Yes        | 10 Gigabit       | Yes  |
| <input type="checkbox"/>            | General purpose | m5d.24xlarge    | 96       | 384          | 4 x 900 (SSD) | Yes        | 25 Gigabit       | Yes  |
| <input checked="" type="checkbox"/> | General purpose | <b>m5.large</b> | <b>2</b> | <b>8</b>     | EBS only      | Yes        | Up to 10 Gigabit | Yes  |
| <input type="checkbox"/>            | General purpose | m5.xlarge       | 4        | 16           | EBS only      | Yes        | Up to 10 Gigabit | Yes  |
| <input type="checkbox"/>            | General purpose | m5.2xlarge      | 8        | 32           | EBS only      | Yes        | Up to 10 Gigabit | Yes  |
| <input type="checkbox"/>            | General purpose | m5.4xlarge      | 16       | 64           | EBS only      | Yes        | Up to 10 Gigabit | Yes  |
| <input type="checkbox"/>            | General purpose | m5.12xlarge     | 48       | 192          | EBS only      | Yes        | 10 Gigabit       | Yes  |
| <input type="checkbox"/>            | General purpose | m5.24xlarge     | 96       | 384          | EBS only      | Yes        | 25 Gigabit       | Yes  |

At the bottom of the page are buttons for "Cancel," "Previous," "Review and Launch" (which is highlighted with a red box and circled with a red number 2), and "Next: Configure Instance Details".

# EC2 pricing example

$\pi$

| 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) <span>▼</span>     |                         |                    |                           |                           |                      |
| vCPU  | ECU                     | Memory (GiB)       | Instance Storage (GB)     | Linux/UNIX Usage          |                      |
| <b>General Purpose - Current Generation</b> |                         |                    |                           |                           |                      |
| 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

π

The screenshot shows the AWS CloudFormation console interface for creating a new stack. The top navigation bar includes the AWS logo, Services dropdown, Resource Groups dropdown, and user information (Sorayut Glomglome, Oregon, Support). Below the navigation is a progress bar with seven steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance (which is highlighted in orange), 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review.

**Step 3: Configure Instance Details**

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

**Instance Configuration:**

- Number of instances:** 1 (with a link to "Launch into Auto Scaling Group")
- Purchasing option:** Request Spot instances (unchecked)
- Network:** vpc-72e14516 (default) (with a link to "Create new VPC")
- Subnet:** No preference (default subnet in any Availability Zone) (with a link to "Create new subnet")
- Auto-assign Public IP:** Use subnet setting (Enable)
- Placement group:** Add instance to placement group (unchecked)
- Capacity Reservation:** Open (with a link to "Create new Capacity Reservation")
- IAM role:** None (with a link to "Create new IAM role")
- CPU options:** Specify CPU options (unchecked)
- Shutdown behavior:** Stop
- Stop - Hibernate behavior:** Enable hibernation as an additional stop behavior (unchecked)
- Enable termination protection:** Protect against accidental termination (unchecked)

**Buttons at the bottom:**

- Cancel
- Previous
- Review and Launch** (highlighted with a red circle containing the number 1)
- Next: Add Storage

**Footer:**

- Feedback
- English (US)
- © 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.
- Privacy Policy
- Terms of Use

# Add Storage : 20 GiB

π

The screenshot shows the AWS EC2 instance creation wizard at Step 4: Add Storage. The 'Size (GiB)' input field for the root volume is highlighted with a red circle and labeled '1'. The volume is set to 'General Purpose SSD (gp2)'. A note below states: 'Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. Learn more about free usage tier eligibility and usage restrictions.' At the bottom right, a red circle labeled '2' is positioned over the 'Review and Launch' button.

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

| Volume Type<br>i | Device<br>i | Snapshot<br>i          | Size (GiB)<br>i | Volume Type<br>i          | IOPS<br>i  | Throughput<br>(MB/s)<br>i | Delete on<br>Termination<br>i       | Encrypted<br>i |
|------------------|-------------|------------------------|-----------------|---------------------------|------------|---------------------------|-------------------------------------|----------------|
| Root             | /dev/sda1   | snap-0252bea5b37202c35 | 20              | General Purpose SSD (gp2) | 100 / 3000 | N/A                       | <input checked="" type="checkbox"/> | Not Encrypted  |

Add New Volume

1

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

Cancel Previous Review and Launch Next: Add Tags

Feedback English (US)

© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

# Name your server

π

The screenshot shows the AWS EC2 instance creation process at Step 5: Add Tags. The top navigation bar includes the AWS logo, Services dropdown, Resource Groups dropdown, and user information (Sorayut Glomglome, Oregon, Support). Below the navigation is a progress bar with steps 1 through 7. The main content area is titled "Step 5: Add Tags" with a sub-instruction about tag key-value pairs. A red circle labeled "1" points to a large red arrow pointing up from the "Add another tag" button. A red circle labeled "2" points to the "Key" input field, which contains "Name". A red circle labeled "3" points to the "Value" input field, which contains "BDA-Hadoop Server". A red circle labeled "4" points to the "Review and Launch" button at the bottom right.

1. Choose AMI   2. Choose Instance Type   3. Configure Instance   4. Add Storage   5. Add Tags   6. Configure Security Group   7. Review

**Step 5: Add Tags**

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key (127 characters maximum)   Value (255 characters maximum)

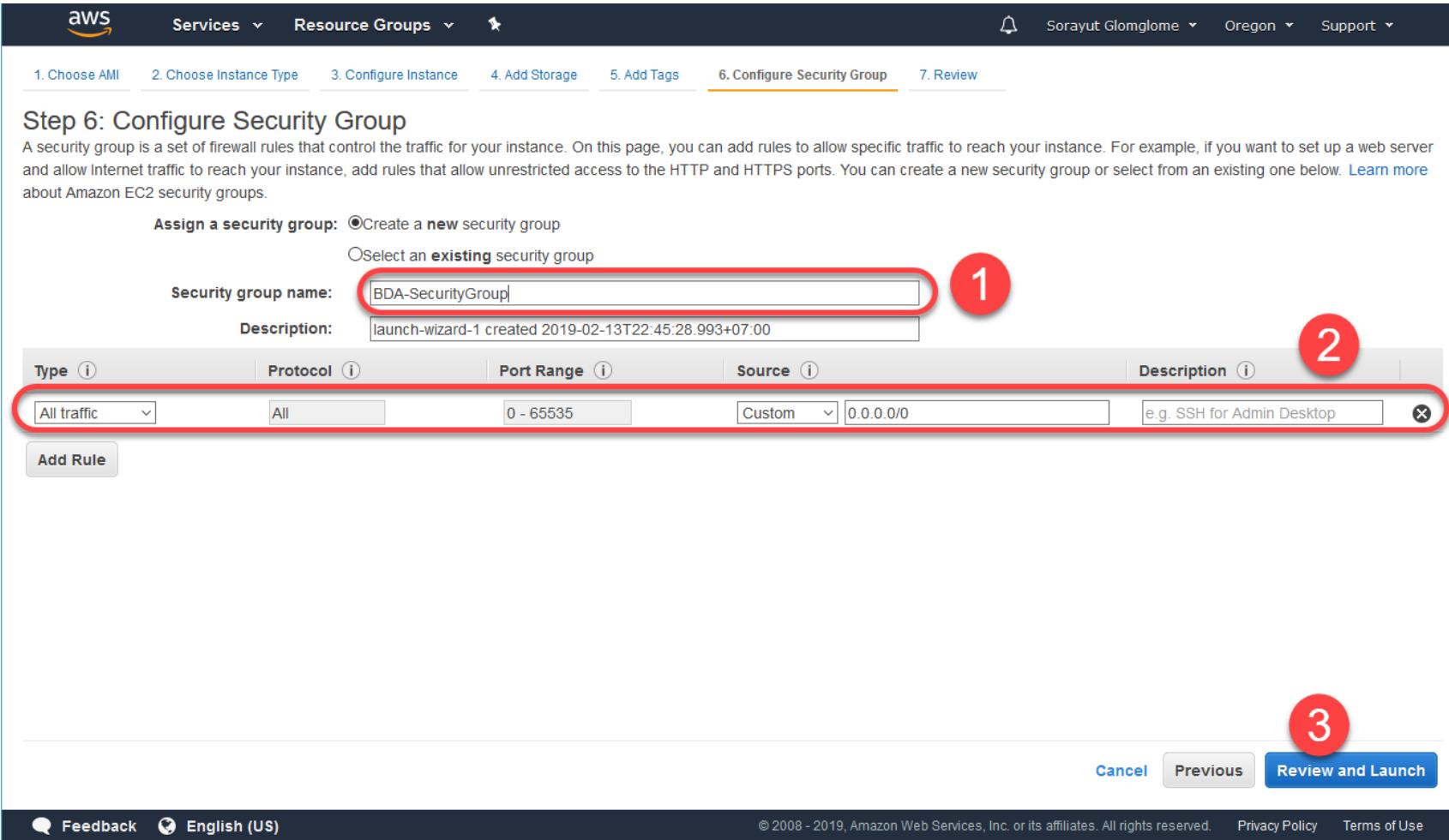
Add another tag (Up to 50 tags maximum)

Cancel Previous Review and Launch Next: Configure Security Group

Feedback English (US) © 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

# Create and name a new security group

## Allow all traffic because of the convenience



π

# Review and then launch the instance

The screenshot shows the AWS Step 7: Review Instance Launch page. At the top, there are tabs: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. The 7. Review tab is selected.

**Step 7: Review Instance Launch**

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

**AMI Details:** Ubuntu Server 16.04 LTS (HVM), SSD Volume Type - ami-076e276d85f524150  
Free tier eligible Root Device Type: ebs Virtualization type: hvm

**Instance Type:**

| Instance Type | ECUs | vCPUs | Memory (GiB) | Instance Storage (GB) | EBS-Optimized Available | Network Performance |
|---------------|------|-------|--------------|-----------------------|-------------------------|---------------------|
| m5.large      | 10   | 2     | 8            | EBS only              | Yes                     | Up to 10 Gigabit    |

**Security Groups:** Edit security groups

**Buttons:** Cancel, Previous, **Launch** (highlighted with a red circle and the number 1).



# Name a new key pair for authentication

## Download to a save location

**\*\*\*IMPORTANT\*\*\* Do not lose the key file**

The screenshot shows the AWS CloudFormation console during the 'Step 7: Review Instance Launch' process. The main page displays instance launch details, including AMI, instance type (m5.large), and security groups. A modal dialog titled 'Select an existing key pair or create a new key pair' is open. Inside the dialog, there's a note about key pairs, a section to 'Create a new key pair' with a 'Key pair name' input field containing 'BDA Key Pair' (circled with red number 1), a 'Download Key Pair' button (circled with red number 2), and a note about downloading the private key file (circled with red number 3). The 'Launch Instances' button is at the bottom right of the dialog.



# Launch Status

Click ① to check the instance

Click ② to read how to connect to Linux instance

The screenshot shows the AWS Launch Status page. At the top, there's a navigation bar with the AWS logo, Services, Resource Groups, a bell icon, Sorayut Glomglome, Oregon, and Support. The main section is titled "Launch Status". It displays a green box with a checkmark stating "Your instances are now launching" and a link to "View launch log". Below this, there's a blue box with an info icon and the text "Get notified of estimated charges". Under "How to connect to your instances", it says instances are launching and will be ready soon. It encourages users to click "View Instances" to monitor status and "Find out" how to connect. A dropdown menu lists helpful resources, with "How to connect to your Linux instance" highlighted by a red circle with the number 2. At the bottom, there are links for status checks, EBS volumes, and security groups, along with standard footer links for Feedback, English (US), Privacy Policy, and Terms of Use.

aws Services Resource Groups

Sorayut Glomglome Oregon Support

Launch Status

1

Your instances are now launching

The following instance launches have been initiated: i-0dd57077f57bc13c0 View launch log

2

Get notified of estimated charges

Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Click [View Instances](#) to monitor your instances' status. Once your instances are in the **running** state, you can **connect** to them from the Instances screen. [Find out](#) how to connect to your instances.

▼ Here are some helpful resources to get you started

- [AWS China Marketplace: Help](#)
- [How to connect to your Linux instance](#) 2
- [Learn about AWS Free Usage Tier](#)
- [Amazon EC2: User Guide](#)
- [Amazon EC2: Discussion Forum](#)

While your instances are launching you can also

[Create status check alarms](#) to be notified when these instances fail status checks. (Additional charges may apply)

[Create and attach additional EBS volumes](#) (Additional charges may apply)

[Manage security groups](#)

Feedback English (US) © 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

# Connect to Your Linux instance

- <https://goo.gl/wdSfSr>

| Your Computer     | Topic   |
|-------------------|---|
| Linux or Mac OS X | <a href="#">Connecting to Your Linux Instance Using SSH</a>   |
| Windows           | <a href="#">Connecting to Your Linux Instance from Windows Using PuTTY</a><br><a href="#">Connecting to Your Linux Instance from Windows Using Windows Subsystem for Linux</a><br><a href="#">Connecting to Your Linux Instance Using SSH</a> |
| All               | <a href="#">Connecting to Your Linux Instance Using MindTerm</a>  |



# Start the instance:

## Right click on the row of instance

### Select Instance State -> Start

The screenshot shows the AWS EC2 Dashboard. On the left, there's a sidebar with navigation links like EC2 Dashboard, Events, Tags, Reports, Limits, Instances (selected), Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, AMIs, Bundle Tasks, Volumes, Snapshots, Lifecycle Manager, and Security Groups. The main area displays a table of instances. One instance, named "BDA-Hadoop...", has a status of "stopped". A context menu is open over this instance, with a red circle labeled "1" around the "Actions" button. The menu itself is a red circle labeled "2" and contains options: Connect, Get Windows Password, Create Template From Instance, Launch More Like This, Instance State (selected), Instance Settings, Image, Networking, CloudWatch Monitoring, Stop, Stop - Hibernate, Reboot, and Terminate.

| Name          | Instance ID         | Instance Type | Availability Zone | Instance State | Status Checks | Alarm Status | Public DNS (IPv4) |
|---------------|---------------------|---------------|-------------------|----------------|---------------|--------------|-------------------|
| BDA-Hadoop... | i-0dd57077f57bc13c0 | m5.large      | us-west-2b        | stopped        |               |              |                   |

Instance: i-0dd57077f57bc13c0 (BDA-Hadoop Server) Private IP: 172.31.16.229

| Description                      | Status Checks   | Monitoring   | Tags |
|----------------------------------|---|--|------|
| Instance ID: i-0dd57077f57bc13c0 | Instance state: stopped                                 | Public DNS (IPv4): -                                     |      |
| Instance type: m5.large          | Elastic IPs   | IPv4 Public IP: -  |      |
| Volumes                          |   | IPv6 IPs: -  |      |
| Snapshots                        |   | Private DNS: ip-172-31-16-229.us-west-2.compute.internal |      |
| Lifecycle Manager                |   | Private IPs: 172.31.16.229                               |      |
| Availability zone: us-west-2b    | Security groups: BDA-SecurityGroup. view inbound rules. | Secondary private IPs                                    |      |

Feedback English (US) © 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

# Right click and select connect to display

## How to connect to instance

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, AMIs, Bundle Tasks, Volumes, Snapshots, Lifecycle Manager, Security Groups, and Network & Security. The 'Instances' section is currently selected.

In the main area, there's a search bar with 'search : i-0dd57077f57bc13c0' and a 'Connect' button. Below the search bar is a table with columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, and Public DNS (IPv4). A single row is selected for an instance named 'BDA-Hadoop...' with the ID 'i-0dd57077f57bc13c0'.

A context menu is open over the selected instance row, with the 'Connect' option highlighted. Other options in the menu include Get Windows Password, Create Template From Instance, Launch More Like This, Instance State, Instance Settings, Image, Networking, and CloudWatch Monitoring.

Below the table, there's a detailed view of the instance 'i-0dd57077f57bc13c0'. It shows the following details:

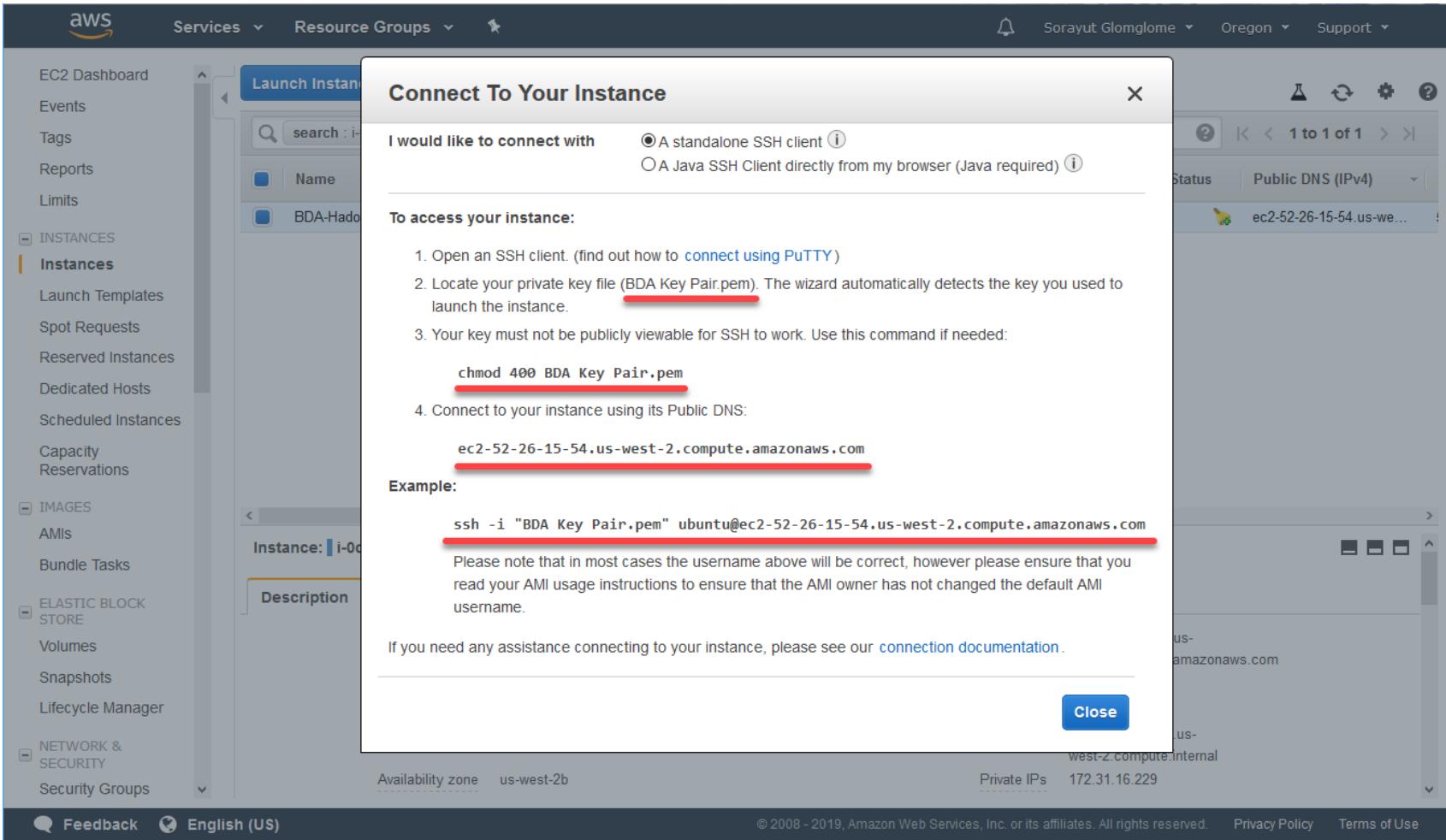
| Description       | Value   |
|-------------------|---|
| Instance ID       | i-0dd57077f57bc13c0                             |
| Instance state    | running   |
| Instance type     | m5.large  |
| Elastic IPs       | -   |
| Availability zone | us-west-2b                                      |
| Public DNS (IPv4) | ec2-52-26-15-54.us-west-2.compute.amazonaws.com |
| IPv4 Public IP    | 52.26.15.54                                     |
| IPv6 IPs          | -   |
| Private DNS       | ip-172-31-16-229.us-west-2.compute.internal     |
| Private IPs       | 172.31.16.229                                   |

Annotations on the page highlight specific information:

- A red box surrounds the 'Public DNS (IPv4)' and 'IPv4 Public IP' fields, with the text 'Change after time passing or restarting the instance' above it.
- A red box surrounds the 'Private DNS' and 'Private IPs' fields, with the text 'Never change' below it.
- A red circle highlights the 'Refresh' button in the top right corner of the main EC2 Instances interface.

# Connect to instance using SSH

π





# Install PuTTY and PuTTYGen

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

The screenshot shows the AWS EC2 Dashboard with the 'Instances' section selected. A modal window titled 'Connect To Your Instance' is open. The window contains instructions for connecting to the instance using a standalone SSH client or a Java SSH Client. It provides steps for locating the private key file ('BDA Key Pair.pem'), changing its permissions ('chmod 400 BDA Key Pair.pem'), and connecting using the Public DNS ('ec2-52-26-15-54.us-west-2.compute.amazonaws.com'). An example command is shown: 'ssh -i "BDA Key Pair.pem" ubuntu@ec2-52-26-15-54.us-west-2.compute.amazonaws.com'. A note states that the username 'ubuntu' is typically correct but to check AMI usage instructions. A 'Close' button is at the bottom right of the modal.

I would like to connect with  A standalone SSH client [\(i\)](#)  A Java SSH Client directly from my browser (Java required) [\(i\)](#)

To access your instance:

1. Open an SSH client. (find out how to [connect using PuTTY](#))
2. Locate your private key file ([BDA Key Pair.pem](#)). The wizard automatically detects the key you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use [this command](#) if needed:  
`chmod 400 BDA Key Pair.pem`
4. Connect to your instance using its Public DNS:  
`ec2-52-26-15-54.us-west-2.compute.amazonaws.com`

Example:

```
ssh -i "BDA Key Pair.pem" ubuntu@ec2-52-26-15-54.us-west-2.compute.amazonaws.com
```

Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.

If you need any assistance connecting to your instance, please see our [connection documentation](#).

