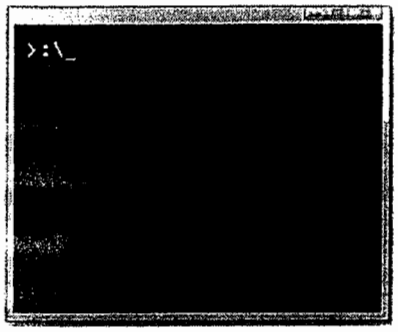
Lab 2

FIRST STEPS INTO THE LINUX CONSOLE

STEP 1: Introduction to the Linux Command Line Interface (CLI)



GNU/Linux is a modern Operating System and it's widely used on servers. Server-like Linux distributions are usually setup without any Graphical User Interface (GUI) for saving computational power and because you can do almost anything using the command-line interface (CLI). In the old days, CLI was the only user interface available on a Unix-like system such as Linux.

The **command-line interface** is a tool into which you can type text commands to perform specific tasks. Since you can directly control the computer by typing, many tasks can be performed more quickly, and some**tasks can be automated** with special commands that loop through and perform the same action on many files—saving you a lot of time.

The application or user interface that accepts your typed responses and displays the data on the screen is called a **shell**, and there are many different varieties that you can choose from (ksh, tcsh, zsh...), but the most common these days is the **Bash** shell, which is the default on Linux and Mac systems. Bash stands for Bourne Again SHell, and it is an enhanced version of the original Unix shell program, **sh**, written by Steve Bourne.

You can use a local or remote shell using a a program called a **terminal emulator**. This is a program that opens a window and lets you interact with the shell.

There are a bunch of different terminal emulators you can use:

PuTTY is a terminal client suitable for Windows users.

Terminal is embedded in all Mac OS versions.

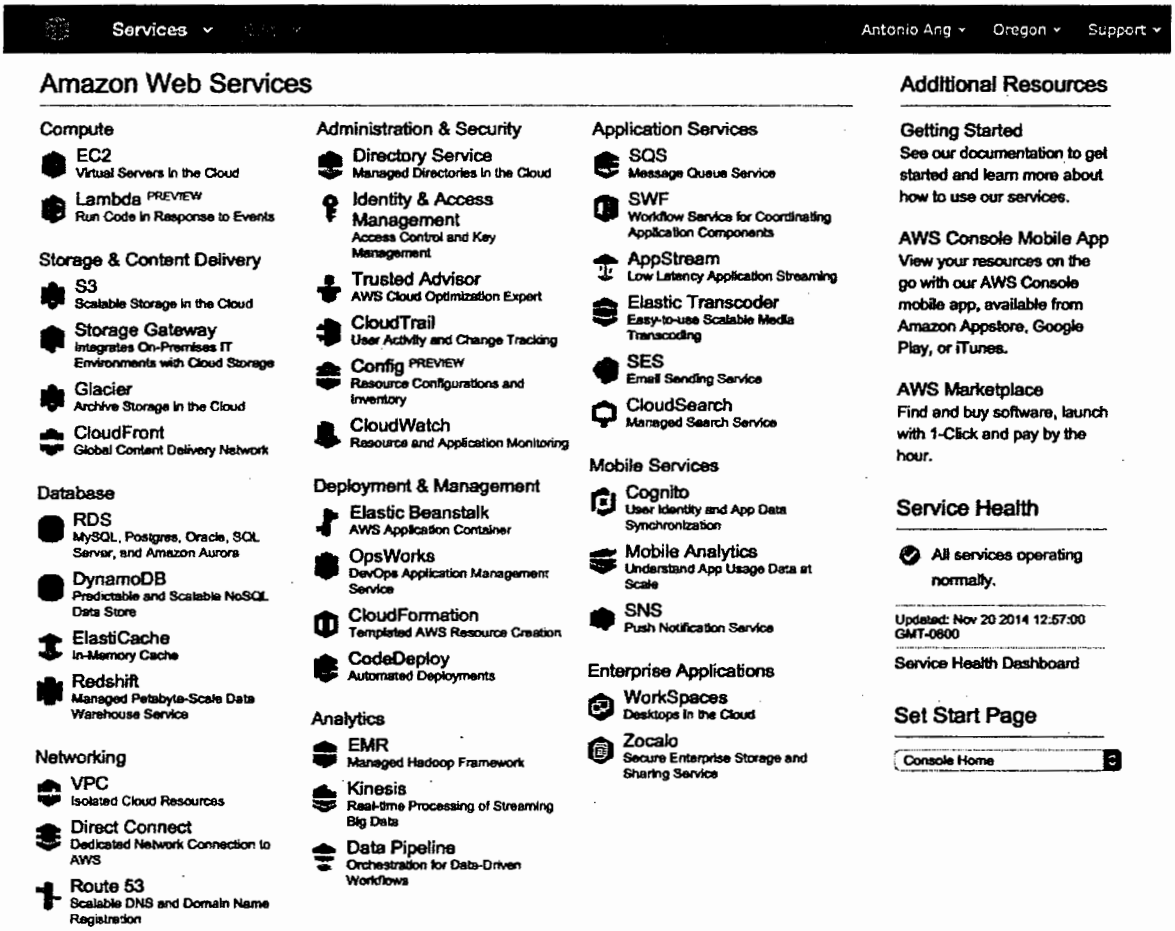
iTerm is en enhanced terminal application for MacOS.