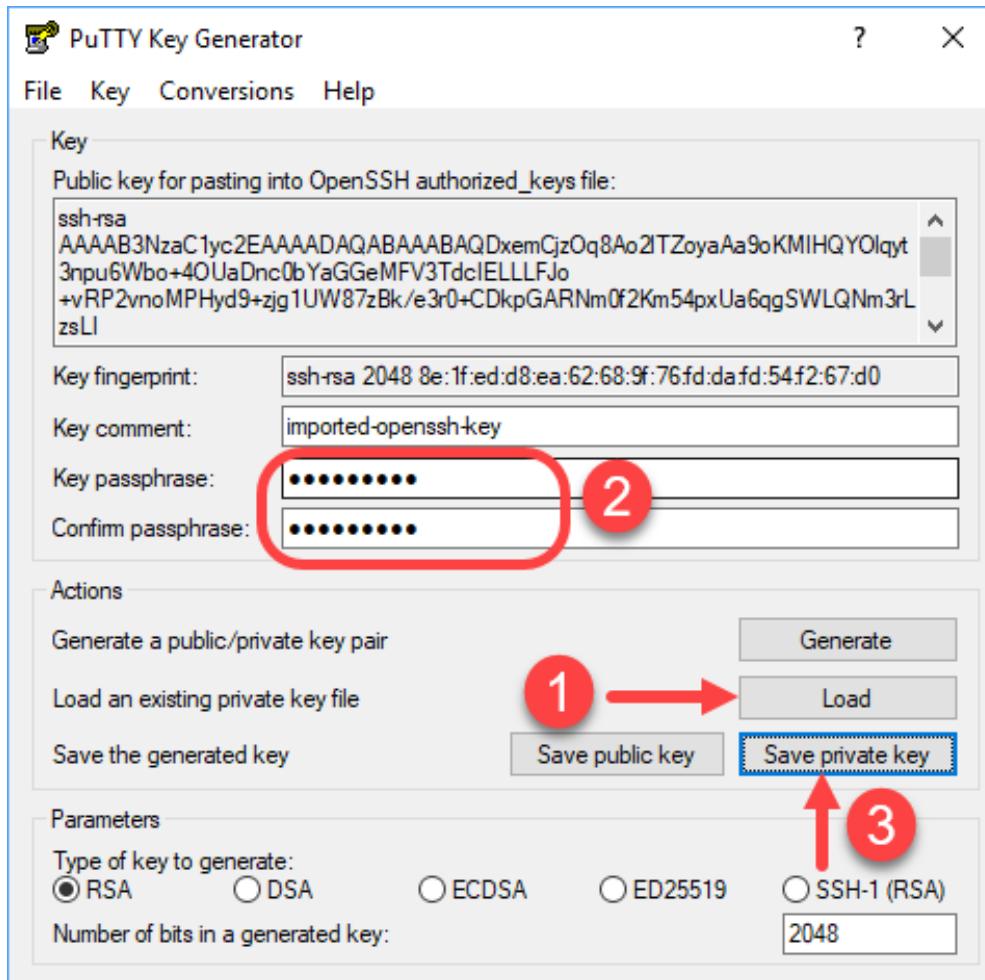
Close

Availability zone us-west-2b Private IPs 172.31.16.229

Feedback English (US) © 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

# Generate .ppk file from .pem file using PuTTYGen

π



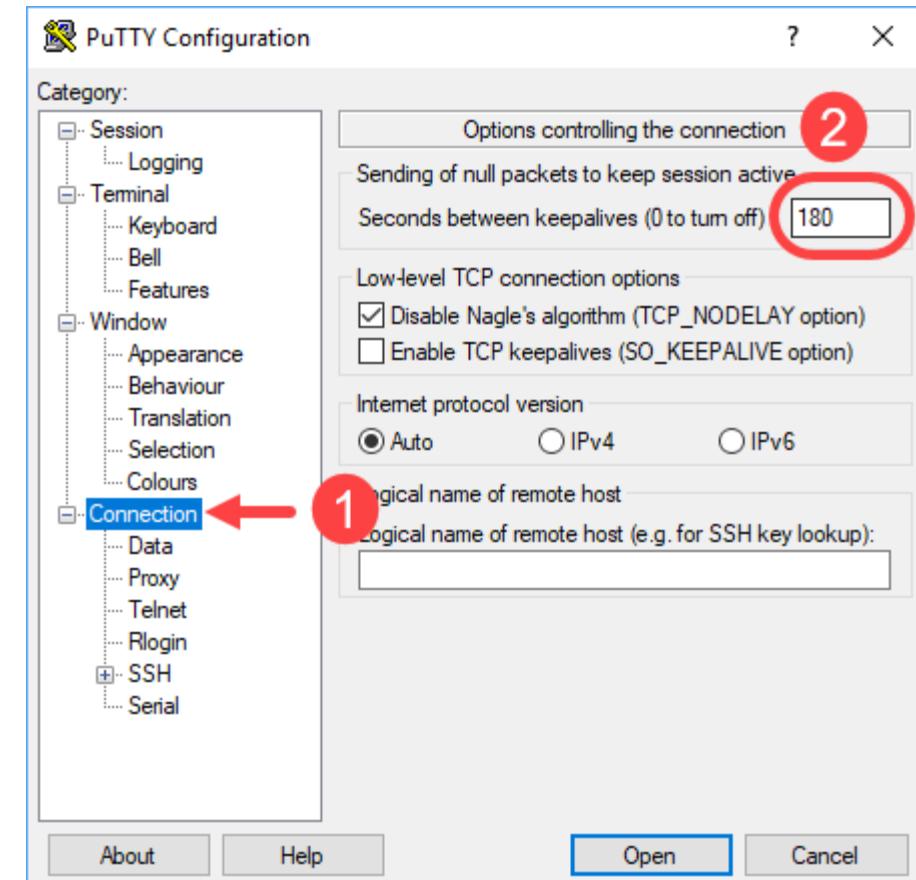
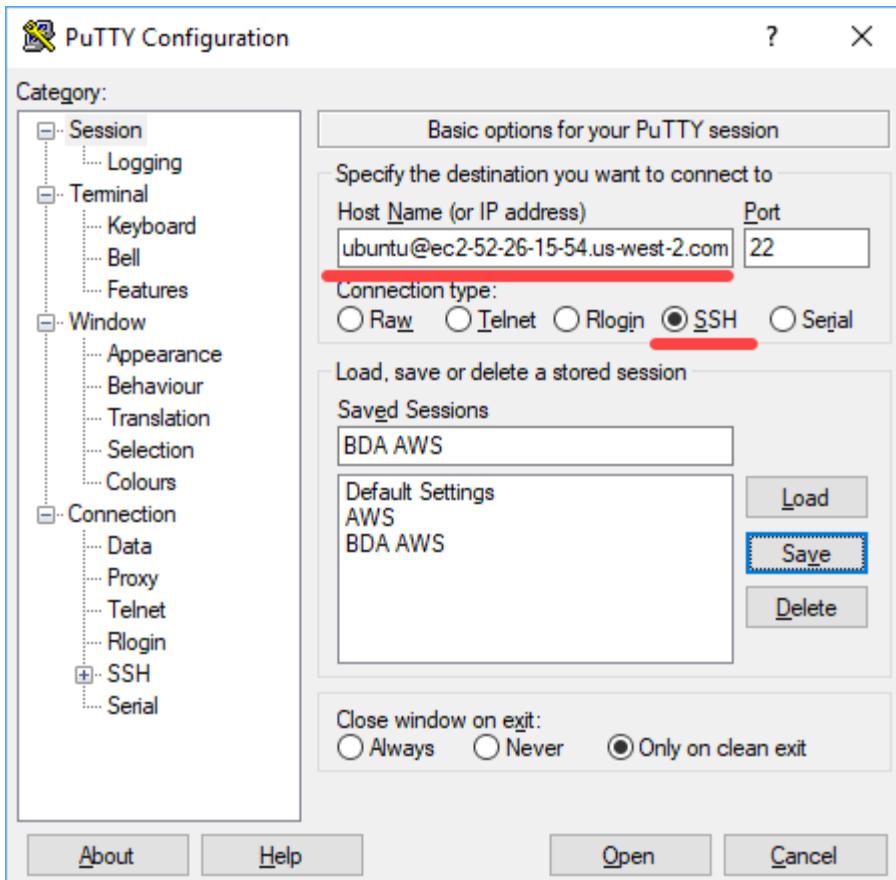
| Name           | Date modified   | Type                 | Size |
|----------------|-----------------|----------------------|------|
| BDAKeyPair.pem | 13-Feb-19 22:53 | PEM File             | 2 KB |
| BDAKeyPair.ppk | 14-Feb-19 0:48  | PuTTY Private Key... | 2 KB |



# Configure PuTTY

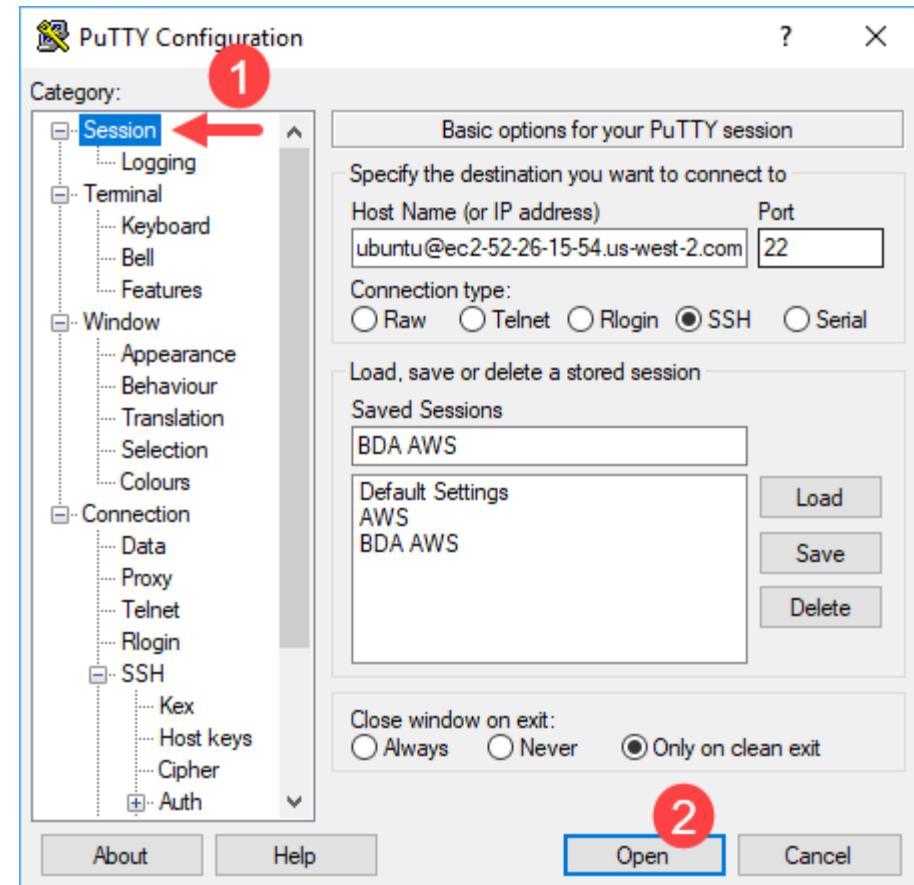
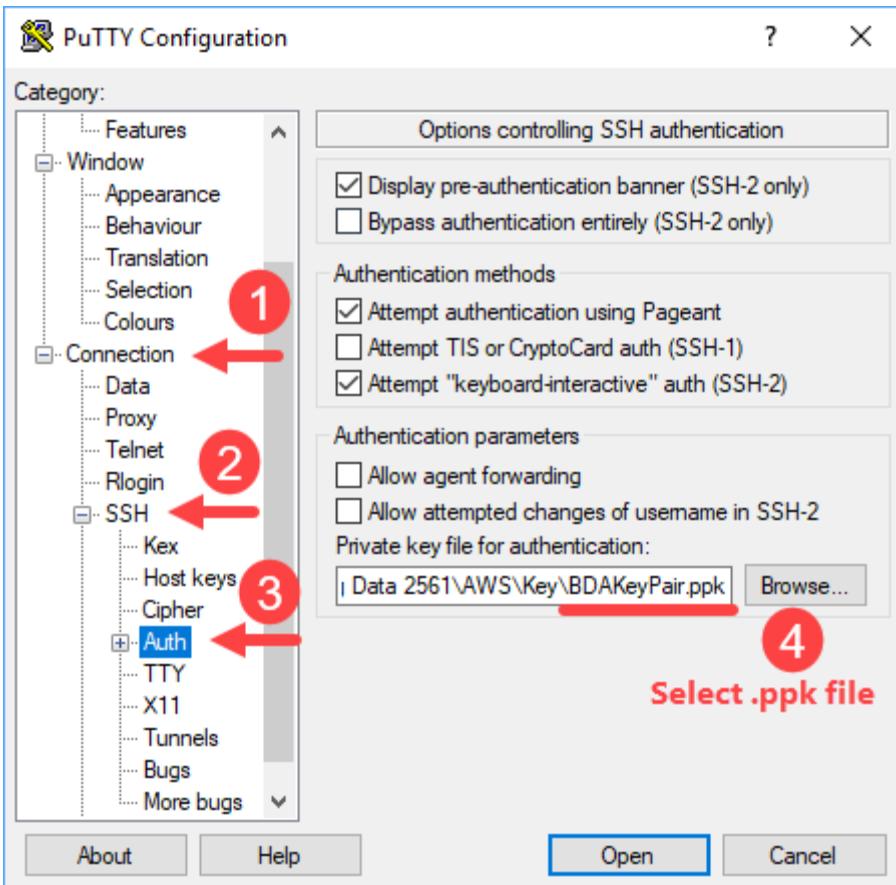
Host name : **ubuntu@[PublicDNS]**

Generate keepalive signal



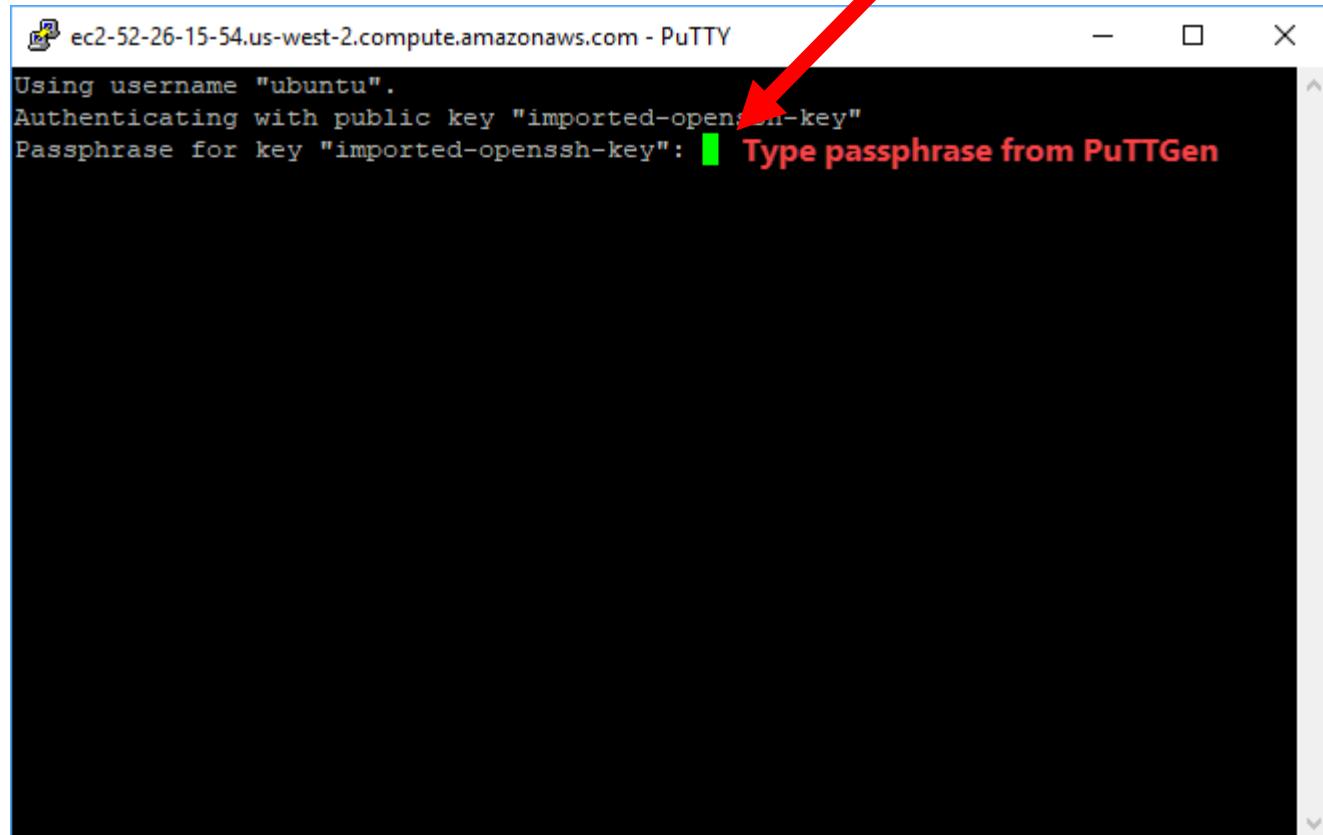
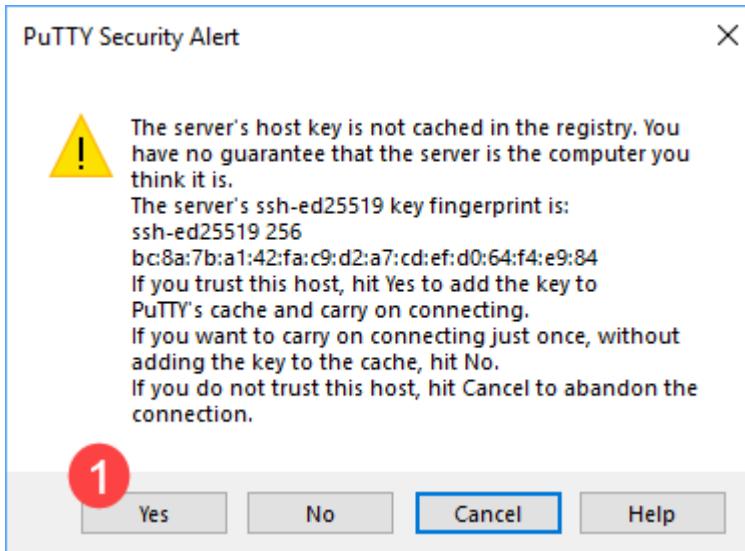
# Select .ppk file for authentication

## Use private key instead of typing password



# Connect to AWS instance

π



ec2-52-26-15-54.us-west-2.compute.amazonaws.com - PuTTY

Using username "ubuntu".  
Authenticating with public key "imported-openssh-key"  
Passphrase for key "imported-openssh-key": **Type passphrase from PuTTGen**

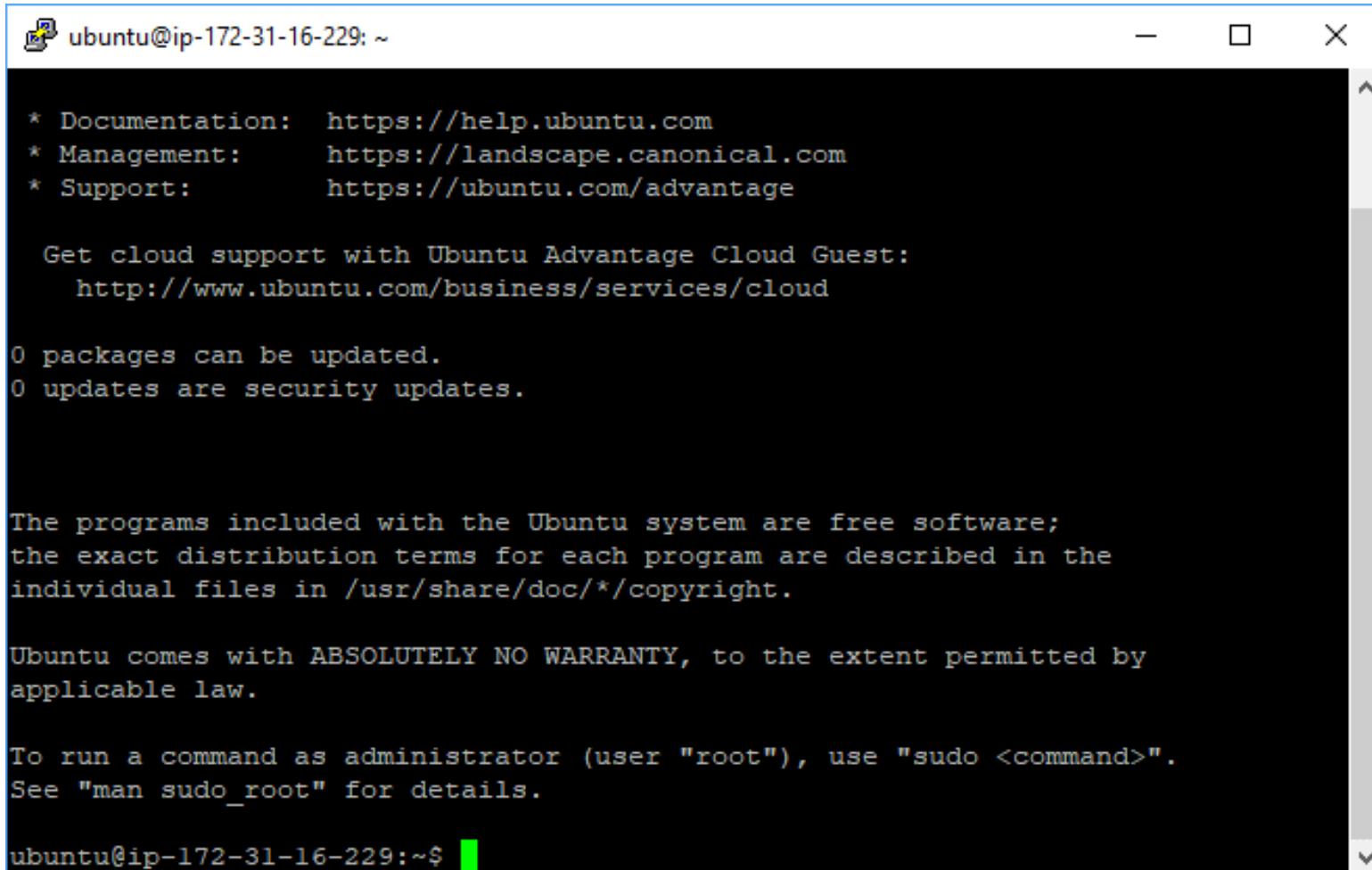
PuTTY Key Generator window details:

- Key fingerprint: ssh-rsa 2048 8e:1fed:d8:ea:62:68:9f:76fd:da:fd:54:f2:67:d0
- Key comment: imported-openssh-key
- Key passphrase: **\*\*\*\*\*** (circled with red 2)
- Confirm passphrase: **\*\*\*\*\*** (circled with red 2)
- Actions: Generate (button), Load (button), Save public key (button), **Save private key** (button circled with red 3)
- Parameters: Type of key to generate: RSA (radio button selected), DSA, ECDSA, ED25519, SSH-1 (RSA), Number of bits in a generated key: 2048

A large red arrow points from the "Save private key" button in the PuTTY Key Generator to the "Passphrase" field in the PuTTY session window.

# Command Prompt shows a success connection

π



A screenshot of a terminal window titled "ubuntu@ip-172-31-16-229: ~". The window contains the following text:

```
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage

Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

0 packages can be updated.
0 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-16-229:~$
```

## 2. Installing Apache Hadoop

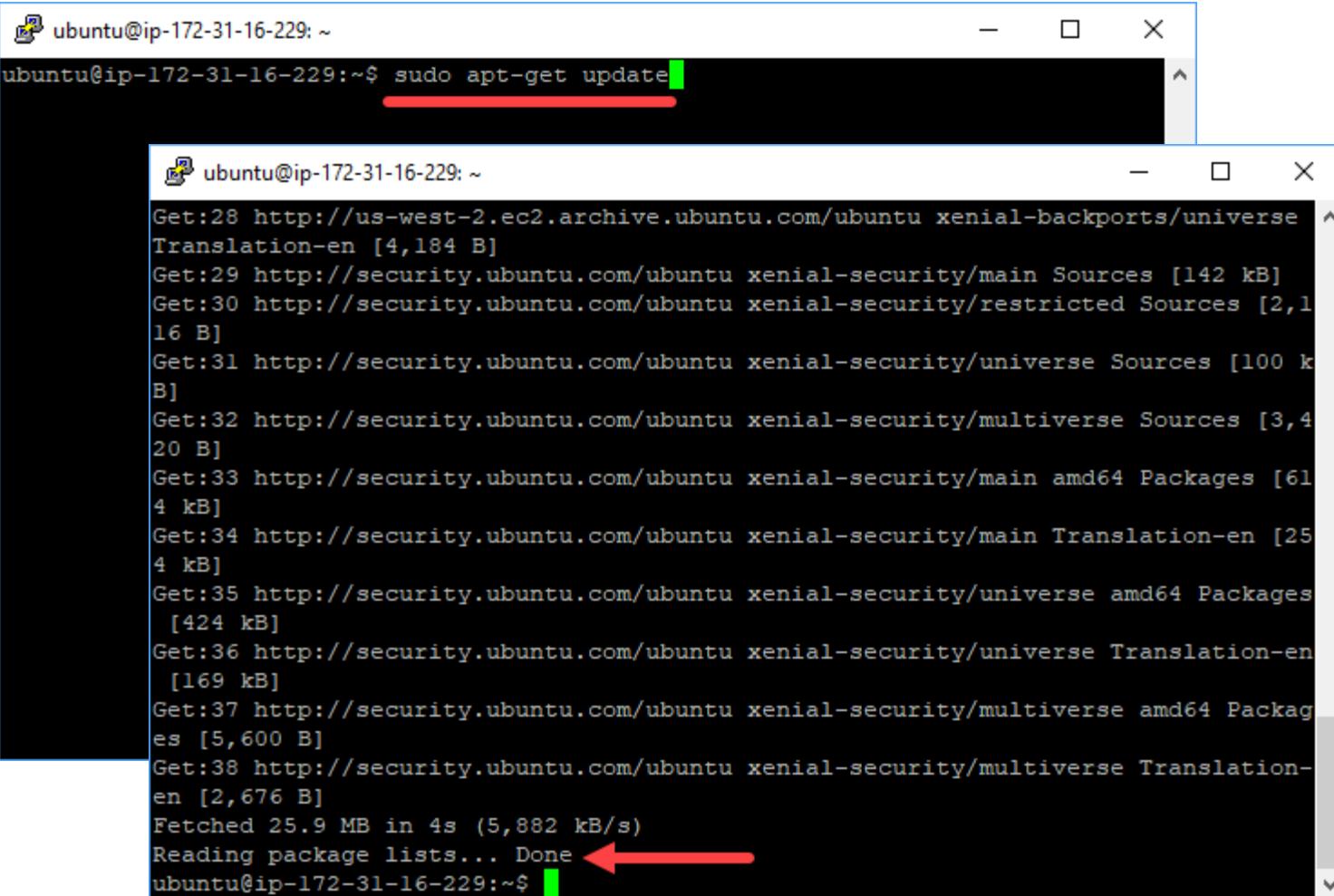
# Installing Hadoop and Ecosystem

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

$\pi$

# \$sudo apt-get update



The image shows two terminal windows side-by-side. The top window displays the command `sudo apt-get update`. The bottom window shows the detailed output of the command, which includes multiple `Get:` entries for different Ubuntu repositories, followed by a summary line `Fetched 25.9 MB in 4s (5,882 kB/s)`, and finally the message `Reading package lists... Done`. A red arrow points to the word `Done` in the bottom window.

```
ubuntu@ip-172-31-16-229: ~
ubuntu@ip-172-31-16-229: ~$ sudo apt-get update
Get:28 http://us-west-2.ec2.archive.ubuntu.com/ubuntu xenial-backports/universe Translation-en [4,184 B]
Get:29 http://security.ubuntu.com/ubuntu xenial-security/main Sources [142 kB]
Get:30 http://security.ubuntu.com/ubuntu xenial-security/restricted Sources [2,116 B]
Get:31 http://security.ubuntu.com/ubuntu xenial-security/universe Sources [100 kB]
Get:32 http://security.ubuntu.com/ubuntu xenial-security/multiverse Sources [3,420 B]
Get:33 http://security.ubuntu.com/ubuntu xenial-security/main amd64 Packages [614 kB]
Get:34 http://security.ubuntu.com/ubuntu xenial-security/main Translation-en [254 kB]
Get:35 http://security.ubuntu.com/ubuntu xenial-security/universe amd64 Packages [424 kB]
Get:36 http://security.ubuntu.com/ubuntu xenial-security/universe Translation-en [169 kB]
Get:37 http://security.ubuntu.com/ubuntu xenial-security/multiverse amd64 Packages [5,600 B]
Get:38 http://security.ubuntu.com/ubuntu xenial-security/multiverse Translation-en [2,676 B]
Fetched 25.9 MB in 4s (5,882 kB/s)
Reading package lists... Done
```

## 2.2 Configure SSH



# Install SSH:

```
$sudo apt-get install openssh-server
```

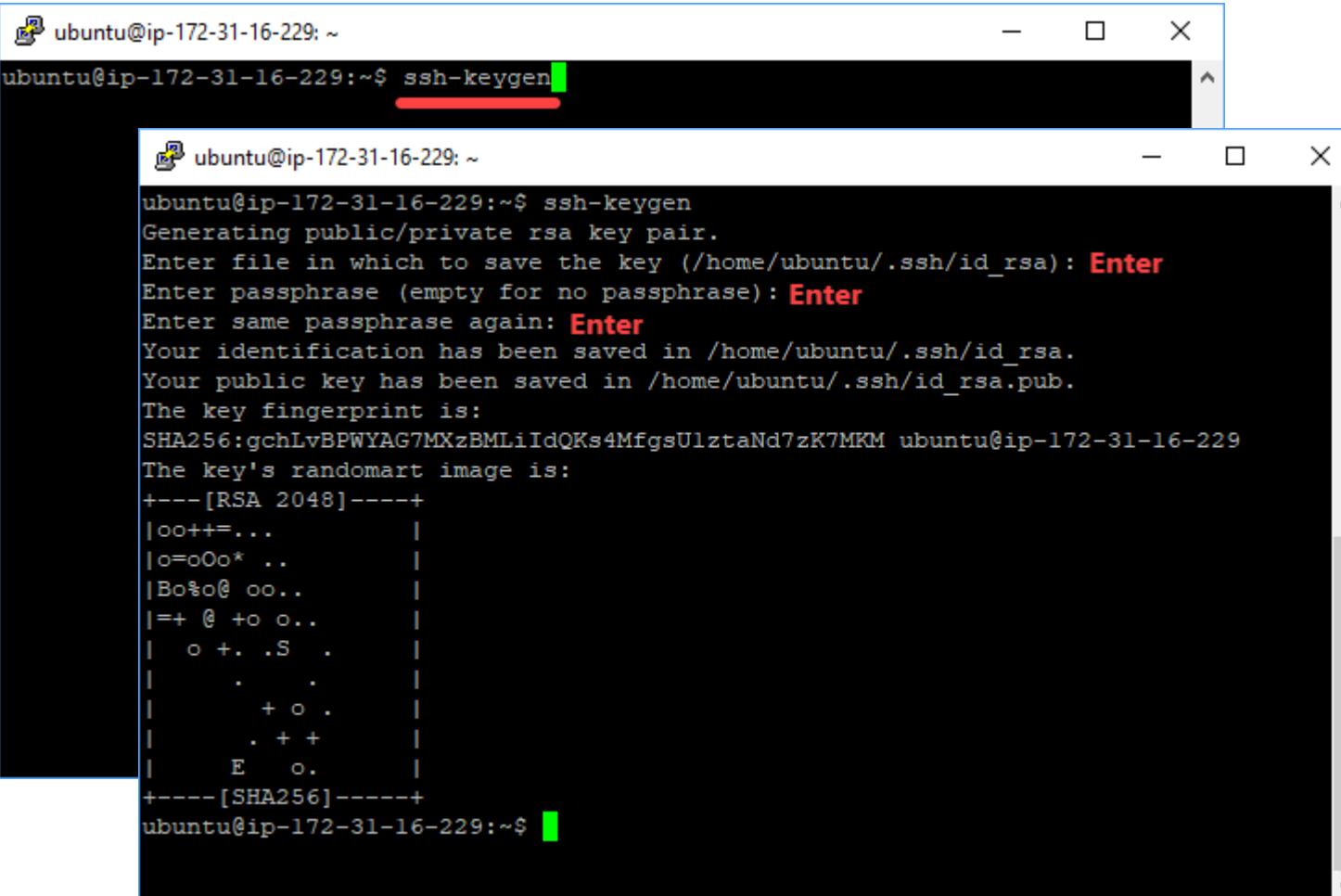
The screenshot shows a terminal window with two tabs. The top tab has the command `sudo apt-get install openssh-server`. The bottom tab displays the execution of this command, showing the package list, dependency tree, state information, and upgrade details. It ends with a prompt asking if the user wants to continue with 'y'.

```
ubuntu@ip-172-31-16-229:~$ sudo apt-get install openssh-server
Reading package lists...
Building dependency tree...
Reading state information...
The following additional packages will be installed:
  openssh-client openssh-sftp-server
Suggested packages:
  ssh-askpass libpam-ssh keychain monkeysphere rssh molly-guard
The following packages will be upgraded:
  openssh-client openssh-server openssh-sftp-server
3 upgraded, 0 newly installed, 0 to remove and 59 not upgraded.
Need to get 964 kB of archives.
After this operation, 8,192 B of additional disk space will be used.
Do you want to continue? [Y/n] y
```

π

# Create Private Key with empty passphrase:

## \$ssh-keygen



```
ubuntu@ip-172-31-16-229:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_rsa): Enter
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
Your identification has been saved in /home/ubuntu/.ssh/id_rsa.
Your public key has been saved in /home/ubuntu/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:gchLvBPWYAG7MXzBMLiIdQKs4MfgsUlztaNd7zK7MKM ubuntu@ip-172-31-16-229
The key's randomart image is:
+---[RSA 2048]---+
|oo++=...
|o=oOo* ...
|Bo%o@ oo...
|=+ @ +o o...
| o +. .S .
| .
| + o .
| .
| . + +
| E o.
+---[SHA256]---+
ubuntu@ip-172-31-16-229:~$
```

# Append and test new key:

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

The image displays two terminal windows side-by-side. Both windows have a title bar with the text "ubuntu@ip-172-31-16-229: ~".

**Top Terminal Window:**

```
ubuntu@ip-172-31-16-229:~$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
ubuntu@ip-172-31-16-229:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:kYNMXQ9jbcRO6wPKy9ONsOUM/02uB4NfIv9o7EHjQw.
Are you sure you want to continue connecting (yes/no)? yes
```

**Bottom Terminal Window:**

```
ubuntu@ip-172-31-16-229:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:kYNMXQ9jbcRO6wPKy9ONsOUM/02uB4NfIv9o7EHjQw.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 16.04.5 LTS (GNU/Linux 4.4.0-1072-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 Get cloud support with Ubuntu Advantage Cloud Guest:
   http://www.ubuntu.com/business/services/cloud

62 packages can be updated.
43 updates are security updates.

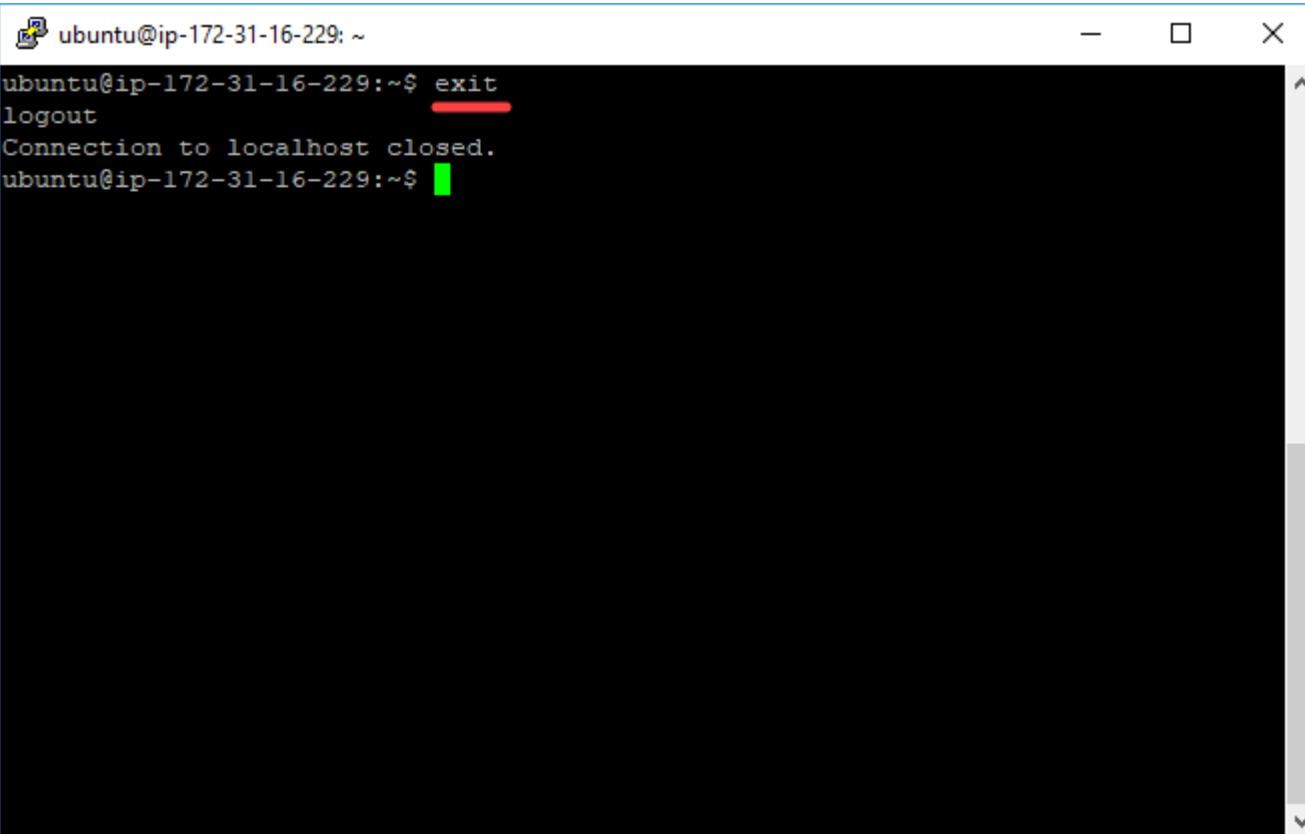
New release '18.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Feb 13 19:45:53 2019 from 161.246.145.13
ubuntu@ip-172-31-16-229:~$
```

$\pi$

# Logout:

## \$exit

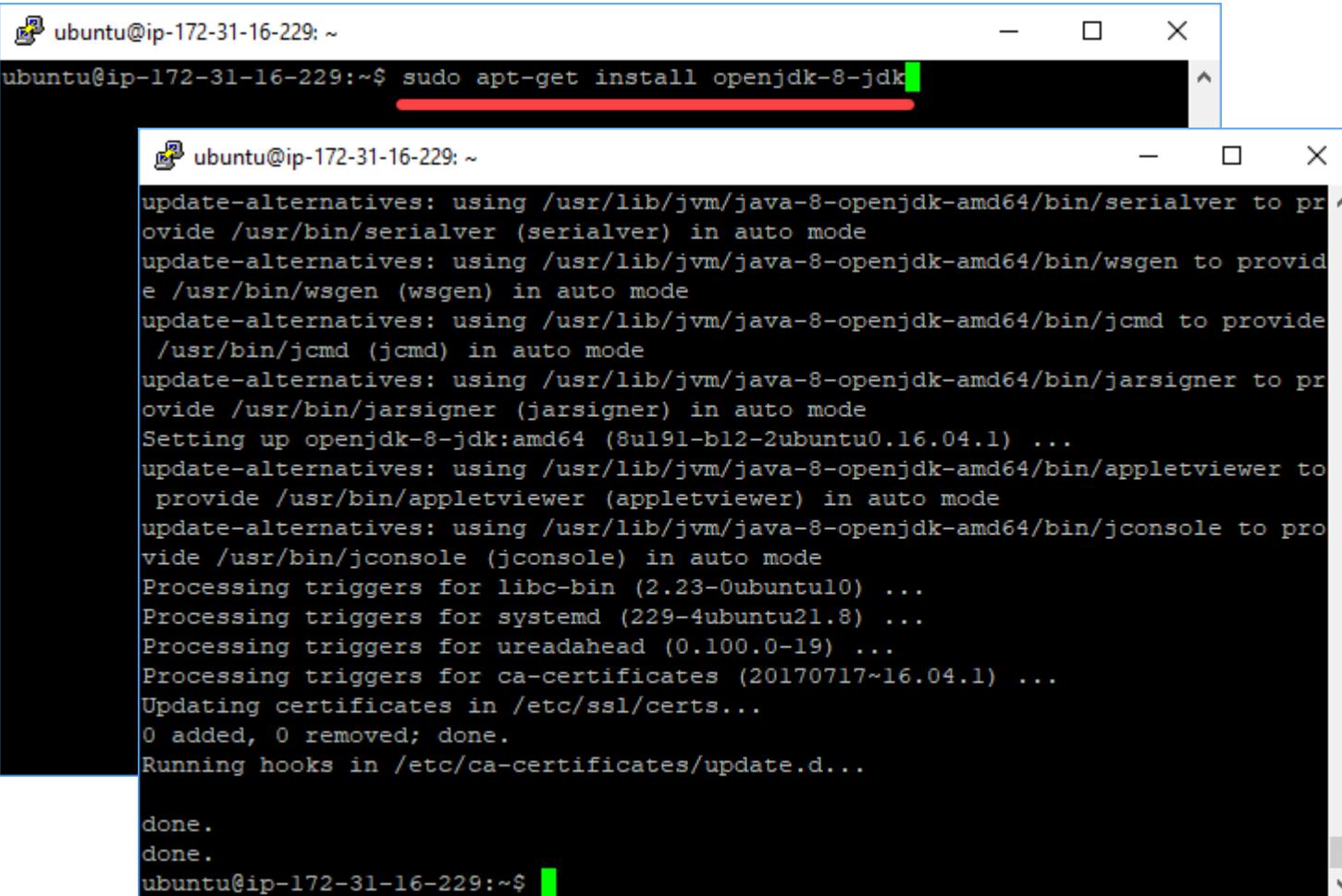


A screenshot of a terminal window titled "ubuntu@ip-172-31-16-229: ~". The window contains the following text:  
ubuntu@ip-172-31-16-229:~\$ exit  
logout  
Connection to localhost closed.  
ubuntu@ip-172-31-16-229:~\$

## 2.3 Install Java

# Install Java:

```
$sudo apt-get install openjdk-8-jdk
```



The screenshot shows a terminal window with two stacked panes. The top pane displays the command being run: `ubuntu@ip-172-31-16-229:~$ sudo apt-get install openjdk-8-jdk`. The bottom pane shows the detailed output of the command, which includes several log messages from the `update-alternatives` package, indicating it is setting up alternatives for Java-related tools like `serialver`, `wsgen`, `jcmd`, and `jarsigner`. It also shows the configuration of the `openjdk-8-jdk:amd64` package, the update of certificates, and the execution of hooks in `/etc/ca-certificates/update.d...`. The process concludes with two "done." messages at the bottom.

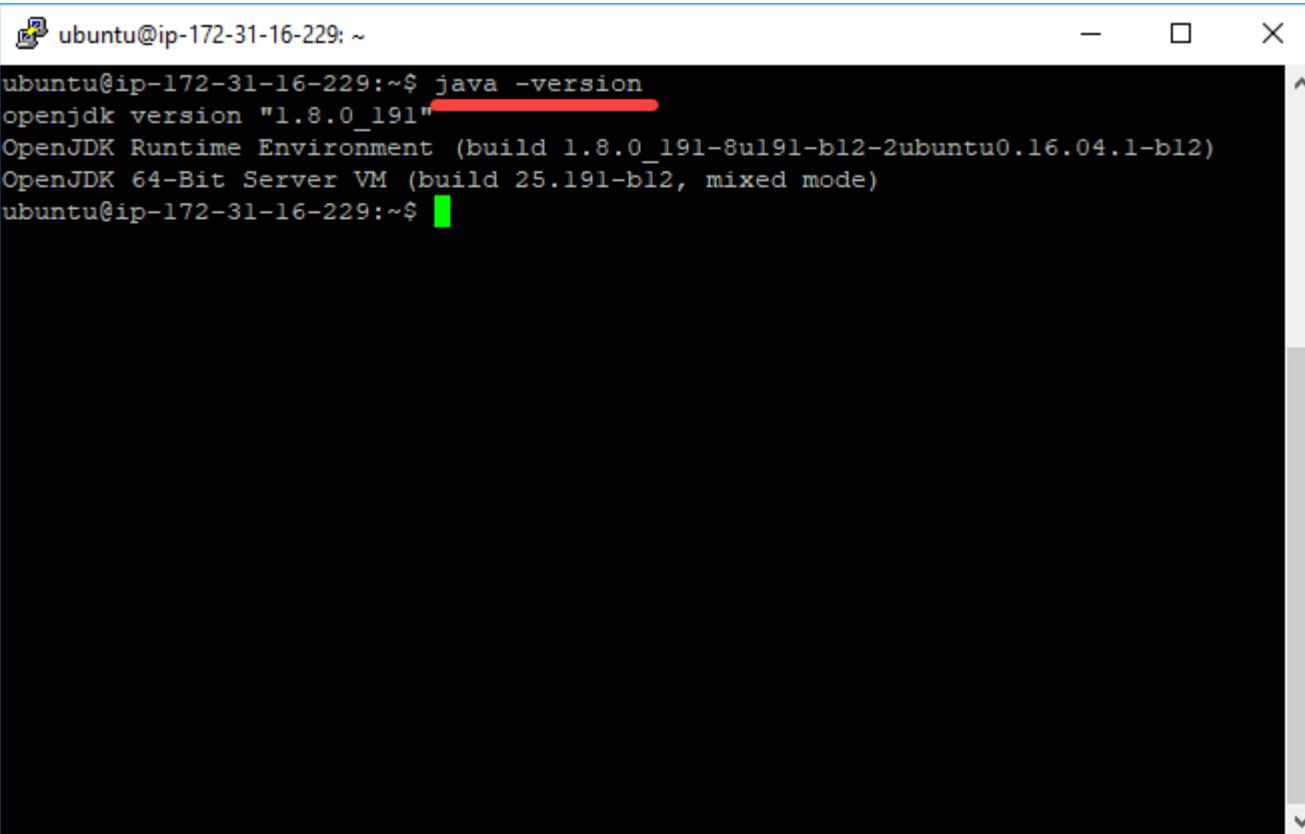
```
ubuntu@ip-172-31-16-229:~$ sudo apt-get install openjdk-8-jdk
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/bin/serialver to provide /usr/bin/serialver (serialver) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/bin/wsgen to provide /usr/bin/wsgen (wsgen) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/bin/jcmd to provide /usr/bin/jcmd (jcmd) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/bin/jarsigner to provide /usr/bin/jarsigner (jarsigner) in auto mode
Setting up openjdk-8-jdk:amd64 (8u191-b12-2ubuntu0.16.04.1) ...
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/bin/appletviewer to provide /usr/bin/appletviewer (appletviewer) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/bin/jconsole to provide /usr/bin/jconsole (jconsole) in auto mode
Processing triggers for libc-bin (2.23-0ubuntu10) ...
Processing triggers for systemd (229-4ubuntu21.8) ...
Processing triggers for ureadahead (0.100.0-19) ...
Processing triggers for ca-certificates (20170717~16.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...

done.
done.
ubuntu@ip-172-31-16-229:~$
```