The next lab steps will show you how to connect to a remote EC2 instance using the SSH protocol and get familiar with the basic shell commands.

Go to the next step and start by creating a basic EC2 Linux instance.

STEP 2: Log In to the Amazon Web Service Console

This laboratory experience is about Amazon Web Services and you will use the AWS Management Console in order to complete all the lab steps.



The AWS Management Console is a web control panel for managing all your AWS resources, from EC2 instances to SNS topics. The console enables cloud management for all aspects of the AWS account, including managing security credentials, or even setting up new IAM Users.

Log in to the AWS Management Console

In order to start the laboratory experience, open the Amazon Console by clicking this button:

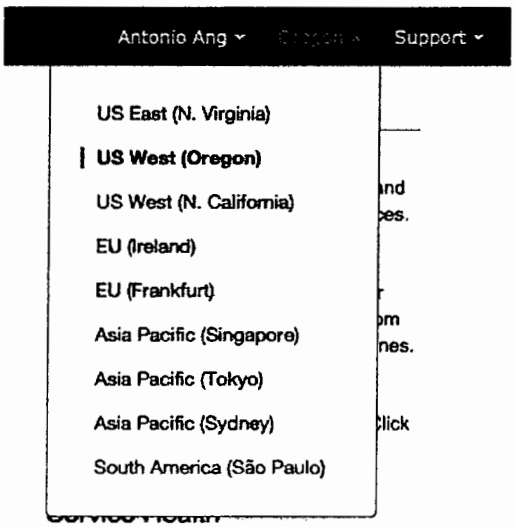
Open AWS Console

Log in with the username **xxxx** and the password **xxxx**.

Select the right AWS Region

Amazon Web Services is available in different regions all over the world, and the console lets you provision resources across multiple regions. You usually choose a region that best suits your business needs to optimize your customer's experience, but you must use the region **US West (Oregon)** for this laboratory.

You can select the **US West (Oregon)** region using the upper right dropdown menu on the AWS Console page.

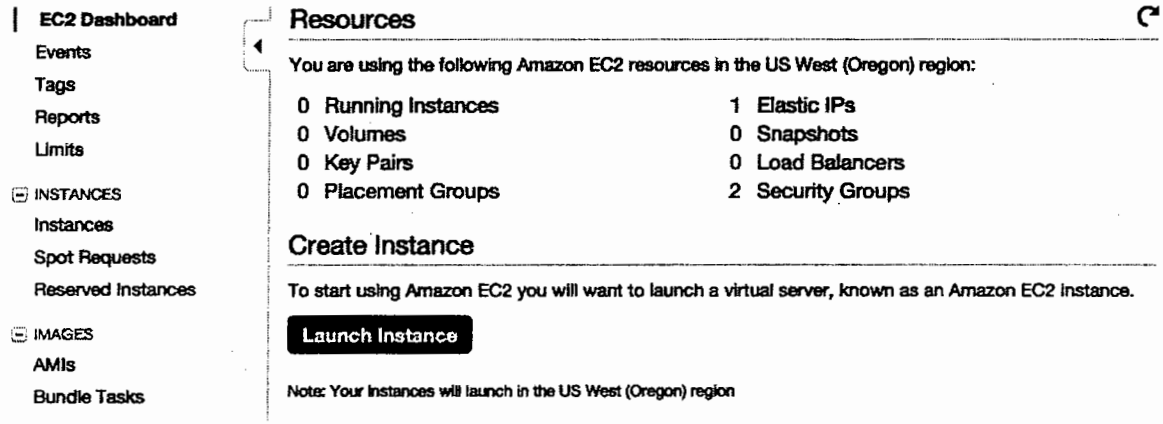


STEP 3: Create an EC2 instance

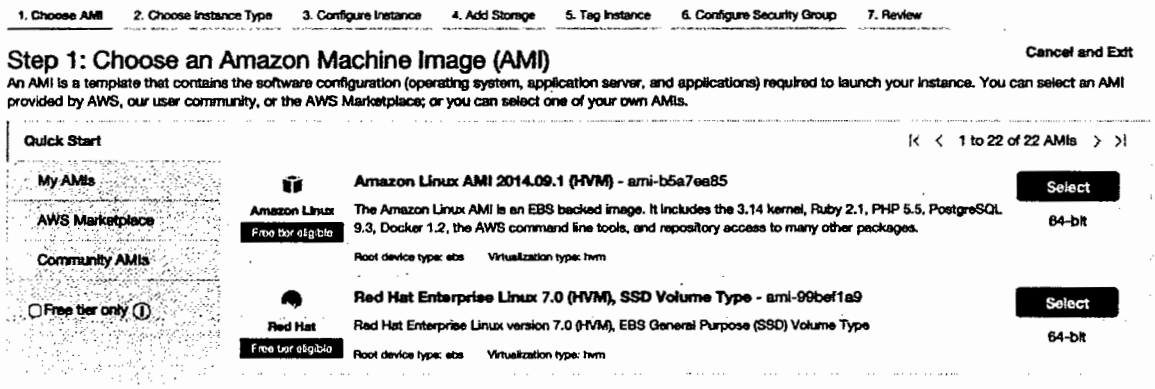
You can launch an EC2 instance using the EC2 launch wizard.

Select the EC2 service from the Management Console dashboard:

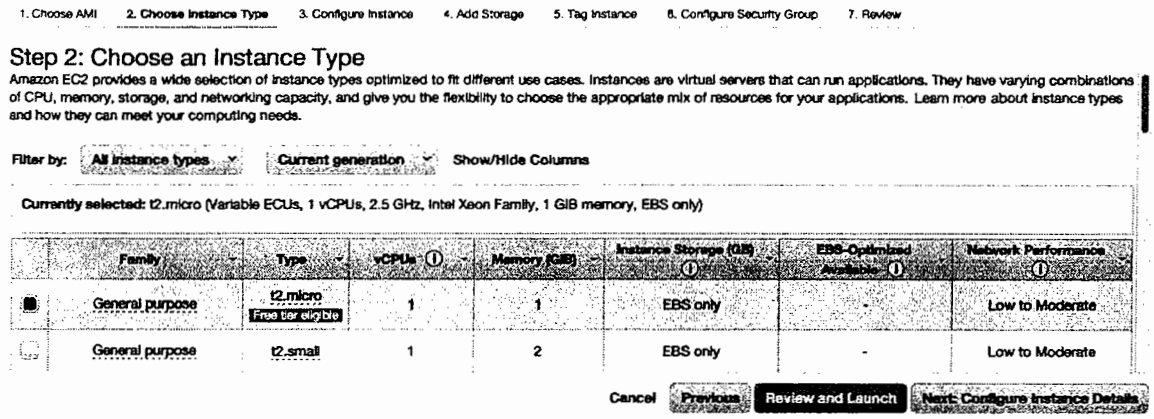
From the EC2 dashboard, click **Launch Instance**.



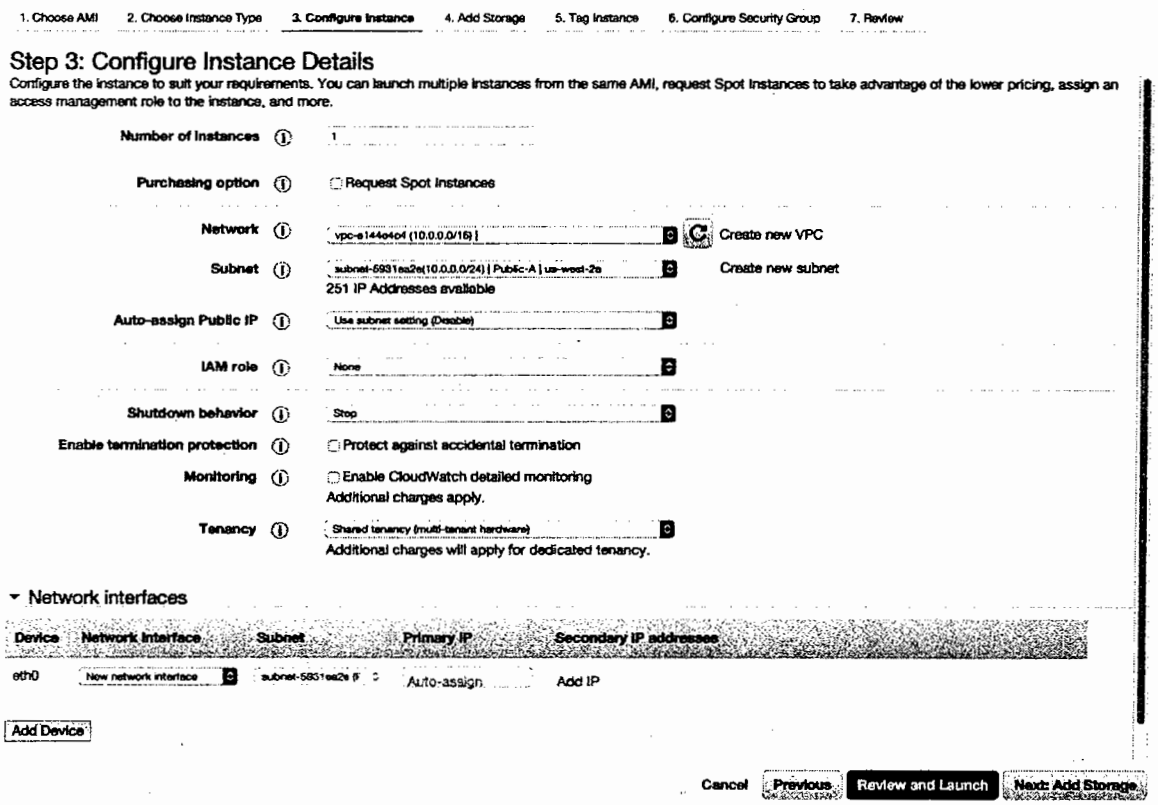
The **Choose an Amazon Machine Image (AMI)** page displays a list of basic configurations called **Amazon Machine Images (AMIs)** that serve as templates for your instance. Select the first listed 64-bit **Amazon Linux AMI**.