π

# Check Java version:

**\$java -version**



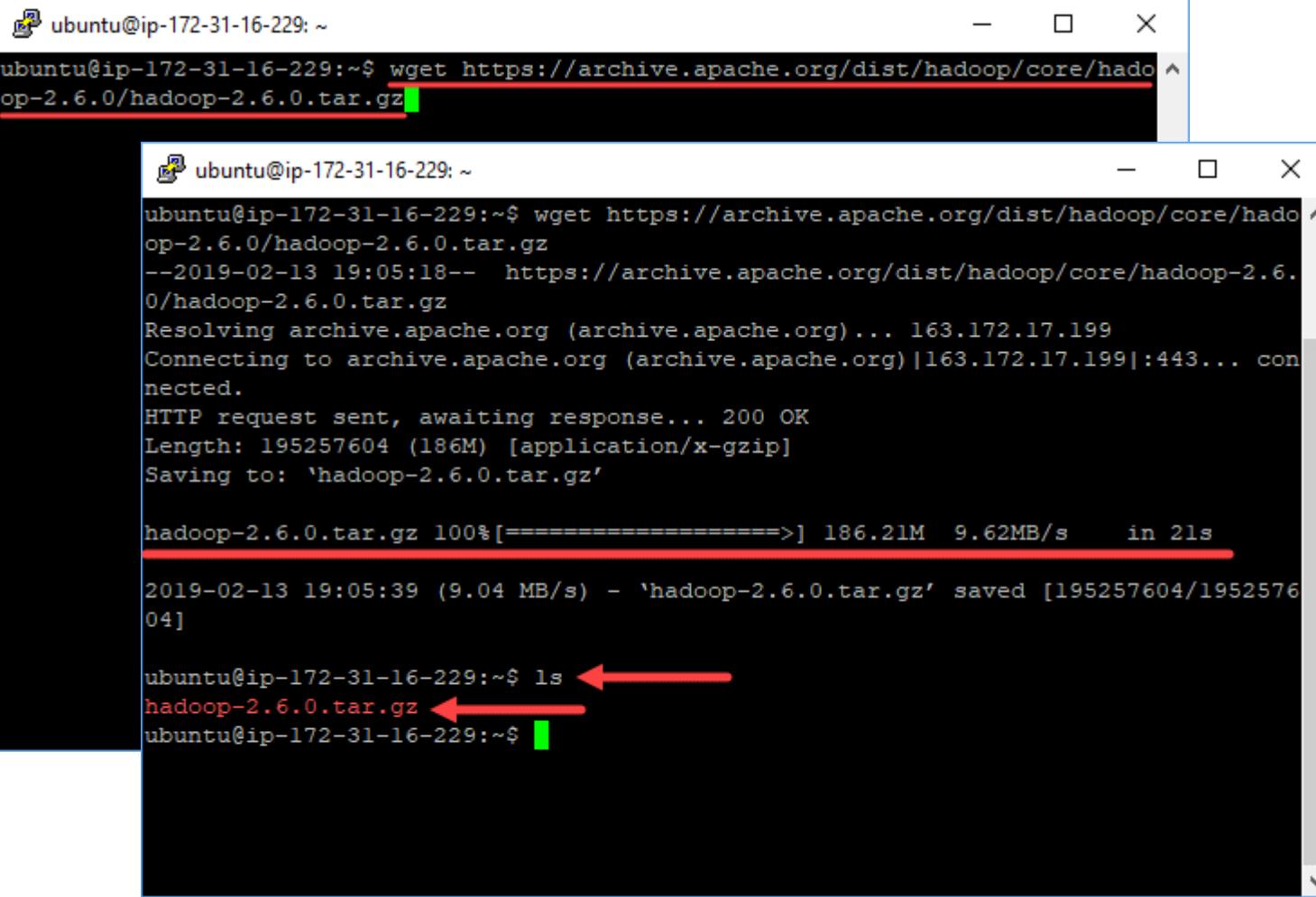
A screenshot of a terminal window titled "ubuntu@ip-172-31-16-229: ~". The window contains the following text output:

```
ubuntu@ip-172-31-16-229:~$ java -version
openjdk version "1.8.0_191"
OpenJDK Runtime Environment (build 1.8.0_191-8u191-b12-2ubuntu0.16.04.1-b12)
OpenJDK 64-Bit Server VM (build 25.191-b12, mixed mode)
ubuntu@ip-172-31-16-229:~$
```

## 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
```



The screenshot shows a terminal window with two tabs. The top tab displays the command being run: `ubuntu@ip-172-31-16-229:~$ wget https://archive.apache.org/dist/hadoop/core/hadoop-2.6.0/hadoop-2.6.0.tar.gz`. The bottom tab shows the progress of the download, including the connection details, file size, and download speed. After the download is complete, the user runs the command `ls` to list the contents of the current directory, which shows the downloaded file `hadoop-2.6.0.tar.gz`.

```
ubuntu@ip-172-31-16-229:~$ wget https://archive.apache.org/dist/hadoop/core/hadoop-2.6.0/hadoop-2.6.0.tar.gz  
--2019-02-13 19:05:18-- https://archive.apache.org/dist/hadoop/core/hadoop-2.6.0/hadoop-2.6.0.tar.gz  
Resolving archive.apache.org (archive.apache.org)... 163.172.17.199  
Connecting to archive.apache.org (archive.apache.org)|163.172.17.199|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 195257604 (186M) [application/x-gzip]  
Saving to: 'hadoop-2.6.0.tar.gz'  
  
hadoop-2.6.0.tar.gz 100%[=====] 186.21M 9.62MB/s in 21s  
  
2019-02-13 19:05:39 (9.04 MB/s) - 'hadoop-2.6.0.tar.gz' saved [195257604/195257604]  
  
ubuntu@ip-172-31-16-229:~$ ls  
hadoop-2.6.0.tar.gz  
ubuntu@ip-172-31-16-229:~$
```

# Extract Hadoop package:

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

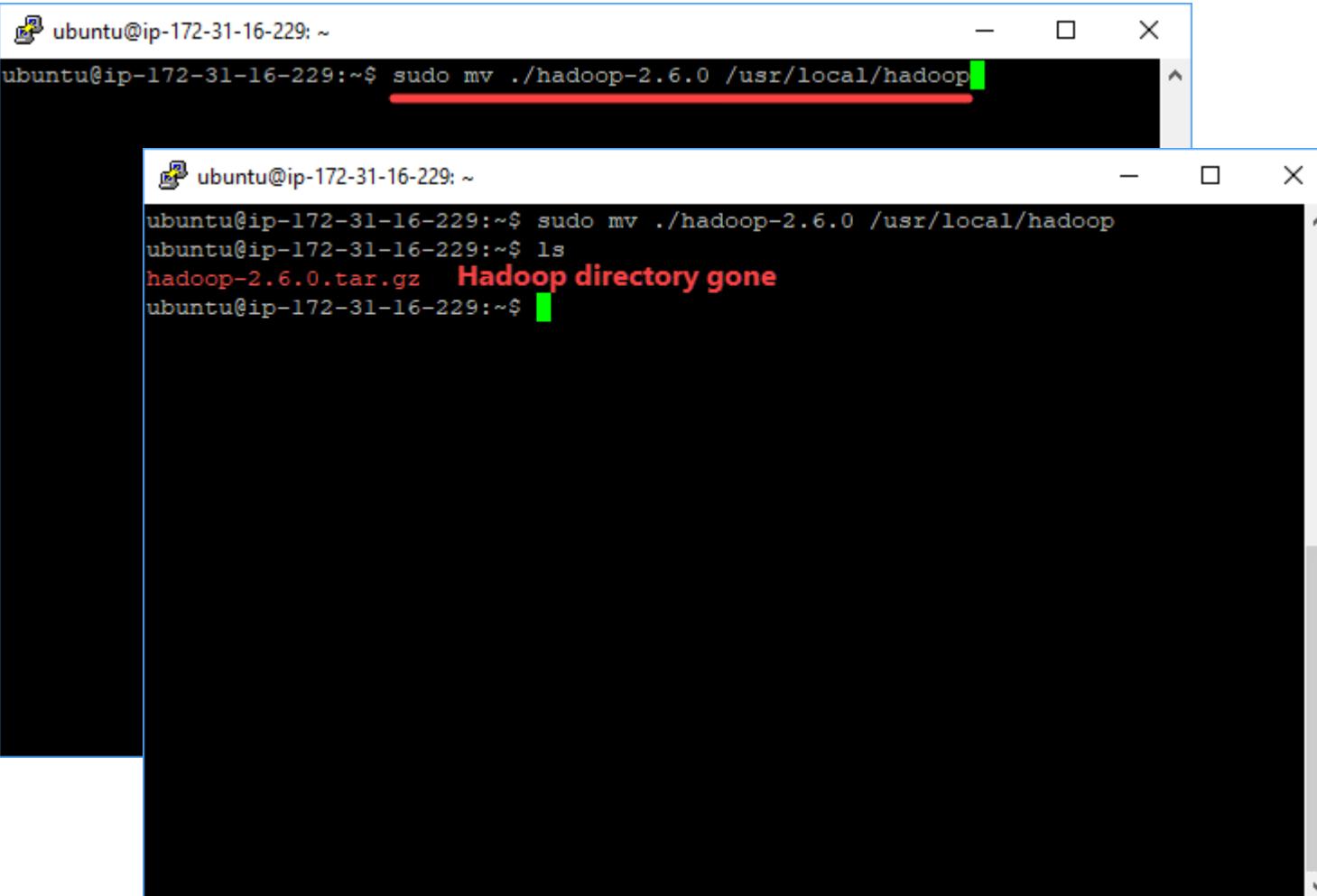
The screenshot shows a terminal window with two panes. The top pane displays the command `tar -xvf hadoop-2.6.0.tar.gz`. The bottom pane shows the directory structure after extraction, listing various Hadoop binary and include files. A red arrow points from the text "Directory after extracting" to the extracted directory entry in the ls output.

```
ubuntu@ip-172-31-16-229:~$ tar -xvf hadoop-2.6.0.tar.gz
ubuntu@ip-172-31-16-229:~$ ls
hadoop-2.6.0  hadoop-2.6.0.tar.gz
ubuntu@ip-172-31-16-229:~$ ls -l
total 190692
drwxr-xr-x 9 ubuntu ubuntu      4096 Nov 13  2014 hadoop-2.6.0 ←
-rw-rw-r-- 1 ubuntu ubuntu 195257604 Nov 30 2014 hadoop-2.6.0.tar.gz
ubuntu@ip-172-31-16-229:~$
```

Directory after extracting

# Move extracted Hadoop directory:

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



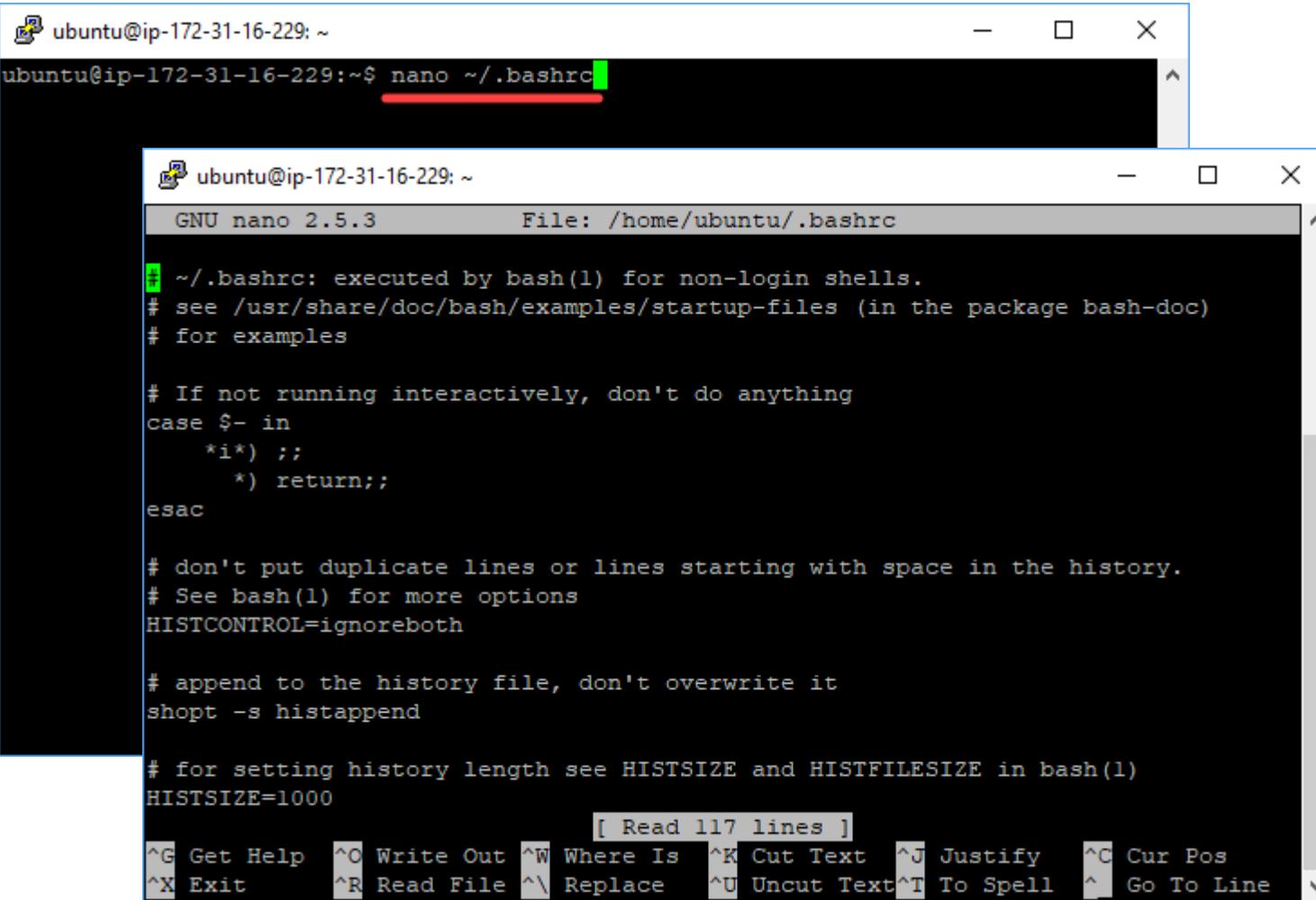
The image shows two terminal windows side-by-side. Both windows have a blue header bar with the text "ubuntu@ip-172-31-16-229: ~". The top window contains the command `sudo mv ./hadoop-2.6.0 /usr/local/hadoop`. The bottom window shows the result of the command: it first displays the command again, then the output of `ls`, which includes "hadoop-2.6.0.tar.gz" and a red message "Hadoop directory gone".

```
ubuntu@ip-172-31-16-229:~$ sudo mv ./hadoop-2.6.0 /usr/local/hadoop
ubuntu@ip-172-31-16-229:~$ ls
hadoop-2.6.0.tar.gz  Hadoop directory gone
ubuntu@ip-172-31-16-229:~$
```

## 2.5 Install Hadoop

# Add path to environment variables:

\$nano ~/.bashrc



The screenshot shows a terminal window with two tabs. The top tab is a standard terminal window titled "ubuntu@ip-172-31-16-229: ~". The bottom tab is the "nano" text editor, also titled "ubuntu@ip-172-31-16-229: ~". The editor window displays the contents of the ".bashrc" file. The file starts with a shebang line and several comments explaining its purpose. It includes logic for non-interactive shells, history control settings (HISTCONTROL=ignoreboth), and a histappend option. It also sets the HISTSIZE variable to 1000. The status bar at the bottom of the editor window indicates "[ Read 117 lines ]".

```
#!/bin/bash
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
    *) return;;
esac

# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

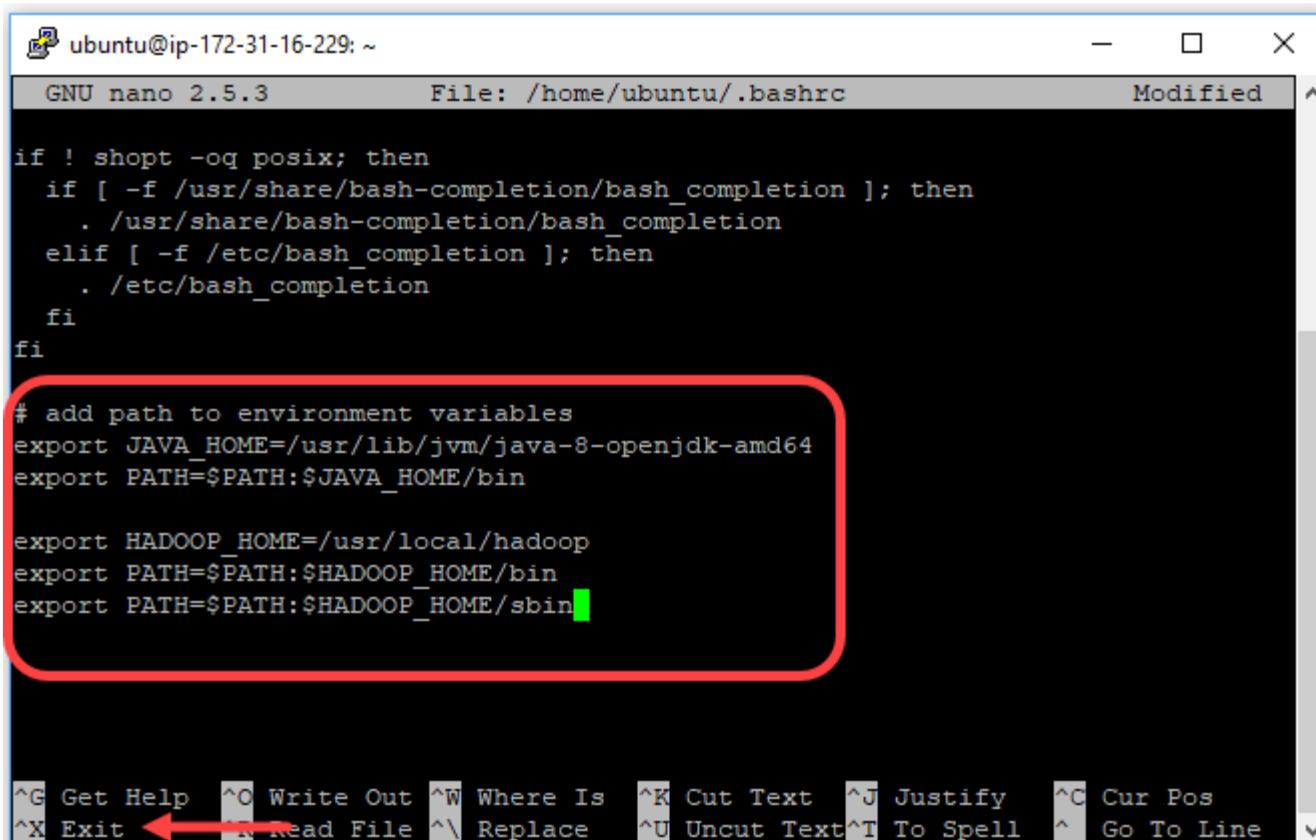
# append to the history file, don't overwrite it
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
```

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
```



The screenshot shows a terminal window titled "ubuntu@ip-172-31-16-229: ~". It displays the contents of the `/home/ubuntu/.bashrc` file using the `GNU nano 2.5.3` editor. The file contains several lines of shell script code. A red box highlights the last three lines, which are the new environment variable assignments. The terminal window has a standard Linux-style interface with a menu bar and a toolbar at the bottom.

```
if ! shopt -oq posix; then  
    if [ -f /usr/share/bash-completion/bash_completion ]; then  
        . /usr/share/bash-completion/bash_completion  
    elif [ -f /etc/bash_completion ]; then  
        . /etc/bash_completion  
    fi  
fi  
  
# add path to environment variables  
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
```

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos  
^X Exit ← ^R Read File ^V Replace ^U Uncut Text ^T To Spell ^ ↑ Go To Line ↓

π

# Save file:

**Ctrl+x, press y, and press enter**

```
ubuntu@ip-172-31-16-229: ~
GNU nano 2.5.3          File: /home/ubuntu/.bashrc

if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

# add path to environment variables
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 modified buffer (ANSWERING "No" WILL DESTROY CHANGES) ?
```

Y Yes  
N No      ^C Cancel

```
ubuntu@ip-172-31-16-229: ~
GNU nano 2.5.3          File: /home/ubuntu/.bashrc

if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

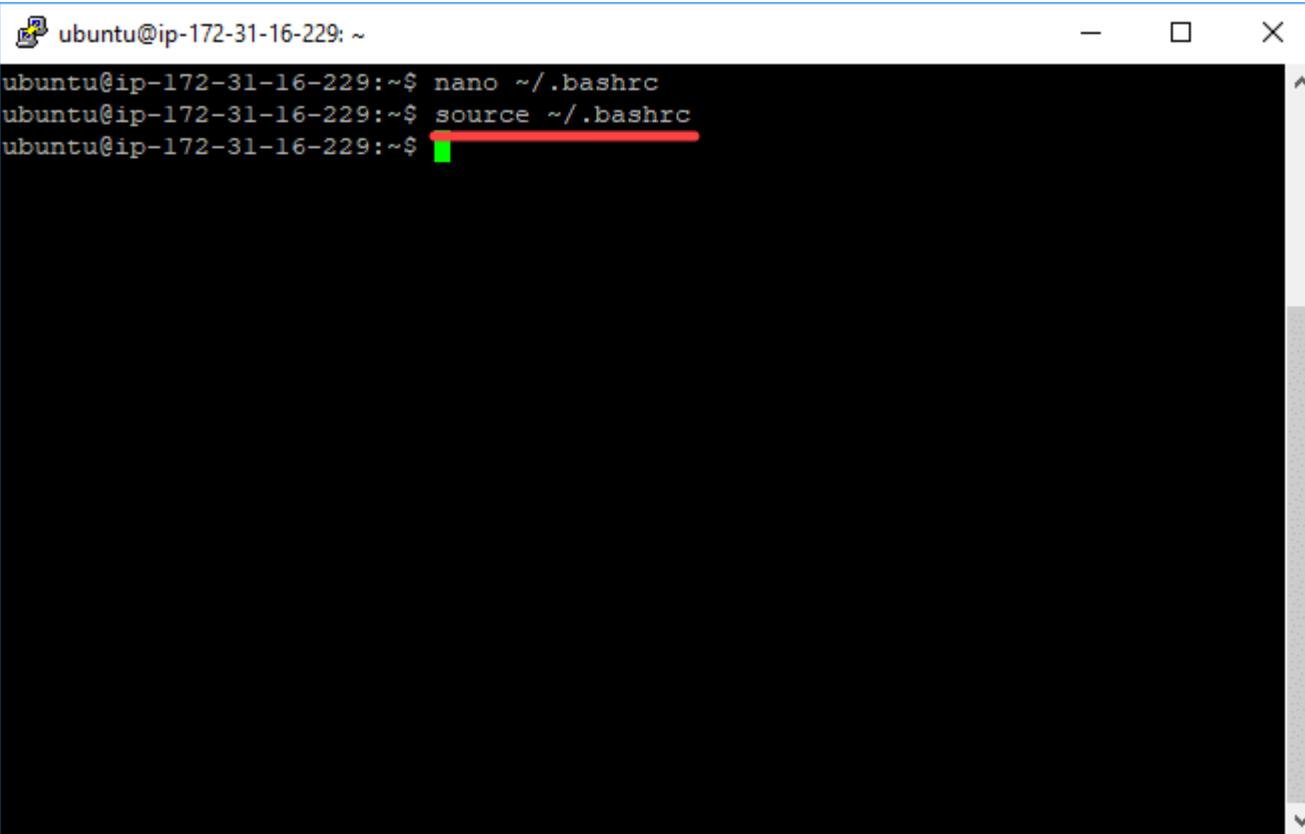
# add path to environment variables
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
export PATH=$PATH:$JAVA_HOME/bin

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

File Name to Write: /home/ubuntu/.bashrc
^G Get Help      M-D DOS Format   M-A Append      M-B Backup File
^C Cancel       M-M Mac Format   M-P Prepend     ^T To Files
```