On the **Choose an Instance Type** page, do **not** change any options and click **Next: Configure Instance Details.**



On the **Configure Instance Details** tab, check the selected **Network (VPC)** and **Subnet**. Change them, if needed, and then click **Next : Add Storage**.



On the **Add Storage** tab, do **not** change any options, and click the **Review and Launch** button.

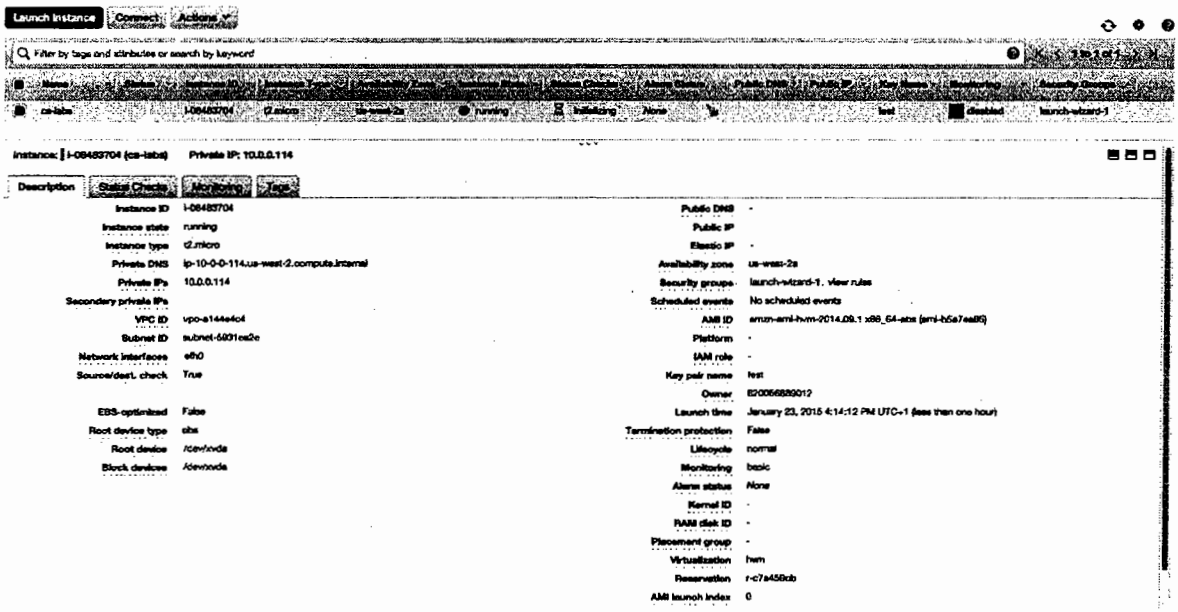
On the Review Instance Launch page, click **Launch**.

In the **Select an existing key pair or create a new key pair** dialog box, select **Create a new key pair**, then type a KeyPair name (e.g., "TestKeys") and download it.

Select the acknowledgment checkbox, and then click **Launch Instances**.

A confirmation page will let you know that your instance is launching. Click **View Instances** to close the confirmation page and return to the console.

On the Instances Screen, you can view the status of your instance. It will take a short time for your instance to be launched. When you launch an instance, its initial state defaults to *pending*. After the instance starts, its Instance State changes to *running*, and it receives a public DNS name.



STEP 4: Convert a PEM key to a PPK key

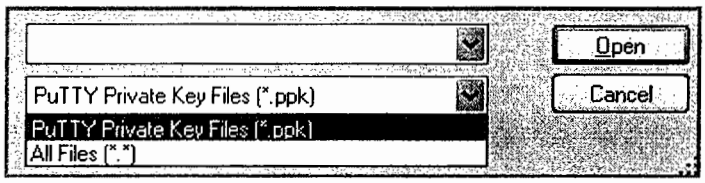
If you are a Windows user, you will probably use **PuTTY** for connecting to the remote instance. PuTTY is a great SSH client, but it does not natively support the PEM key format. Fortunately, PuTTY has a tool called **PuTTYgen**, which can convert keys to the required PPK format.

Converting a PEM key is easy and fast:

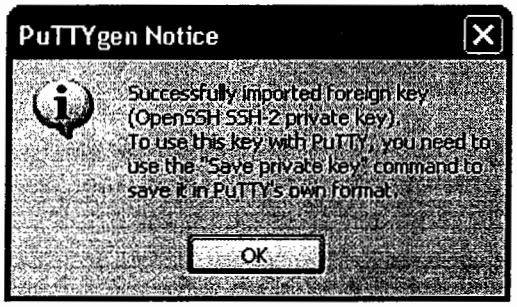
If you do not already have it, download the PuTTYgen executable from its main website: **PuTTYgen**

Start PuTTYgen (no installation required).

Click **Load** and browse to the location of the private key file that you want to convert (e.g. ec2key.pem). By default, PuTTYgen displays only files with extension .ppk. You'll need to change that default to display files of all types in order to see your .pem key file.zy



Select your .pem key file and click **Open**. PuTTYgen displays the following message.



When you click **OK**, PuTTYgen displays a dialog box with information about the key you loaded, such as the public key and the fingerprint.

Click **Save private key** to save the key in PuTTY's format.

Do NOT select a passphrase and save your private key somewhere secure.

Now you are ready to use PuTTY for connecting to the previously created instance!

STEP 5: Connect to a remote shell using an SSH connection

In order to manage a remote Linux server, you must employ an **SSH Client**. Secure Shell (SSH) is a cryptographic network protocol for securing data communication. It establishes a secure channel over an insecure network. Common applications include remote command-line login and remote command execution.

Connect using Linux / Mac OS

Linux distributions and Mac OS are shipped with a fully working SSH client that accepts standard PEM Keys.

Starting a remote SSH session is easy:

Open your **Terminal** application

Write and run the following command: **ssh -i /path/to/your/keypair.pem user@server-ip**.

server-ip is the Public IP of your server, you can find it in the EC2 instance details

user is the remote system user that will be used for the remote authentication

Amazon Linux AMIs typically use **ec2-user** as username.

Ubuntu AMIs login user is **ubuntu**, Debian AMIs use **admin** instead.

Assuming that you selected the Amazon Linux AMI, your assigned public IP is 123.123.123.123, and your keypair (named "keypair.pem") is stored in /home/youruser/keypair.pem, the right command to run is: **ssh -i /home/youruser/keypair.pem ec2-user@123.123.123.123**

Note: your SSH Client may refuse to start the connection, warning that the key file is unprotected. You should deny the file access to any other system user by changing its permissions. Issue the following command and then try again:

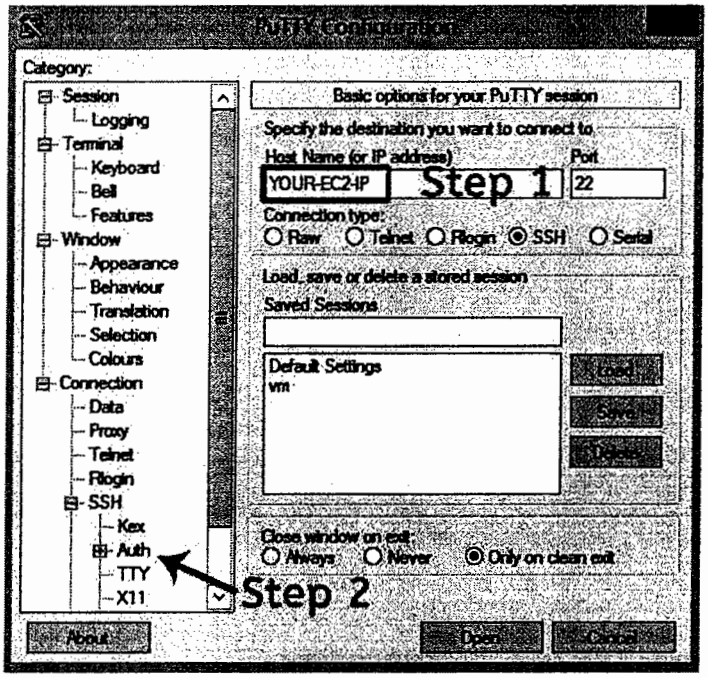
chmod 600 /home/youruser/keypair.pem

Connect using Windows

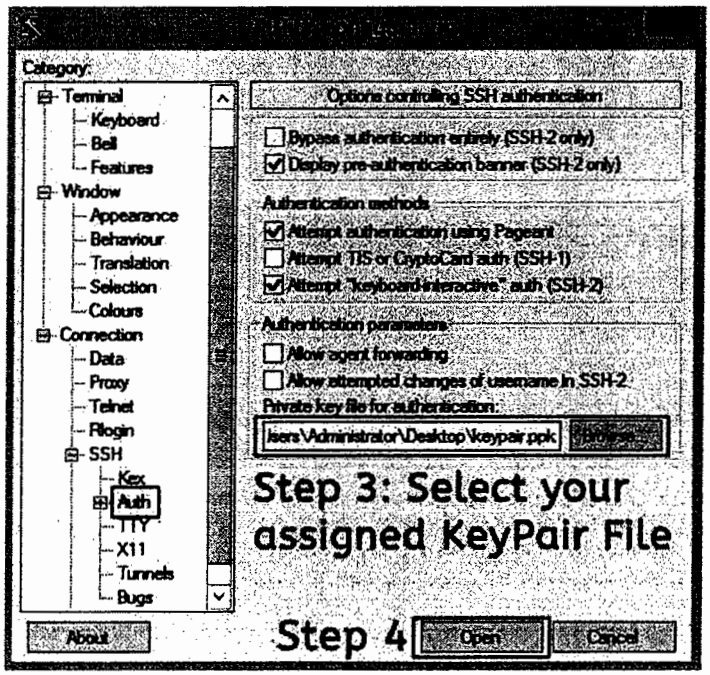
Windows has no SSH client, so you must use PuTTY and convert the PEM key to PPK using PuTTYgen.