# Execute environment variables:

**\$source ~/.bashrc**



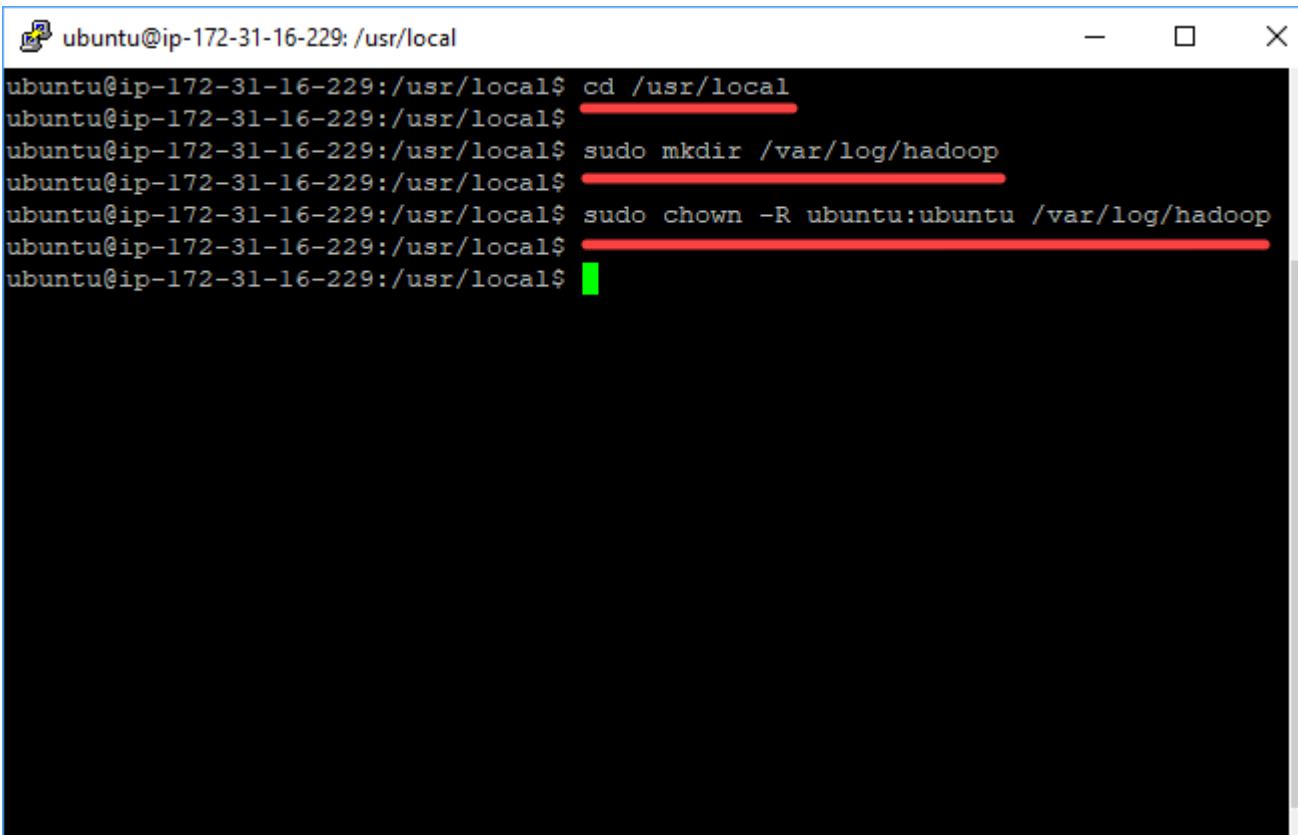
A screenshot of a terminal window titled "ubuntu@ip-172-31-16-229: ~". The window contains the following text:  
ubuntu@ip-172-31-16-229:~\$ nano ~/.bashrc  
ubuntu@ip-172-31-16-229:~\$ source ~/.bashrc  
ubuntu@ip-172-31-16-229:~\$

# Create Hadoop log directory:

```
$cd /usr/local
```

```
$sudo mkdir /var/log/hadoop
```

```
$sudo chown -R ubuntu:ubuntu /var/log/hadoop
```



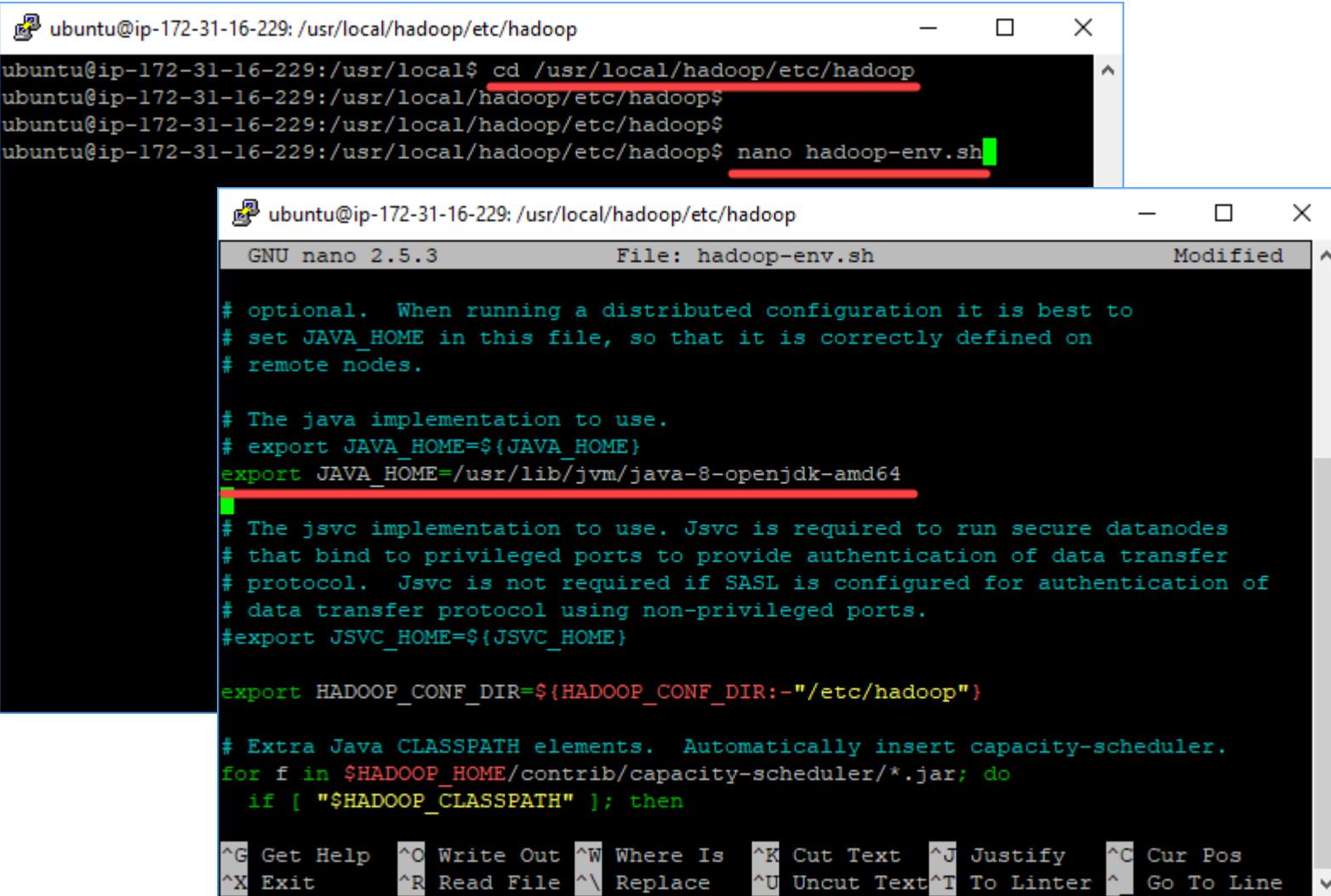
The screenshot shows a terminal window with a blue header bar containing the text "ubuntu@ip-172-31-16-229: /usr/local". The terminal window has standard window controls (minimize, maximize, close) at the top right. The main area of the terminal is black with white text. It displays the following command history:

```
ubuntu@ip-172-31-16-229:/usr/local$ cd /usr/local
ubuntu@ip-172-31-16-229:/usr/local$ sudo mkdir /var/log/hadoop
ubuntu@ip-172-31-16-229:/usr/local$ sudo chown -R ubuntu:ubuntu /var/log/hadoop
```

The command "cd /usr/local" is partially visible at the top. The subsequent two commands, "sudo mkdir /var/log/hadoop" and "sudo chown -R ubuntu:ubuntu /var/log/hadoop", are highlighted with red horizontal bars above them. A small green vertical bar is located at the bottom right of the terminal window.

# Edit Hadoop shell script:

```
$cd /usr/local/hadoop/etc/hadoop  
$nano hadoop-env.sh
```



The screenshot shows two windows. The top window is a terminal session on an Ubuntu system (ip-172-31-16-229) with the following commands entered:

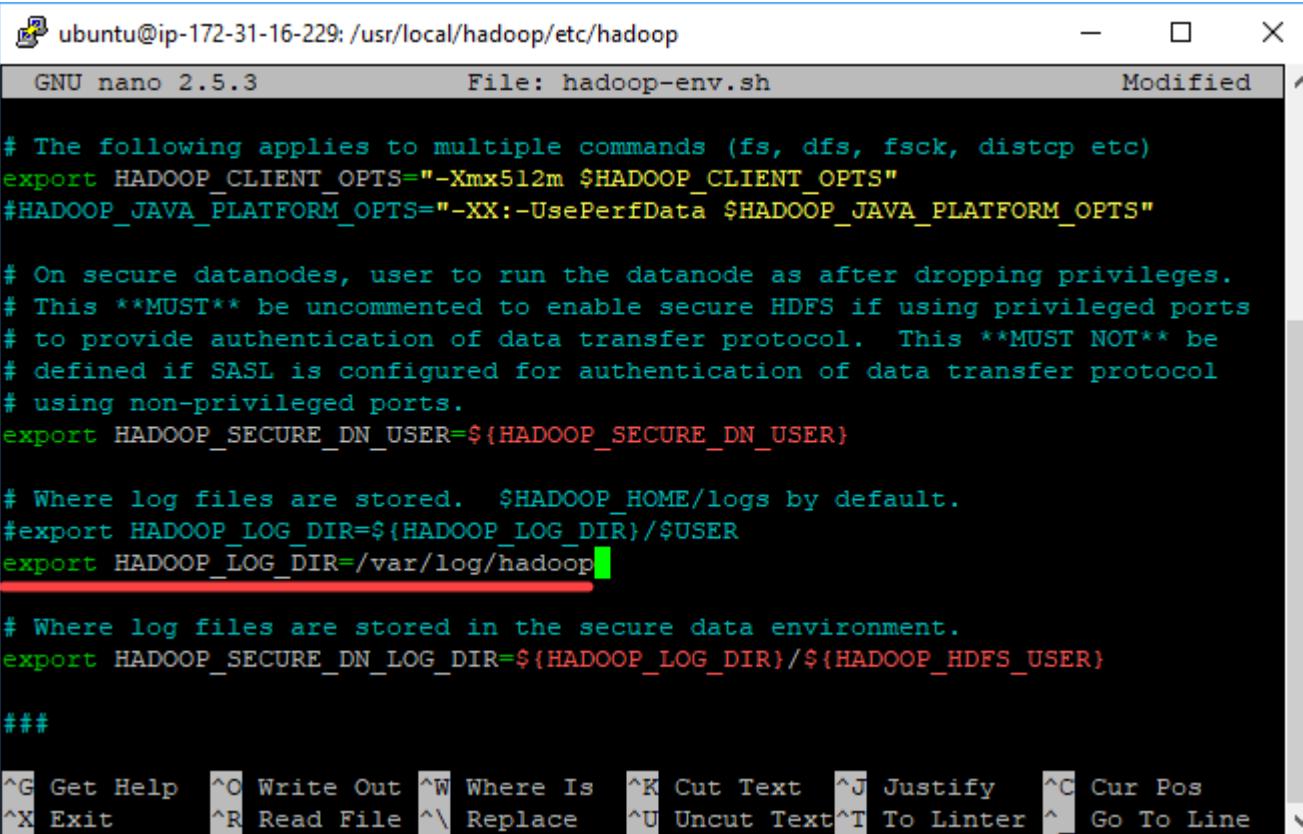
```
ubuntu@ip-172-31-16-229:/usr/local$ cd /usr/local/hadoop/etc/hadoop  
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$  
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$  
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ nano hadoop-env.sh
```

The bottom window is the nano text editor showing the contents of the `hadoop-env.sh` file. The file contains configuration for Java and Hadoop. A specific line is highlighted with a red rectangle:

```
GNU nano 2.5.3          File: hadoop-env.sh          Modified  
  
# 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.  
# export JAVA_HOME=${JAVA_HOME}  
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64  
  
# The jsvc implementation to use. Jsvc is required to run secure datanodes  
# that bind to privileged ports to provide authentication of data transfer  
# protocol. Jsvc is not required if SASL is configured for authentication of  
# data transfer protocol using non-privileged ports.  
#export JSVC_HOME=${JSVC_HOME}  
  
export HADOOP_CONF_DIR=${HADOOP_CONF_DIR:-"/etc/hadoop"}  
  
# Extra Java CLASSPATH elements. Automatically insert capacity-scheduler.  
for f in $HADOOP_HOME/contrib/capacity-scheduler/*.jar; do  
    if [ "$HADOOP_CLASSPATH" ]; then  
  
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos  
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Linter  ^ Go To Line
```

# Edit Hadoop shell script:

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



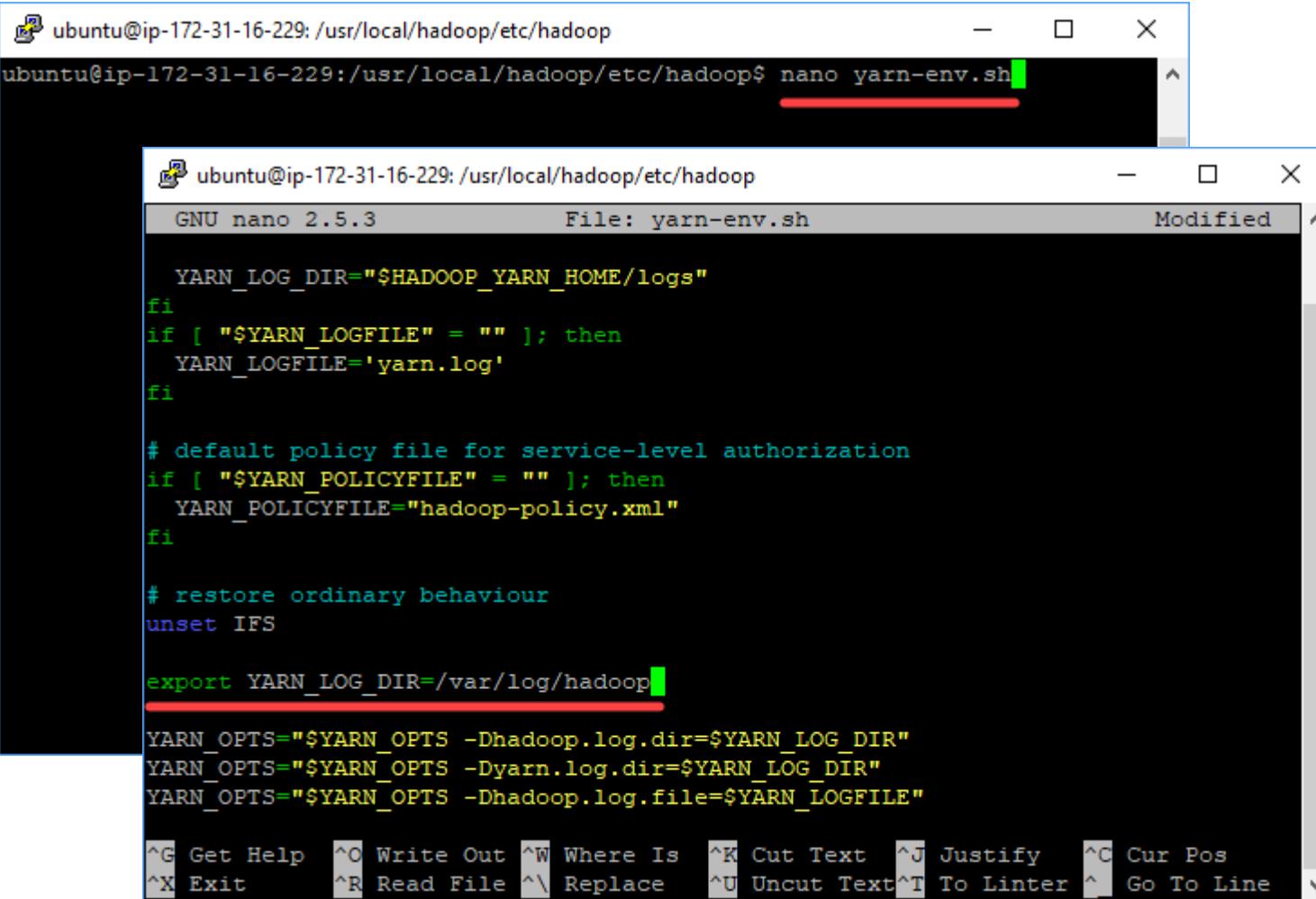
The screenshot shows a terminal window titled "ubuntu@ip-172-31-16-229: /usr/local/hadoop/etc/hadoop". The file being edited is "hadoop-env.sh". The code in the file is as follows:

```
GNU nano 2.5.3          File: hadoop-env.sh          Modified ^  
  
# The following applies to multiple commands (fs, dfs, fsck, distcp etc)  
export HADOOP_CLIENT_OPTS="-Xmx512m $HADOOP_CLIENT_OPTS"  
#HADOOP_JAVA_PLATFORM_OPTS="-XX:-UsePerfData $HADOOP_JAVA_PLATFORM_OPTS"  
  
# On secure datanodes, user to run the datanode as after dropping privileges.  
# This **MUST** be uncommented to enable secure HDFS if using privileged ports  
# to provide authentication of data transfer protocol. This **MUST NOT** be  
# defined if SASL is configured for authentication of data transfer protocol  
# using non-privileged ports.  
export HADOOP_SECURE_DN_USER=${HADOOP_SECURE_DN_USER}  
  
# Where log files are stored. $HADOOP_HOME/logs by default.  
#export HADOOP_LOG_DIR=${HADOOP_LOG_DIR}/${USER}  
export HADOOP_LOG_DIR=/var/log/hadoop  
  
# Where log files are stored in the secure data environment.  
#export HADOOP_SECURE_DN_LOG_DIR=${HADOOP_LOG_DIR}/${HADOOP_HDFS_USER}  
  
###  
  
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos  
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Linter  ^ Go To Line
```

# Edit YARN shell script:

```
$nano yarn-env.sh
```

```
export YARN_LOG_DIR=/var/log/hadoop
```



The screenshot shows a terminal window with two tabs. The top tab is titled "ubuntu@ip-172-31-16-229: /usr/local/hadoop/etc/hadoop" and contains the command "nano yarn-env.sh". The bottom tab is also titled "ubuntu@ip-172-31-16-229: /usr/local/hadoop/etc/hadoop" and shows the contents of the "yarn-env.sh" file in the nano editor. The file contains the following code:

```
YARN_LOG_DIR="$HADOOP_YARN_HOME/logs"
fi
if [ "$YARN_LOGFILE" = "" ]; then
  YARN_LOGFILE='yarn.log'
fi

# default policy file for service-level authorization
if [ "$YARN_POLICYFILE" = "" ]; then
  YARN_POLICYFILE="hadoop-policy.xml"
fi

# restore ordinary behaviour
unset IFS

export YARN_LOG_DIR=/var/log/hadoop
YARN_OPTS="$YARN_OPTS -Dhadoop.log.dir=$YARN_LOG_DIR"
YARN_OPTS="$YARN_OPTS -Dyarn.log.dir=$YARN_LOG_DIR"
YARN_OPTS="$YARN_OPTS -Dhadoop.log.file=$YARN_LOGFILE"
```

The line "export YARN\_LOG\_DIR=/var/log/hadoop" is highlighted with a red rectangle. The nano editor interface includes a menu bar with "File: yarn-env.sh" and "Modified", and a status bar at the bottom with various keyboard shortcuts.