Starting a remote SSH session using PuTTY is easy:

Open PuTTY and insert the EC2 instance IP Address in the Host Name field.



Select **Connection > SSH > Auth** section and then select the downloaded Keypair that you previously converted to PPK format.



After some seconds, you will see the authentication form. **Login as ec2-user** and you will see the EC2 server welcome banner.

STEP 6: Move between directories and list file content

Now you are ready to send the first commands to your Linux instance. Let's start by using some basic commands for browsing the file-system.

The **pwd** command will allow you to know in which directory you're located (**pwd** stands for "print working directory").

The **ls** command will show you the files in your current directory. Used with certain options (ls -larth), you can see sizes of files in a human readable version, when files were made, and permissions of files.

The **cd** command will allow you to change directories. When you open a terminal you will be in your home directory. To move around the file system you will use **cd**.

Examples:

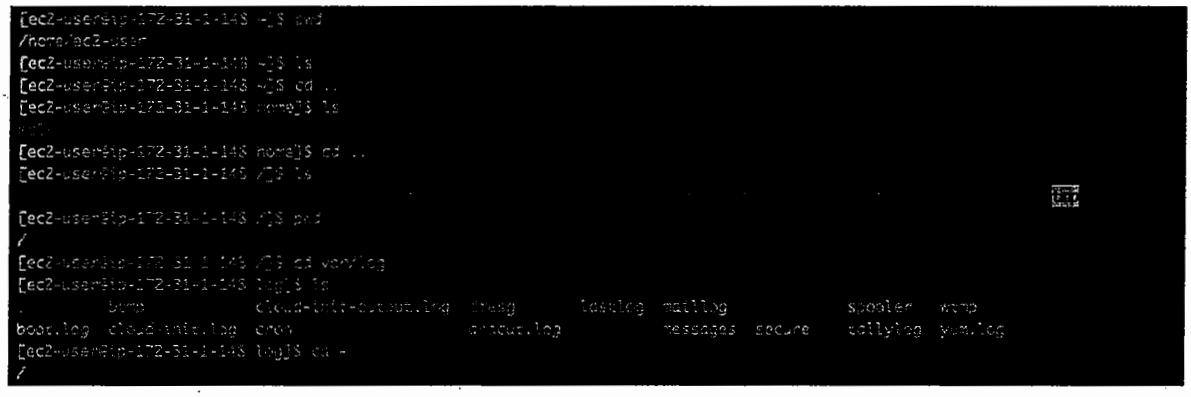
To navigate into the root directory, use **"cd /"**

To navigate to your home directory, use **"cd"** or **"cd ~"**. (The tilde (~) symbol stands for your home directory).

To navigate up one directory level, use **"cd .."**

To navigate to the previous directory (or back), use **"cd -"**

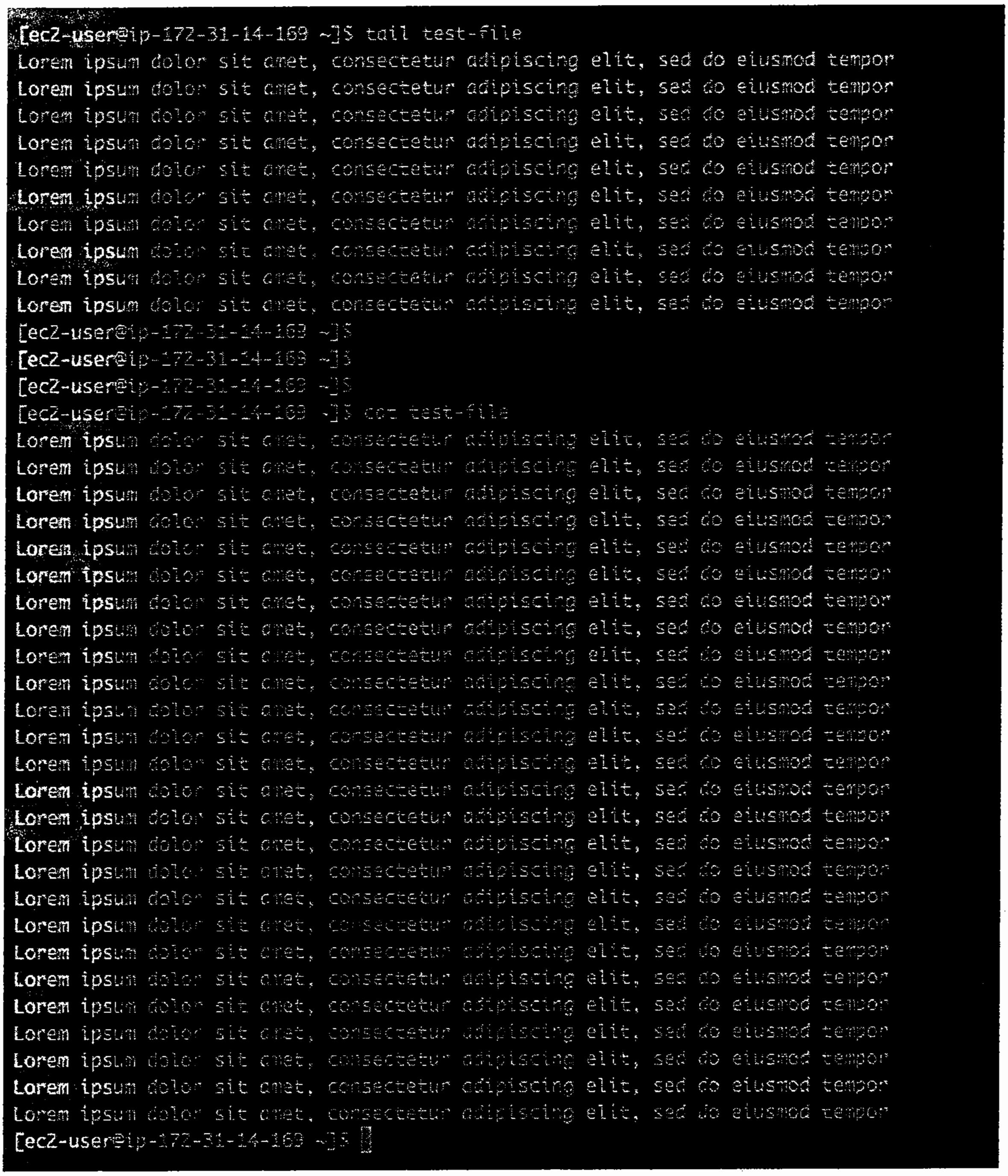
To navigate through multiple levels of directory at once, specify the full directory path that you want to go to. For example, use, **"cd /var/www"** to go directly to the /www subdirectory of /var/.



The **cat** command will output the contents of a specific file and can be used to concatenate and list files. The name is an abbreviation of *catenate*, a synonym of concatenate.

The **tail** command prints the last 10 lines of each file to standard output.

Here you can see a demo file composed by 20+ lines printed using tail and cat.



```
[ec2-user@ip-172-31-14-169 ~]$ tail test-file
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
[ec2-user@ip-172-31-14-169 ~]$
[ec2-user@ip-172-31-14-169 ~]$
[ec2-user@ip-172-31-14-169 ~]$
[ec2-user@ip-172-31-14-169 ~]$ cat test-file
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
[ec2-user@ip-172-31-14-169 ~]$
```

STEP 7: Manage files and their permissions

Here there are a series of useful command for copying, moving, renaming and removing files and directories.

The **cp** command will make a copy of a file for you.

Example: **"cp file foo"** will make an exact copy of "file" and name it "foo", but the file "file" will still be there.

If you are copying a directory, you must use **"cp -r directory foo"** (copy recursively).

The **mv** command will move a file to a different location or will rename a file. Examples:

"mv file foo" will rename the file "file" to "foo".

"mv foo /tmp" will move the file "foo" to the root temp directory, but it will not rename it. You must specify a new file name to rename a file.

The **rm** command will remove or delete a file in your directory.

The **rmdir** command will delete an empty directory. To delete a directory and all of its contents recursively, use **rm -r** instead.

The **mkdir** command will allow you to create directories. Example: **"mkdir music"** will create a directory called "music".

```
[ec2-user@ip-172-31-14-169 ~]$ cp test-file copied-file
[ec2-user@ip-172-31-14-169 ~]$ ls
copied-file  test-file
[ec2-user@ip-172-31-14-169 ~]$ mkdir new-dir
[ec2-user@ip-172-31-14-169 ~]$ ls
copied-file           test-file
[ec2-user@ip-172-31-14-169 ~]$ mv copied-file new-dir/
[ec2-user@ip-172-31-14-169 ~]$ ls new-dir/
copied-file
[ec2-user@ip-172-31-14-169 ~]$ ls
          test-file
[ec2-user@ip-172-31-14-169 ~]$ rm new-dir/
rm: impossibile rimuovere "new-dir/": È una directory
[ec2-user@ip-172-31-14-169 ~]$ rm -r new-dir/
[ec2-user@ip-172-31-14-169 ~]$ ls
test-file
[ec2-user@ip-172-31-14-169 ~]$
```

Permission Management

*nix like Operating Systems (GNU/Linux, *BSD, MacOS...) manage resource access control using users and groups. Each file and directory has an ACL set to control who can read, write, and execute itself.

Two very important commands, **chmod** and **chown**, deal with permissions and ownership (respectively).

The chmod command allows you to change permissions on a file. The basic usage is: **chmod PERMISSIONS FILE**

Where **PERMISSIONS** is either the numeric or the alpha equivalent of the permissions you want to assign and **FILE** is the list of file (or folder) you want to effect.