## 2.6 Configure Hadoop



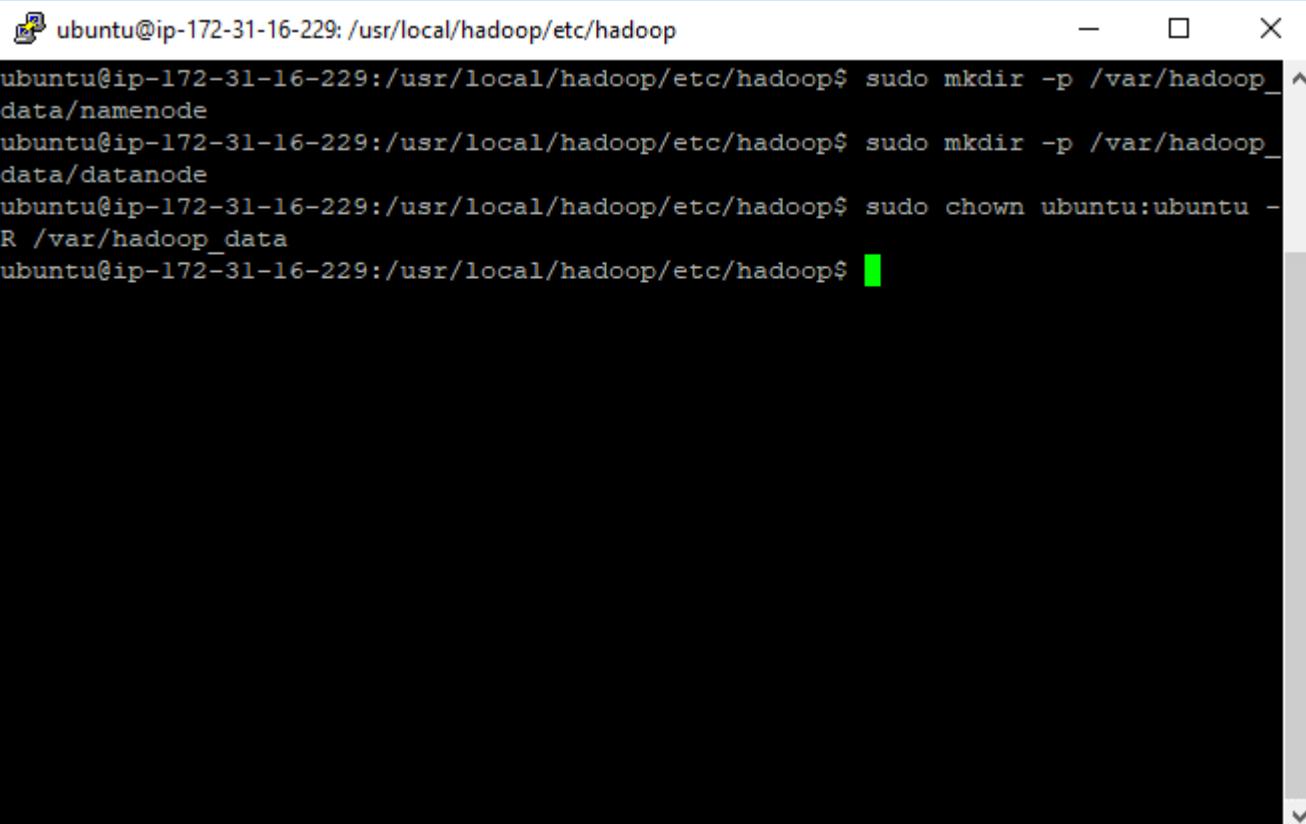
# Edit.hadoop-core-site.xml:

\$nano core-site.xml

```
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ nano core-site.xml
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ nano core-site.xml
GNU nano 2.5.3          File: core-site.xml
distributed under the License
WITHOUT WARRANTY
See the License for
limitations under
-->
<!-- Put site-specific
&lt;configuration&gt;
&lt;/configuration&gt;
Unless re
distribut
WITHOUT W
See the L
limitatio
--&gt;
&lt;!-- Put site specific property overrides in this file. --&gt;
&lt;configuration&gt;
&lt;property&gt;
&lt;name&gt;fs.defaultFS&lt;/name&gt;
&lt;value&gt;hdfs://172.31.16.229:9000&lt;/value&gt;
&lt;/property&gt;
&lt;/configuration&gt;
&lt;configuration&gt;
&lt;property&gt;
&lt;name&gt;fs.defaultFS&lt;/name&gt;
&lt;value&gt;hdfs://172.31.16.229:9000&lt;/value&gt;
&lt;/property&gt;
&lt;/configuration&gt;
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit      ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line</pre>
```

# 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
```

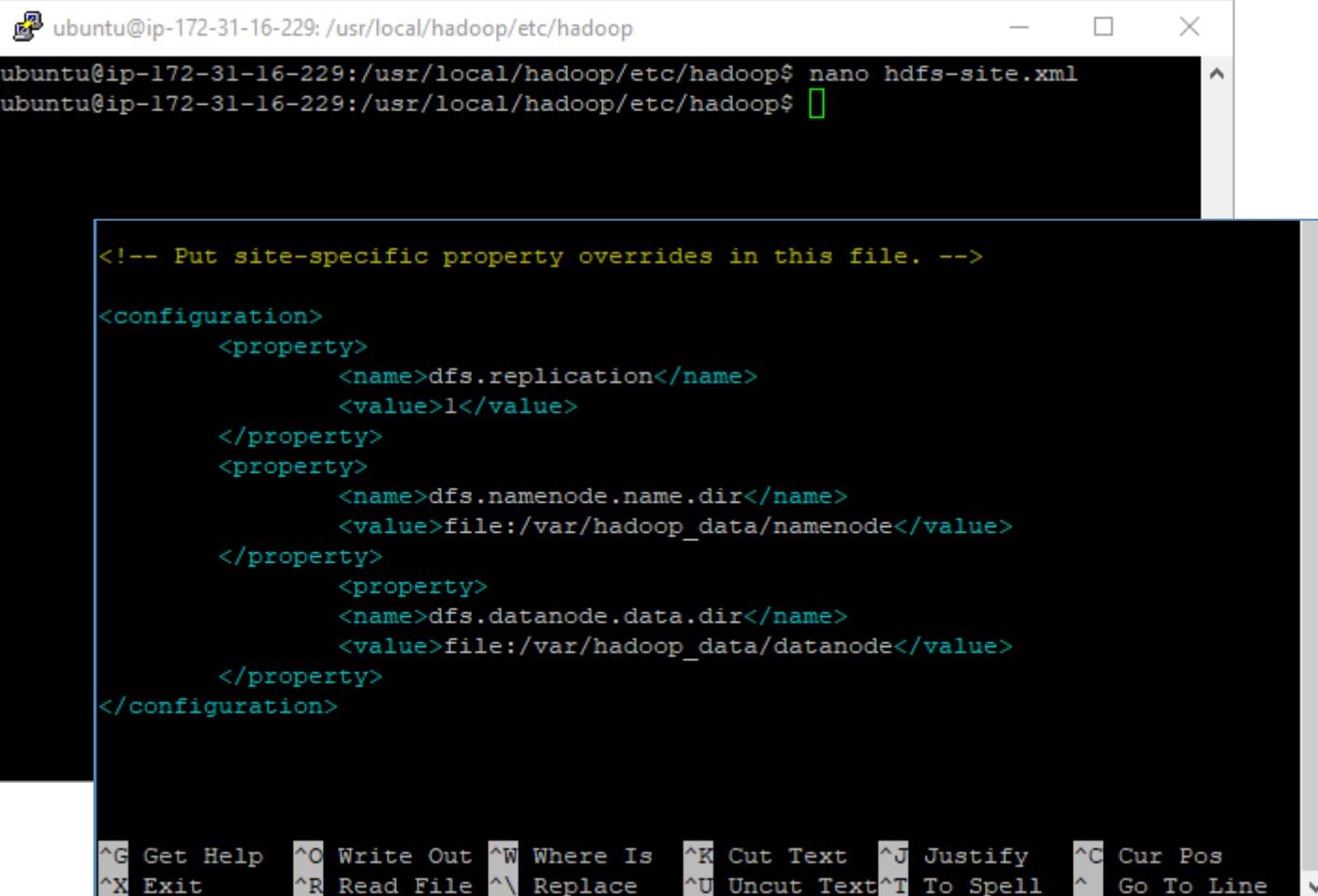


The screenshot shows a terminal window titled "ubuntu@ip-172-31-16-229: /usr/local/hadoop/etc/hadoop". The window contains the following command history:

```
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ sudo mkdir -p /var/hadoop_data/namenode  
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ sudo mkdir -p /var/hadoop_data/datanode  
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ sudo chown ubuntu:ubuntu -R /var/hadoop_data  
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$
```

# Edit hdfs-site.xml:

\$nano hdfs-site.xml



The screenshot shows a terminal window titled "ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop". The command "nano hdfs-site.xml" is run, opening the XML configuration file. The file contains site-specific overrides for HDFS properties. The terminal window has a standard Linux-style interface with a title bar, a scroll bar, and a menu bar at the bottom.

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

<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>
```

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos  
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^ Go To Line

# 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>
```

π

# Edit yarn-site.xml:

\$nano yarn-site.xml

change to  
nano yarn-site.xml

```
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ nano yarn-env.sh
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ nano yarn-site.xml
GNU nano 2.5.3          File: yarn-site.xml

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.

-->
<configuration>

<!-- Site specific YARN configuration properties -->
<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>

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell  ^  Go To Line
```

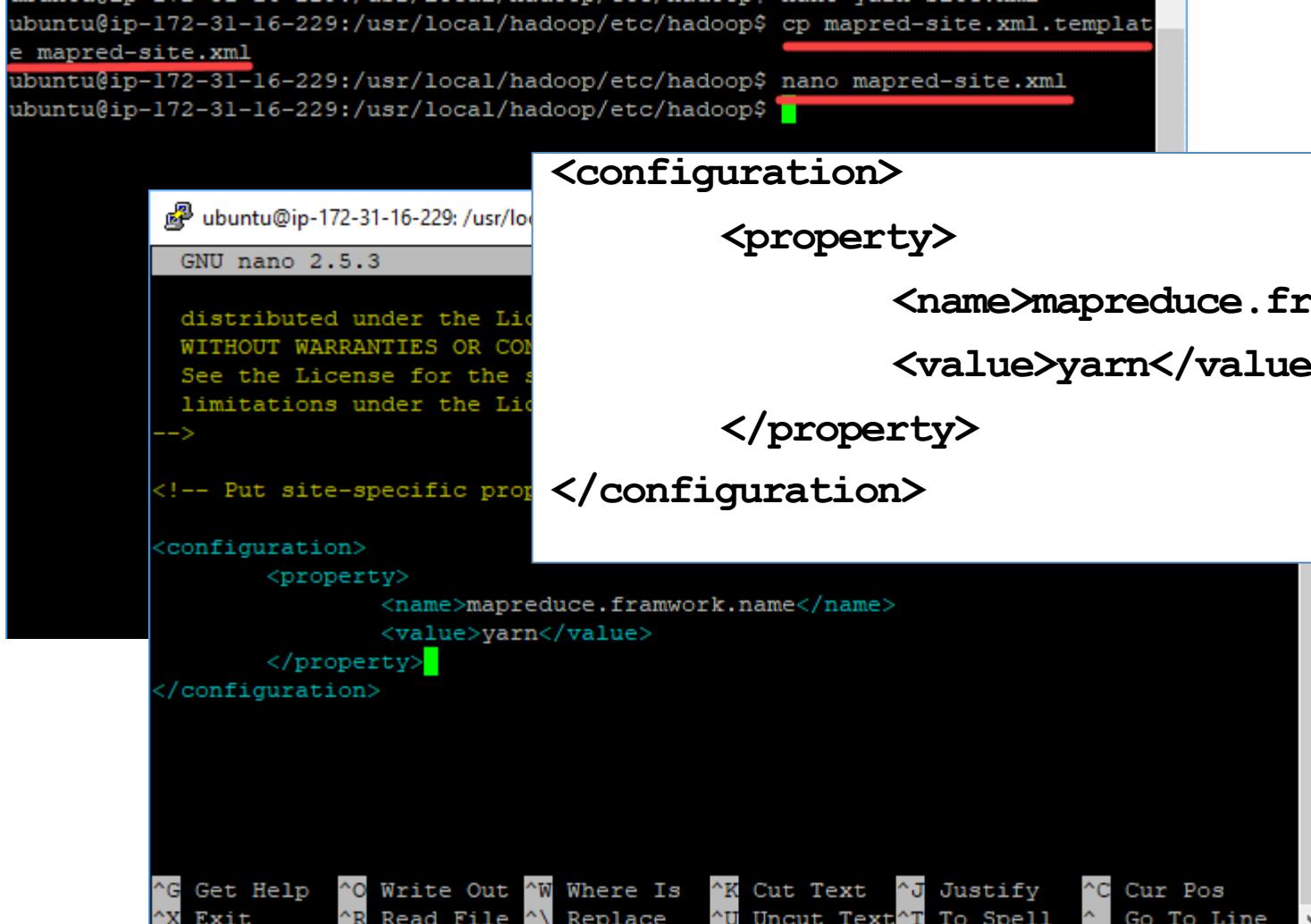
# Edit yarn-site.xml:

```
<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
```



```
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ cp mapred-site.xml.template mapred-site.xml
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ nano mapred-site.xml
```

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

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

GNU nano 2.5.3

distributed under the License  
WITHOUT WARRANTIES OR CONDITIONS  
See the License for the  
limitations under the License

-->

<!-- Put site-specific properties here -->

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos  
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^\_ Go To Line

## 2.7 Format Namenode

π

# Formatting Namenode:

\$hdfs namenode -format

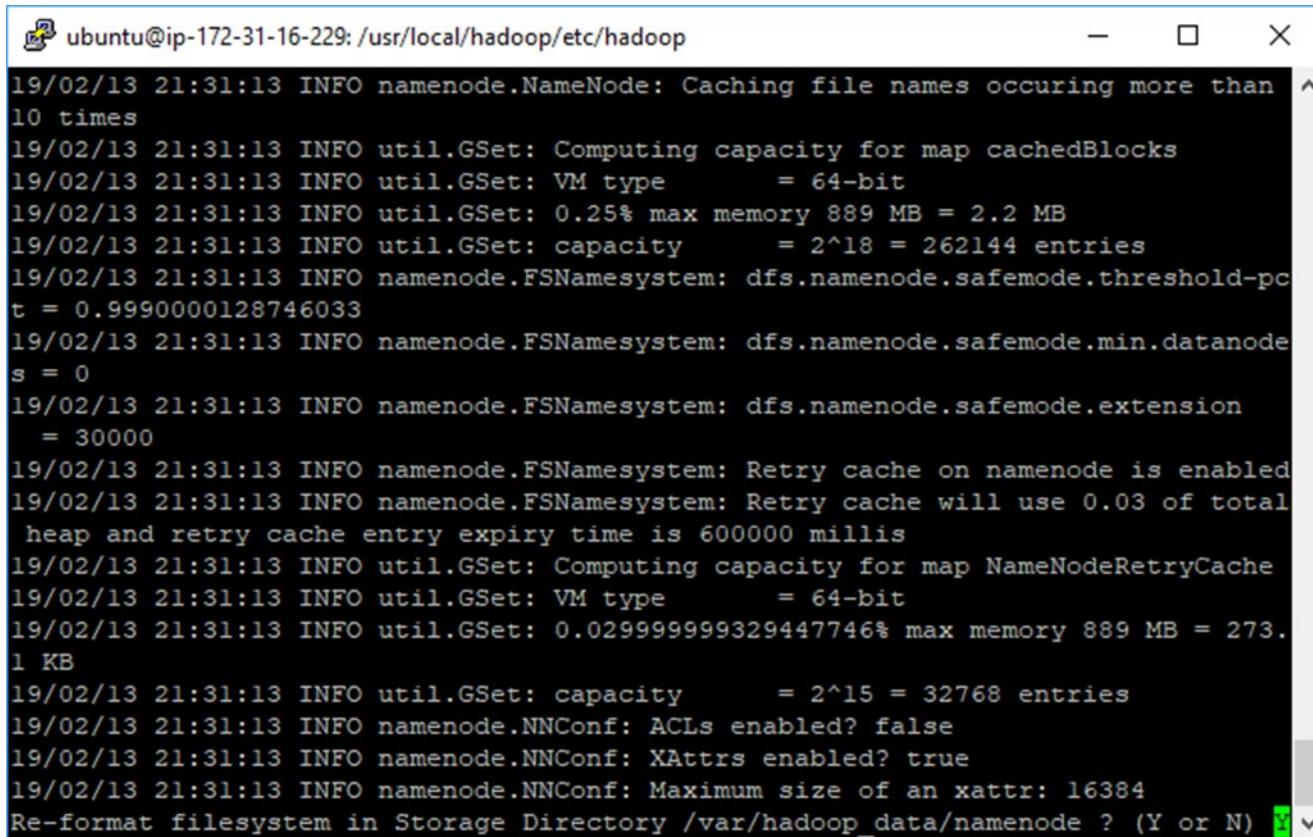


KEEP  
CALM  
AND  
MAY THE FORCE  
BE WITH YOU

```
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ hdfs namenode -format
19/02/13 21:26:19 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
19/02/13 21:26:19 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total
  heap and retry cache entry expiry time is 600000 millis
19/02/13 21:26:19 INFO util.GSet: Computing capacity for map NameNodeRetryCache
19/02/13 21:26:19 INFO util.GSet: VM type      = 64-bit
19/02/13 21:26:19 INFO util.GSet: 0.029999999329447746% max memory 889 MB = 273.
1 KB
19/02/13 21:26:19 INFO util.GSet: capacity      = 2^15 = 32768 entries
19/02/13 21:26:19 INFO namenode.NNConf: ACLs enabled? false
19/02/13 21:26:19 INFO namenode.NNConf: XAttrs enabled? true
19/02/13 21:26:19 INFO namenode.NNConf: Maximum size of an xattr: 16384
19/02/13 21:26:19 INFO namenode.FSImage: Allocated new BlockPoolId: BP-33293181-
172.31.16.229-1550093179263
19/02/13 21:26:19 INFO common.Storage: Storage directory /var/hadoop_data/nameno
de has been successfully formatted.
19/02/13 21:26:19 INFO namenode.NNStorageRetentionManager: Going to retain 1 ima
ges with txid >= 0
19/02/13 21:26:19 INFO util.ExitUtil: Exiting with status 0
19/02/13 21:26:19 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at ip-172-31-16-229.us-west-2.compute.inter
nal/172.31.16.229
************************************************************/
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$
```

# Reformatting Namenode to verify:

\$hdfs namenode -format



A terminal window titled "ubuntu@ip-172-31-16-229: /usr/local/hadoop/etc/hadoop" displaying log output from the HDFS namenode -format command. The log shows various configuration details and asks if the user wants to re-format the filesystem.

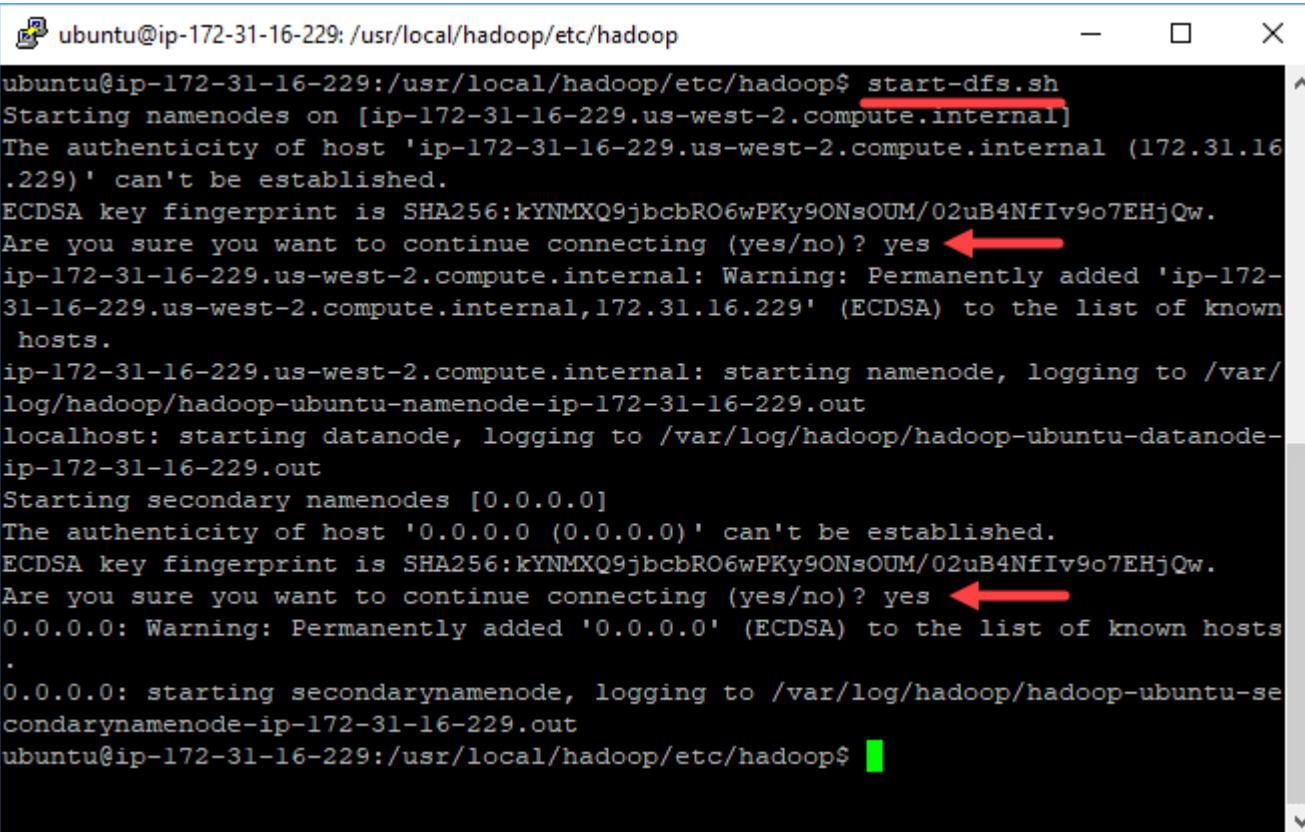
```
19/02/13 21:31:13 INFO namenode.NameNode: Caching file names occurring more than 10 times
19/02/13 21:31:13 INFO util.GSet: Computing capacity for map cachedBlocks
19/02/13 21:31:13 INFO util.GSet: VM type      = 64-bit
19/02/13 21:31:13 INFO util.GSet: 0.25% max memory 889 MB = 2.2 MB
19/02/13 21:31:13 INFO util.GSet: capacity      = 2^18 = 262144 entries
19/02/13 21:31:13 INFO namenode.FSNamesystem: dfs.namenode.safemode.threshold-pct = 0.9990000128746033
19/02/13 21:31:13 INFO namenode.FSNamesystem: dfs.namenode.safemode.min.datanodes = 0
19/02/13 21:31:13 INFO namenode.FSNamesystem: dfs.namenode.safemode.extension = 30000
19/02/13 21:31:13 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
19/02/13 21:31:13 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache entry expiry time is 600000 millis
19/02/13 21:31:13 INFO util.GSet: Computing capacity for map NameNodeRetryCache
19/02/13 21:31:13 INFO util.GSet: VM type      = 64-bit
19/02/13 21:31:13 INFO util.GSet: 0.029999999329447746% max memory 889 MB = 273.1 KB
19/02/13 21:31:13 INFO util.GSet: capacity      = 2^15 = 32768 entries
19/02/13 21:31:13 INFO namenode.NNConf: ACLs enabled? false
19/02/13 21:31:13 INFO namenode.NNConf: XAttrs enabled? true
19/02/13 21:31:13 INFO namenode.NNConf: Maximum size of an xattr: 16384
Re-format filesystem in Storage Directory /var/hadoop_data/namenode ? (Y or N) Y
```

## 2.8 Start Hadoop



# Start Namenode and Datanode:

**\$start-dfs.sh**

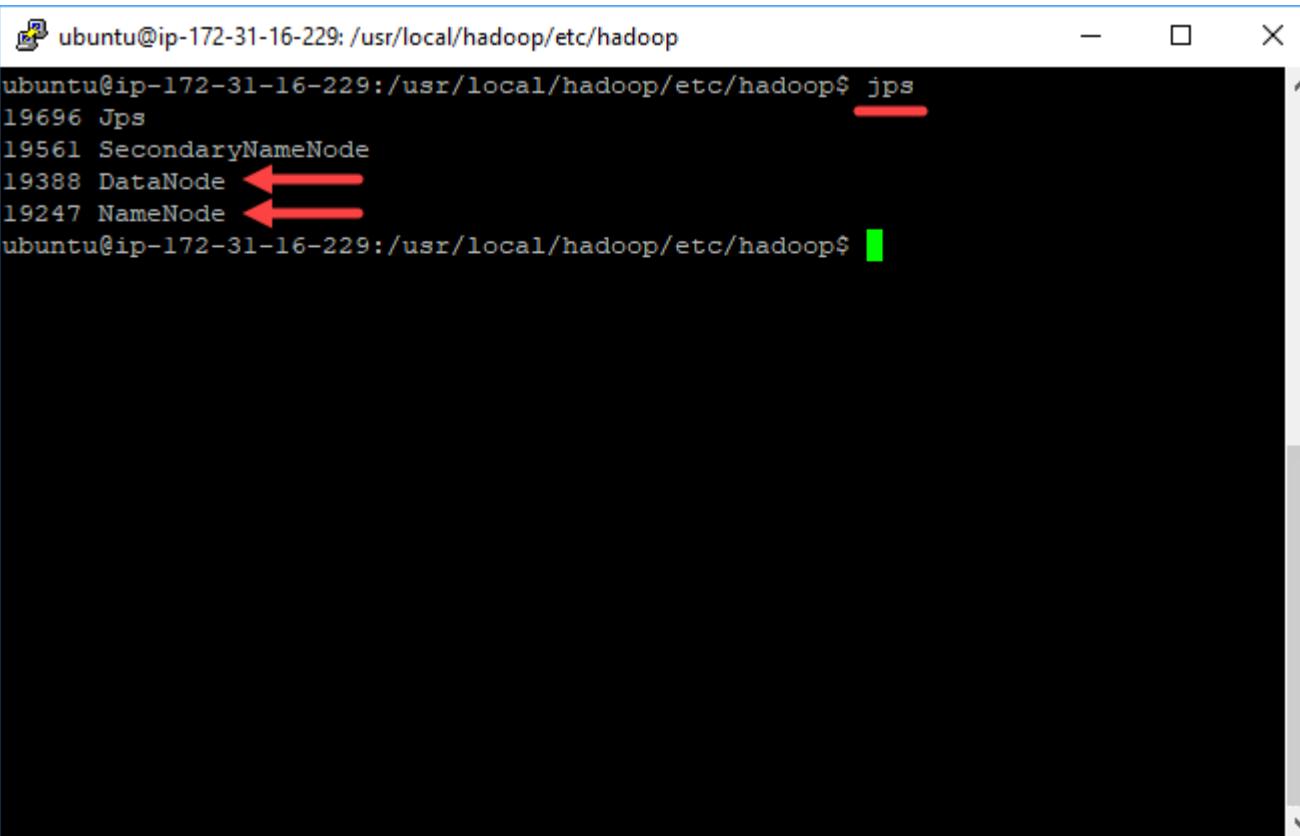


```
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ start-dfs.sh
Starting namenodes on [ip-172-31-16-229.us-west-2.compute.internal]
The authenticity of host 'ip-172-31-16-229.us-west-2.compute.internal (172.31.16
.229)' can't be established.
ECDSA key fingerprint is SHA256:kYNMXQ9jbcBRO6wPKy9ONsOUM/02uB4NfIv9o7EHjQw.
Are you sure you want to continue connecting (yes/no)? yes ←
ip-172-31-16-229.us-west-2.compute.internal: Warning: Permanently added 'ip-172-
31-16-229.us-west-2.compute.internal,172.31.16.229' (ECDSA) to the list of known
hosts.
ip-172-31-16-229.us-west-2.compute.internal: starting namenode, logging to /var/
log/hadoop/hadoop-ubuntu-namenode-ip-172-31-16-229.out
localhost: starting datanode, logging to /var/log/hadoop/hadoop-ubuntu-datanode-
ip-172-31-16-229.out
Starting secondary namenodes [0.0.0.0]
The authenticity of host '0.0.0.0 (0.0.0.0)' can't be established.
ECDSA key fingerprint is SHA256:kYNMXQ9jbcBRO6wPKy9ONsOUM/02uB4NfIv9o7EHjQw.
Are you sure you want to continue connecting (yes/no)? yes ←
0.0.0.0: Warning: Permanently added '0.0.0.0' (ECDSA) to the list of known hosts
.
0.0.0.0: starting secondarynamenode, logging to /var/log/hadoop/hadoop-ubuntu-se
condarynamenode-ip-172-31-16-229.out
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$
```

# Verify Namenode and Datanode:

$\pi$

\$jps



```
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ jps
19696 Jps
19561 SecondaryNameNode
19388 DataNode ←
19247 NameNode ←
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$
```

# Start YARN:

**\$start-yarn.sh**

The image shows two terminal windows side-by-side. The left window displays the command `start-yarn.sh` being run, followed by logs indicating the startup of the Resource Manager and Node Manager daemons. The right window shows the output of the `jps` command, listing the Resource Manager process (pid 19762) and other Hadoop components.

```
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /var/log/hadoop/yarn-ubuntu-resourcemanager-ip-172-31-16-229.out
localhost: starting nodemanager, logging to /var/log/hadoop/yarn-ubuntu-nodemanager-ip-172-31-16-229.out
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$
```

```
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$ jps
19793 Jps
19762 ResourceManager ←
19561 SecondaryNameNode
19388 DataNode
19247 NameNode
ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop$
```

## 2.9 Access Hadoop Web Console

# Checking Public IP and DNS

π

The screenshot shows the AWS EC2 Instances page. On the left, the navigation pane includes links for EC2 Dashboard, Events, Tags, Reports, Limits, Instances (selected), Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, AMIs, and Elastic Block Store. The main content area displays a table with one row for an instance named "BDA-Hadoop...". A context menu is open over this instance, with the "Connect" option highlighted. The "Actions" dropdown menu also has a "Refresh" item circled in red. Below the table, the instance details are shown: Instance ID: i-0dd57077f57bc13c0, Public DNS: ec2-52-26-15-54.us-west-2.compute.amazonaws.com. The "Description" tab is selected. The instance state is running, type is m5.large, and the availability zone is us-west-2b. The elastic IPs section shows two entries: a public IP (52.26.15.54) and a private IP (172.31.16.229). A red arrow points from the "Change after time passing or restarting the instance" note to the public IP entry. Another red box highlights the public IP entry, and another red box highlights the private IP entry.

Public DNS (IPv4): ec2-52-26-15-54.us-west-2.compute.amazonaws.com  
IPv4 Public IP: 52.26.15.54

Private DNS: ip-172-31-16-229.us-west-2.compute.internal  
Private IPs: 172.31.16.229

# Namenode Information:

<http://52.26.15.54:50070>

The screenshot shows the Hadoop NameNode Overview page. The top navigation bar includes links for Hadoop, Overview, Datanodes, Snapshot, Startup Progress, and Utilities. The main content area is titled "Overview 'ip-172-31-16-229.us-west-2.compute.internal:9000' (active)". Below this, there is a table with the following data:

|                |   |
|----------------|---|
| Started:       | Wed Feb 13 21:38:16 UTC 2019                              |
| Version:       | 2.6.0, re3496499ecb8d220fba99dc5ed4c99c8f9e33bb1          |
| Compiled:      | 2014-11-13T21:10Z by jenkins from (detached from e349649) |
| Cluster ID:    | CID-f54b60ab-50a0-4722-92d9-e7efb83dfc78                  |
| Block Pool ID: | BP-1448748555-172.31.16.229-1550093483004                 |

## Summary

Security is off.  
Safemode is off.  
4 files and directories, 1 blocks = 5 total filesystem object(s).  
Heap Memory used 47.3 MB of 283.5 MB Heap Memory. Max Heap Memory is 889 MB.  
Non Heap Memory used 48.33 MB of 49.27 MB Committed Non Heap Memory. Max Non Heap Memory is -1 B.

|                      |          |
|----------------------|----------|
| Configured Capacity: | 19.32 GB |
| DFS Used:            | 752 KB   |
| Non DFS Used:        | 2.07 GB  |
| DFS Remaining:       | 17.25 GB |

# 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
```

The screenshot shows two terminal windows side-by-side. The left window displays the command \$wget followed by a URL, which is then resolved and downloaded. The right window shows the resulting file being renamed from 1342-0.txt to input\_data.txt.

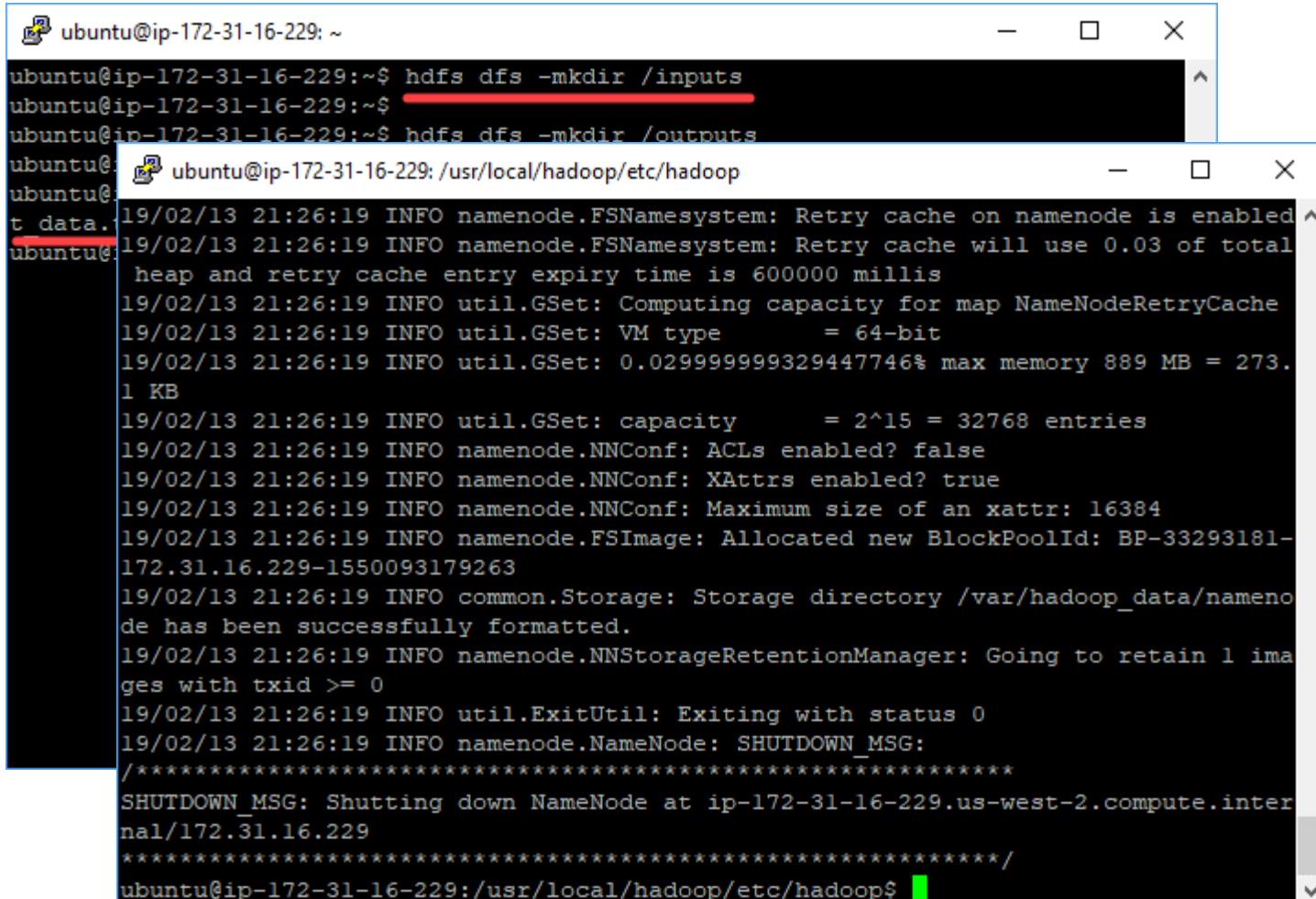
```
ubuntu@ip-172-31-16-229:~$ cd  
ubuntu@ip-172-31-16-229:~$ wget https://www.gutenberg.org/files/1342/1342-0.txt  
--2019-02-13 22:14:41-- https://www.gutenberg.org/files/1342/1342-0.txt  
Resolving www.gutenberg.org (www.gutenberg.org)... 152.19.134.47, 2610:28:3090:3  
000:0:bad:cfa:15  
Connecting to www.gutenberg.org (www.gutenberg.org)|152.19.134.47|:80... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 724 [text/plain]  
Saving to: 1342-0.txt  
1342-0.txt  
2019-02-13 22:14:41  
ubuntu@ip-172-31-16-229:~$ ls  
1342-0.txt hadoop-2.6.0.tar.gz  
ubuntu@ip-172-31-16-229:~$ mv 1342-0.txt input_data.txt  
ubuntu@ip-172-31-16-229:~$ ls  
hadoop-2.6.0.tar.gz input_data.txt  
ubuntu@ip-172-31-16-229:~$
```

# Importing data to HDFS:

```
$hdfs dfs -mkdir /inputs
```

```
$hdfs dfs -mkdir /outputs
```

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



The screenshot shows a terminal window with two tabs. The left tab displays the command history and execution of HDFS commands:

```
ubuntu@ip-172-31-16-229:~$ hdfs dfs -mkdir /inputs
ubuntu@ip-172-31-16-229:~$ hdfs dfs -mkdir /outputs
ubuntu@ip-172-31-16-229:~$ hdfs dfs -copyFromLocal ./input_data.txt /inputs/input_data.txt
```

The right tab shows the log output from the NameNode process, which includes configuration details and the start of the shutdown sequence:

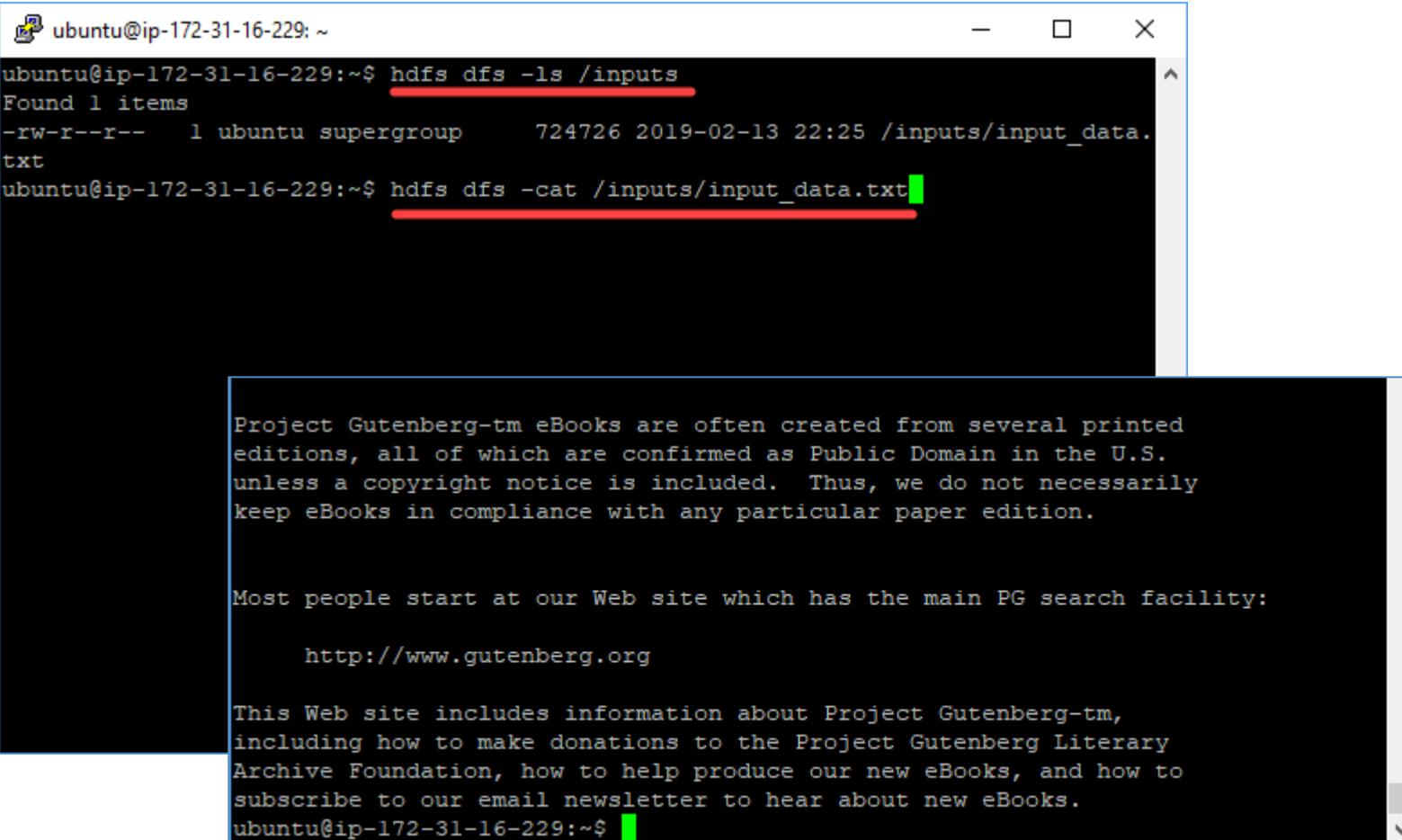
```
19/02/13 21:26:19 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
19/02/13 21:26:19 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total
heap and retry cache entry expiry time is 600000 millis
19/02/13 21:26:19 INFO util.GSet: Computing capacity for map NameNodeRetryCache
19/02/13 21:26:19 INFO util.GSet: VM type          = 64-bit
19/02/13 21:26:19 INFO util.GSet: 0.029999999329447746% max memory 889 MB = 273.
1 KB
19/02/13 21:26:19 INFO util.GSet: capacity        = 2^15 = 32768 entries
19/02/13 21:26:19 INFO namenode.NNConf: ACLs enabled? false
19/02/13 21:26:19 INFO namenode.NNConf: XAttrs enabled? true
19/02/13 21:26:19 INFO namenode.NNConf: Maximum size of an xattr: 16384
19/02/13 21:26:19 INFO namenode.FSImage: Allocated new BlockPoolId: BP-33293181-
172.31.16.229-1550093179263
19/02/13 21:26:19 INFO common.Storage: Storage directory /var/hadoop_data/namenode has been successfully formatted.
19/02/13 21:26:19 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
19/02/13 21:26:19 INFO util.ExitUtil: Exiting with status 0
19/02/13 21:26:19 INFO namenode.NameNode: SHUTDOWN_MSG:
*****SHUTDOWN_MSG: Shutting down NameNode at ip-172-31-16-229.us-west-2.compute.internal/172.31.16.229
*****
```

ubuntu@ip-172-31-16-229:/usr/local/hadoop/etc/hadoop\$

# Verify data in HDFS:

```
$hdfs dfs -ls /inputs
```

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



The screenshot shows a terminal window on an Ubuntu system (version 19.04) with a blue title bar. The title bar displays the command: `ubuntu@ip-172-31-16-229:~$`. The terminal content is as follows:

```
ubuntu@ip-172-31-16-229:~$ hdfs dfs -ls /inputs
Found 1 items
-rw-r--r-- 1 ubuntu supergroup 724726 2019-02-13 22:25 /inputs/input_data.txt
ubuntu@ip-172-31-16-229:~$ hdfs dfs -cat /inputs/input_data.txt
```

The output of the `-cat` command is displayed in a large black box at the bottom of the terminal window. It contains the following text:

Project Gutenberg-tm eBooks are often created from several printed editions, all of which are confirmed as Public Domain in the U.S. unless a copyright notice is included. Thus, we do not necessarily keep eBooks in compliance with any particular paper edition.

Most people start at our Web site which has the main PG search facility:

<http://www.gutenberg.org>

This Web site includes information about Project Gutenberg-tm, including how to make donations to the Project Gutenberg Literary Archive Foundation, how to help produce our new eBooks, and how to subscribe to our email newsletter to hear about new eBooks.

```
ubuntu@ip-172-31-16-229:~$
```

# Verify data using web console:

<http://52.26.15.54:50070>

Utilities -> Browse the file system ->inputs

The screenshot shows the Hadoop Web Console interface. At the top, there is a navigation bar with links: Hadoop, Overview, Datanodes, Snapshot, Startup Progress, Utilities (with a red circle '1' over it), and a dropdown menu. The dropdown menu contains 'Browse the file system' (with a red circle '2' over it) and Logs. Below the navigation bar, the page title is 'Browse Directory'. There is a search bar with a '/' icon and a 'Go!' button. The main content area displays a table of files and directories. The table has columns: Permission, Owner, Group, Size, Replication, Block Size, and Name. Two entries are listed: 'inputs' (with a red circle '3' over it) and 'outputs'. Both entries have permissions drwxr-xr-x, owned by ubuntu, belong to supergroup, and have 0 B size, 0 replication, and 0 B block size. At the bottom of the page, the text 'Hadoop, 2014.' is visible, and at the very bottom, the URL '52.26.15.54:50070/explorer.html#' is shown.

| Permission | Owner  | Group      | Size | Replication | Block Size | Name    |
|------------|--------|------------|------|-------------|------------|---------|
| drwxr-xr-x | ubuntu | supergroup | 0 B  | 0           | 0 B        | inputs  |
| drwxr-xr-x | ubuntu | supergroup | 0 B  | 0           | 0 B        | outputs |

# Verify by web console:

**http://52.26.15.54:50070**

**Utilities -> Browse the file system ->inputs**

The screenshot shows the Hadoop Web UI interface. At the top, there is a green navigation bar with the following links: Hadoop, Overview, Datanodes, Snapshot, Startup Progress, Utilities, and a dropdown menu. Below the navigation bar, the page title is "Browse Directory". A search bar contains the path "/inputs" and a "Go!" button. The main content area displays a table with the following data:

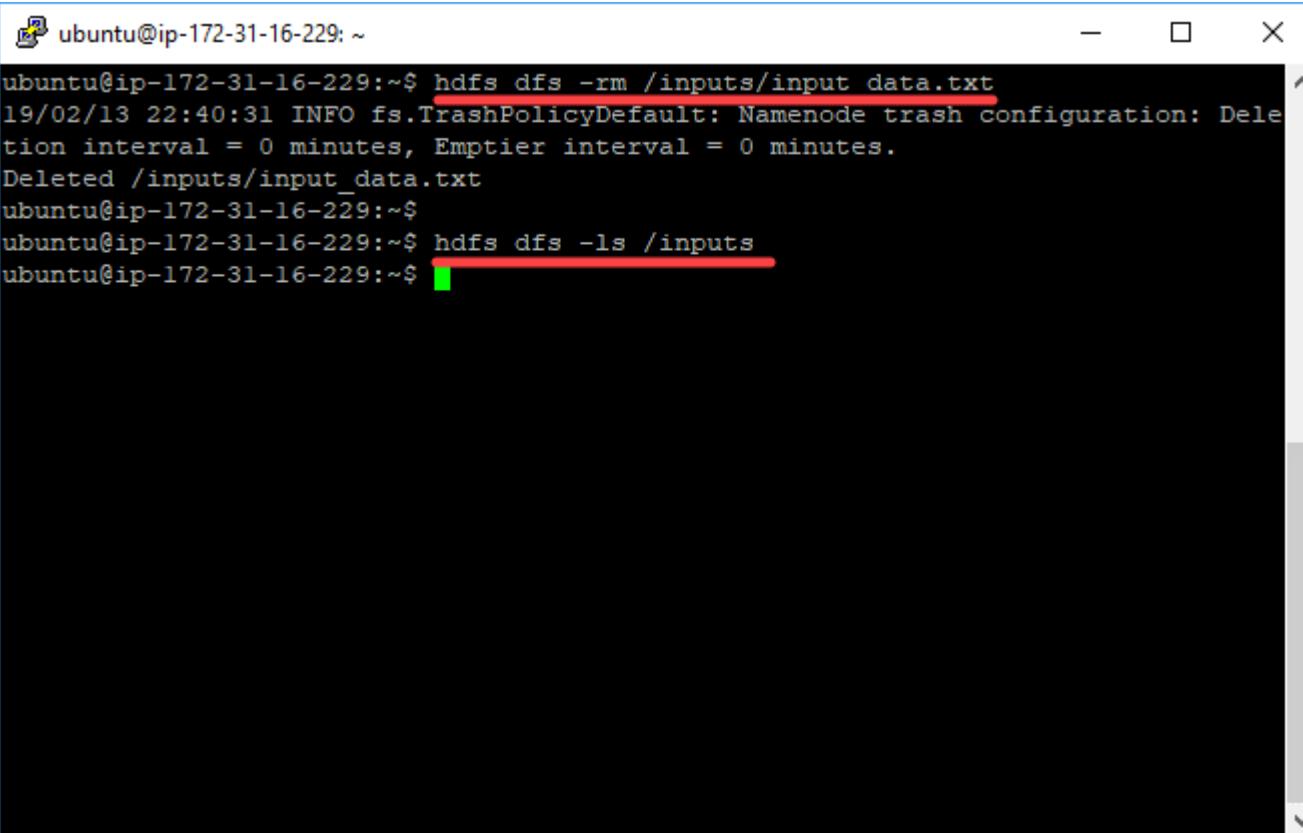
| Permission | Owner  | Group      | Size      | Replication | Block Size | Name                           |
|------------|--------|------------|-----------|-------------|------------|--------------------------------|
| -rw-r--r-- | ubuntu | supergroup | 707.74 KB | 1           | 128 MB     | <a href="#">input_data.txt</a> |

At the bottom of the page, there is a footer note: "Hadoop, 2014."

# Remove data from HDFS:

π

```
$hdfs dfs -rm /inputs/input_data.txt
```



The screenshot shows a terminal window titled "ubuntu@ip-172-31-16-229: ~". The terminal displays the following command and its execution:

```
ubuntu@ip-172-31-16-229:~$ hdfs dfs -rm /inputs/input_data.txt
19/02/13 22:40:31 INFO fs.TrashPolicyDefault: Namenode trash configuration: Dele
tion interval = 0 minutes, Emettier interval = 0 minutes.
Deleted /inputs/input_data.txt
ubuntu@ip-172-31-16-229:~$ hdfs dfs -ls /inputs
ubuntu@ip-172-31-16-229:~$
```

The command `hdfs dfs -rm /inputs/input_data.txt` is highlighted with a red underline. The command `hdfs dfs -ls /inputs` is also highlighted with a red underline. A small green square is visible at the bottom left of the terminal window.

# Verify using web console:

<http://52.26.15.54:50070>

Utilities -> Browse the file system ->inputs

The screenshot shows the Hadoop Web Console interface. At the top, there is a green navigation bar with the following links: Hadoop, Overview, Datanodes, Snapshot, Startup Progress, Utilities, and a dropdown menu. Below the navigation bar, the main content area has a title 'Browse Directory'. A search bar contains the path '/inputs'. Underneath the search bar is a table header with columns: Permission, Owner, Group, Size, Replication, Block Size, and Name. The table body is currently empty, displaying only the footer text 'Hadoop, 2014.'

## 2.11 Stop Hadoop



# Stop YARN:

\$stop-yarn.sh

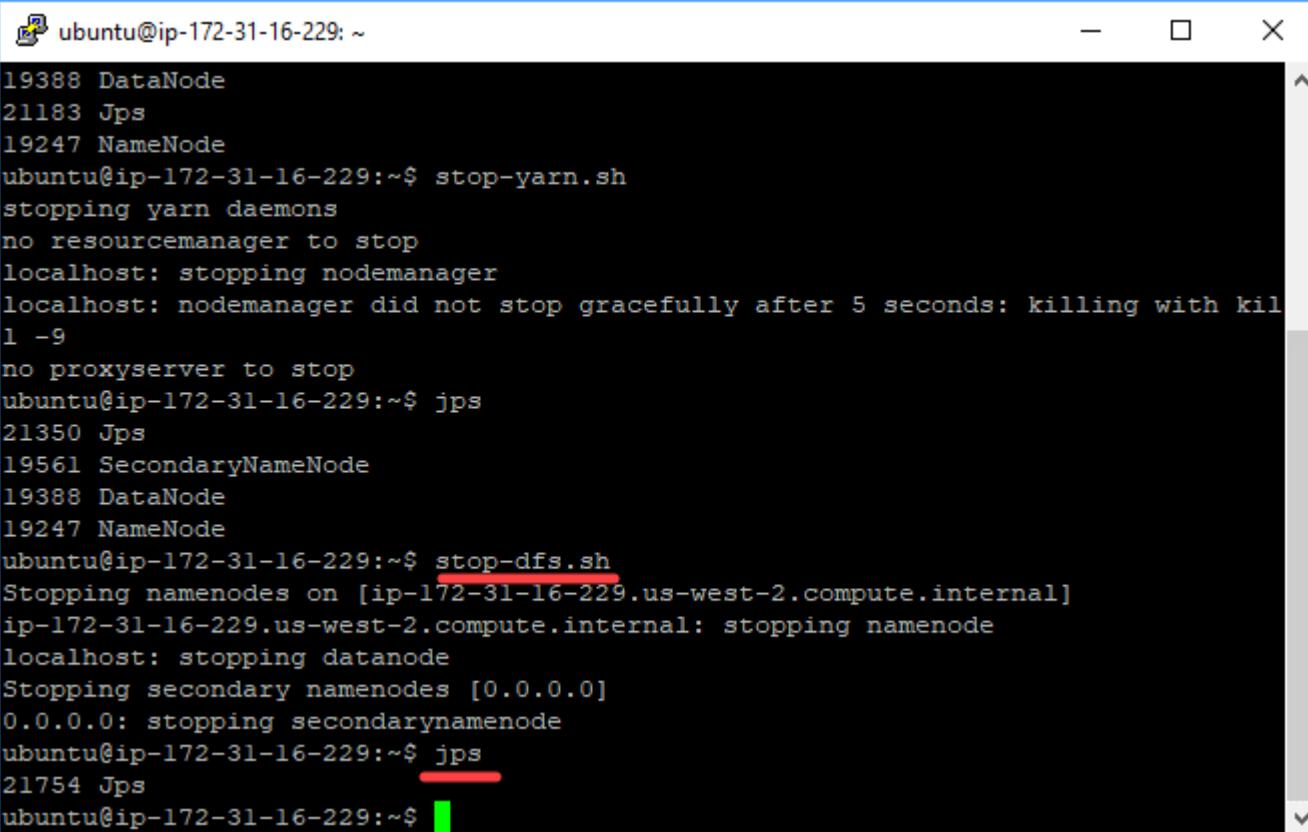
The screenshot shows a terminal window with a blue header bar containing the text "ubuntu@ip-172-31-16-229: ~". The main area of the terminal displays the following command sequence and its output:

```
ubuntu@ip-172-31-16-229:~$ jps
21029 NodeManager
19561 SecondaryNameNode
19388 DataNode
21183 Jps
19247 NameNode
ubuntu@ip-172-31-16-229:~$ stop-yarn.sh
stopping yarn daemons
no resourcemanager to stop
localhost: stopping nodemanager
localhost: nodemanager did not stop gracefully after 5 seconds: killing with kill
1 -9
no proxyserver to stop
ubuntu@ip-172-31-16-229:~$ jps
21350 Jps
19561 SecondaryNameNode
19388 DataNode
19247 NameNode
ubuntu@ip-172-31-16-229:~$ stop-dfs.sh
```

The command `stop-yarn.sh` is highlighted with a red underline. The command `stop-dfs.sh` at the bottom is highlighted with a green underline.

# Stop DFS:

## \$stop-dfs.sh



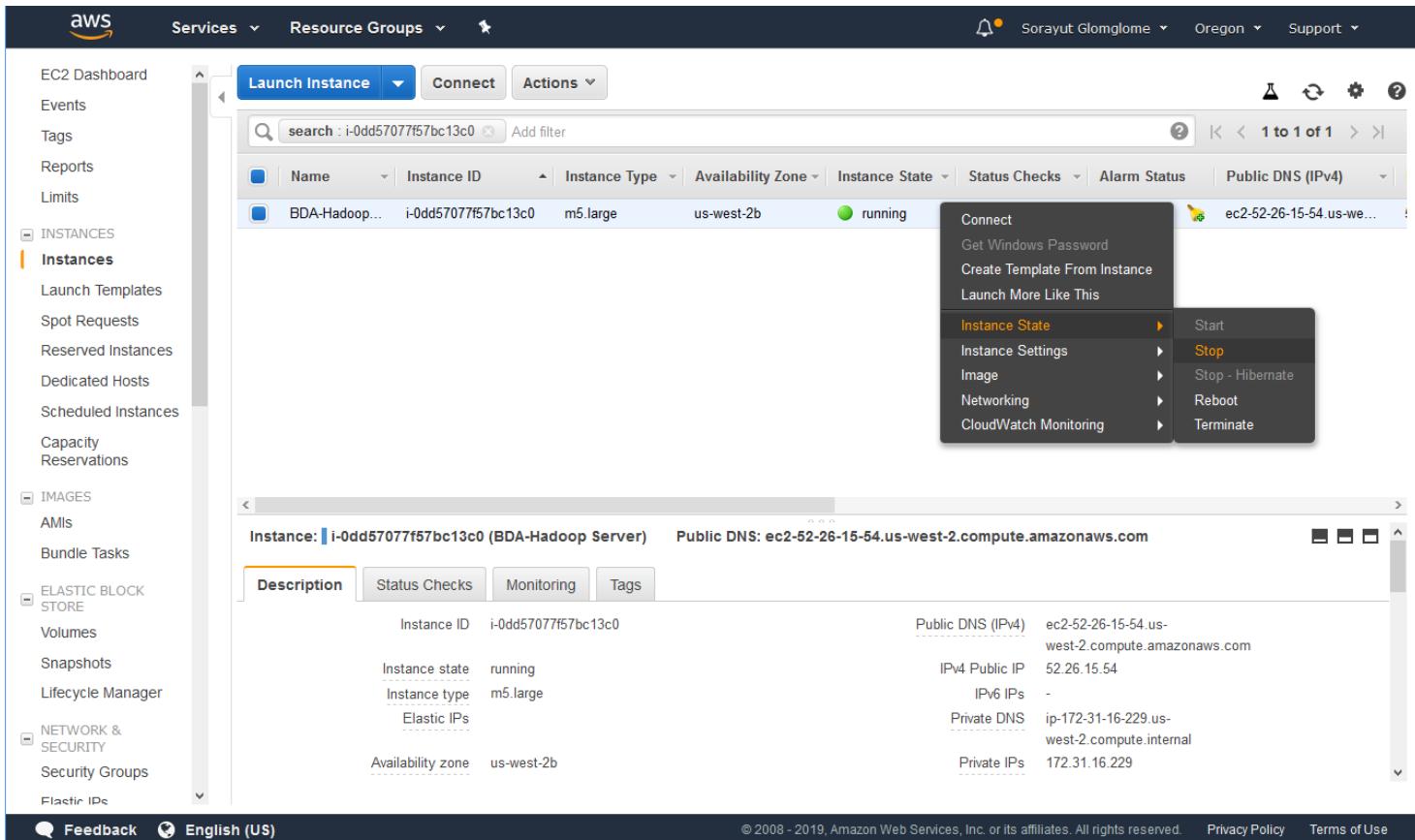
The screenshot shows a terminal window titled "ubuntu@ip-172-31-16-229: ~". The terminal displays the following command-line session:

```
19388 DataNode
21183 Jps
19247 NameNode
ubuntu@ip-172-31-16-229:~$ stop-yarn.sh
stopping yarn daemons
no resourcemanager to stop
localhost: stopping nodemanager
localhost: nodemanager did not stop gracefully after 5 seconds: killing with kill
1 -9
no proxyserver to stop
ubuntu@ip-172-31-16-229:~$ jps
21350 Jps
19561 SecondaryNameNode
19388 DataNode
19247 NameNode
ubuntu@ip-172-31-16-229:~$ stop-dfs.sh
Stopping namenodes on [ip-172-31-16-229.us-west-2.compute.internal]
ip-172-31-16-229.us-west-2.compute.internal: stopping namenode
localhost: stopping datanode
Stopping secondary namenodes [0.0.0.0]
0.0.0.0: stopping secondarynamenode
ubuntu@ip-172-31-16-229:~$ jps
21754 Jps
ubuntu@ip-172-31-16-229:~$
```

### 3. Create instance image

$\pi$

# Stop instance:



# Create image:

π

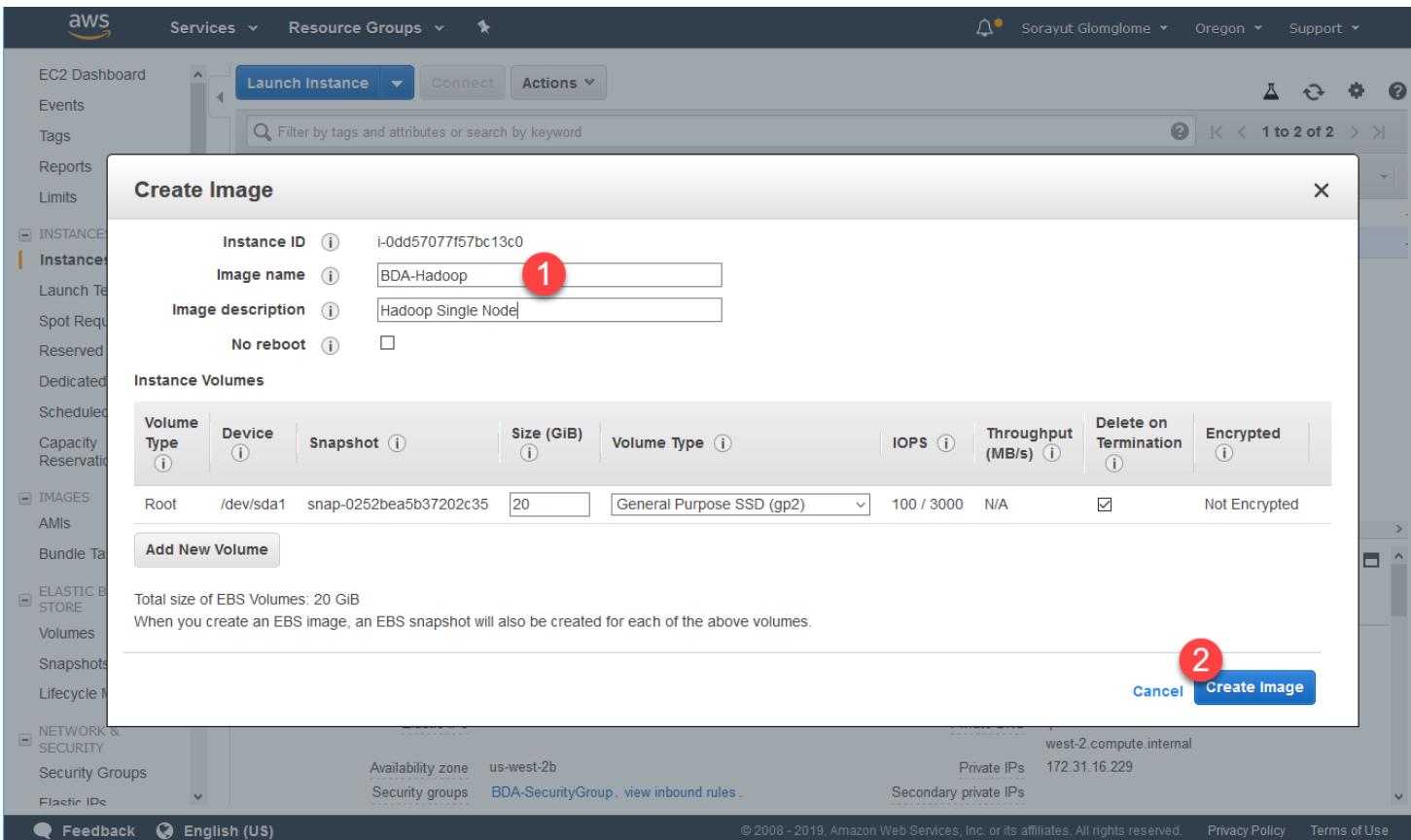
The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with options like EC2 Dashboard, Events, Tags, Reports, Limits, Instances (selected), Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, AMIs, and Elastic Block Store. The main area displays a table of instances. One instance, "BDA-Hadoop...", is selected. A context menu is open over this instance, with the "Image" option highlighted. The menu also includes "Connect", "Get Windows Password", "Create Template From Instance", "Launch More Like This", "Instance State", "Instance Settings", "Networking", and "CloudWatch Monitoring". At the bottom of the instance details, there's a table with fields like Instance ID, Instance state, Instance type, Availability zone, Security groups, Public DNS (IPv4), IPv4 Public IP, IPv6 IPs, Private DNS, Private IPs, and Secondary private IPs.

| Name                | Instance ID         | Instance Type | Availability Zone | Instance State | Status Checks | Alarm Status | Public DNS (IPv4) |
|---------------------|---------------------|---------------|-------------------|----------------|---------------|--------------|-------------------|
| i-0d35056ebb33ba5e2 | t2.large            | us-west-2a    | stopped           |                |               |              |                   |
| BDA-Hadoop...       | i-0dd57077f57bc13c0 | m5.large      | us-west-2b        | stopped        |               |              |                   |

| Description   | Status Checks  | Monitoring | Tags |
|---|--|------------|------|
| Instance ID: i-0dd57077f57bc13c0 (BDA-Hadoop Server)    | Private IP: 172.31.16.229                                |            |      |
| Instance ID: i-0dd57077f57bc13c0                        | Public DNS (IPv4): -                                     |            |      |
| Instance state: stopped                                 | IPv4 Public IP: -  |            |      |
| Instance type: m5.large                                 | IPv6 IPs: -  |            |      |
| Elastic IPs:  | Private DNS: ip-172-31-16-229.us-west-2.compute.internal |            |      |
| Availability zone: us-west-2b                           | Private IPs: 172.31.16.229                               |            |      |
| Security groups: BDA-SecurityGroup, view inbound rules. | Secondary private IPs:                                   |            |      |

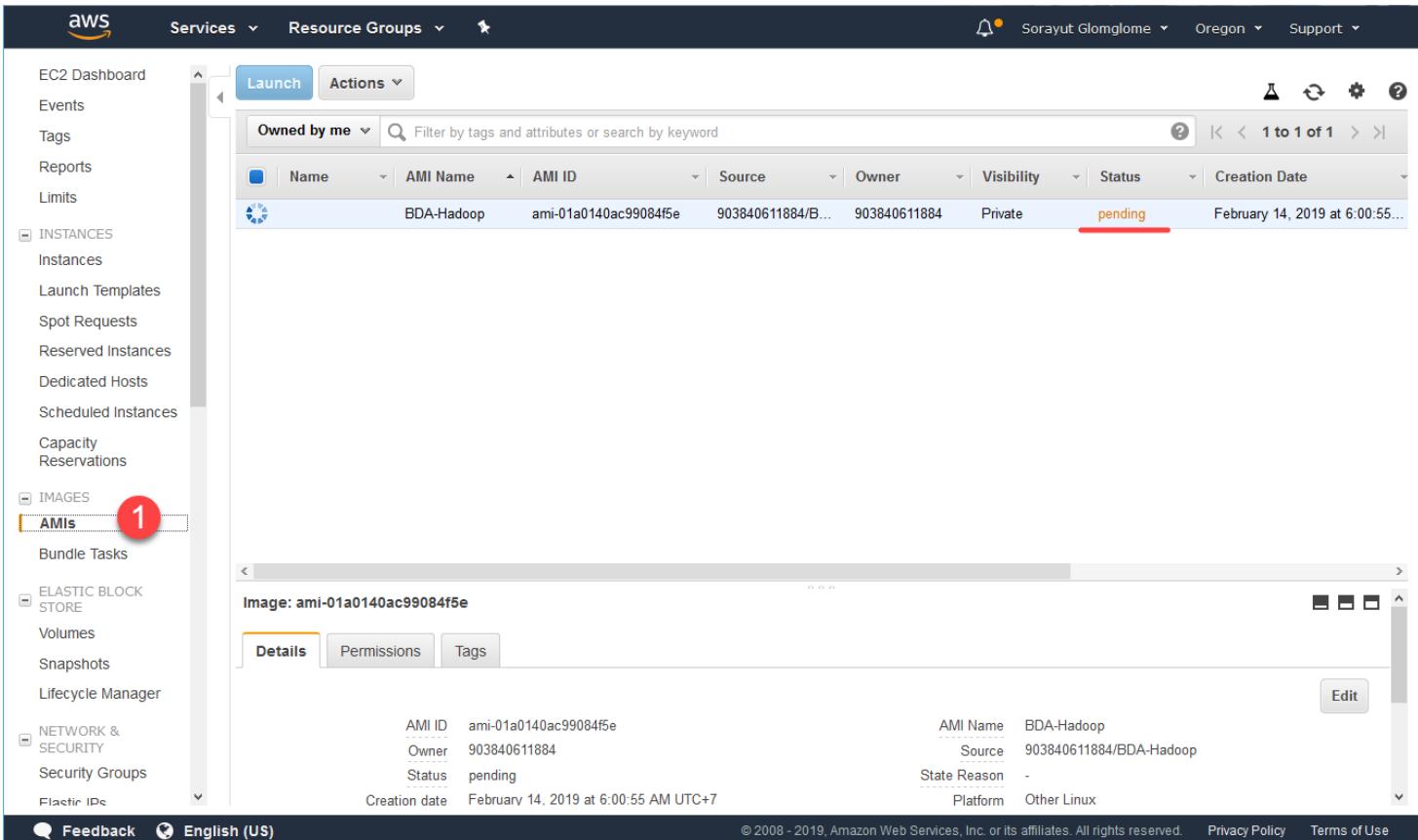
# Create image:

π



# Waiting for creation:

π



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