File permissions are in the form: OWNER | GROUP | OTHERS

Each of those sections includes: READ | WRITE | EXECUTE

Each permission (read, write, execute) is represented with the binary representation of the initial letter:

r - 4

w - 2

x - 1

To get the numeric permission you add which permissions you want to use together. It you want:

r+w you get 6.

r+w+x you get 7.

r+x you get 5.

r you get 4.

You need to set the permissions for three different "users" (Owner, Group, All Others). If you want Owner and Group to have full permissions (read, write and execute) and All Others to only have r permissions, you should use "774" as permission mask.

To change the permission of a particular file to 774 you should issue the command: **chmod 774 FILENAME**

```
SUNSET:test antonioangelino$ chmod 600 test.pem
SUNSET:test antonioangelino$ ls -l
total 8
-rw-------@ 1 antonioangelino  staff  1692 23 Gen 16:14 test.pem
SUNSET:test antonioangelino$ chmod 660 test.pem
SUNSET:test antonioangelino$ ls -l
total 8
-rw-rw----@ 1 antonioangelino  staff  1692 23 Gen 16:14 test.pem
SUNSET:test antonioangelino$ chmod 747 test.pem
SUNSET:test antonioangelino$ ls -l
total 8
-rwxr--rwx@ 1 antonioangelino  staff  1692 23 Gen 16:14
SUNSET:test antonioangelino$
```

If you want to change the ownership of a file or folder you need the chown command. Its basic usage is: **chown USER.GROUP FILE**

Where USER and GROUP and the new user and group that you want to assing to FILE. If you want to leave the FILE belonging to the original group, you can use chown without specify the Owner GROUP.

STEP 8: Monitor and manage processes

There are some useful system commands that can help you to understand what is going on in your instance.

The **free** command displays the amount of free and used memory in the system. **"free -m"** will give the information using megabytes, which is probably most useful nowadays.

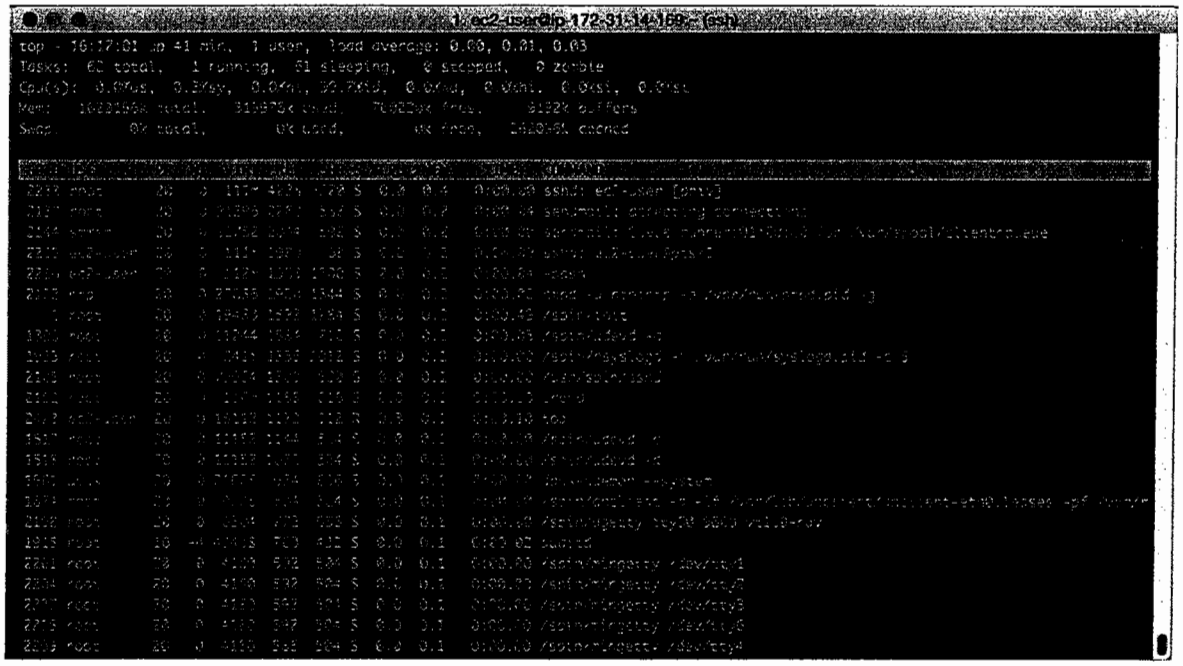
```
[ec2-user@ip-172-31-14-169 ~]$ free
             total       used       free     shared    buffers     cached
Mem:       1020196     319880     700316          0       9188     262016
-/+ buffers/cache:      48676     971520
Swap:            0          0          0
[ec2-user@ip-172-31-14-169 ~]$ free -m
             total       used       free     shared    buffers     cached
Mem:           996        312        683          0          8        255
-/+ buffers/cache:         47        948
Swap:            0          0          0
```

The **top** ('table of processes') command displays information on your Linux system, running processes and system resources, including CPU, RAM & swap usage and total number of tasks being run. To exit **top**, press **"q"**.

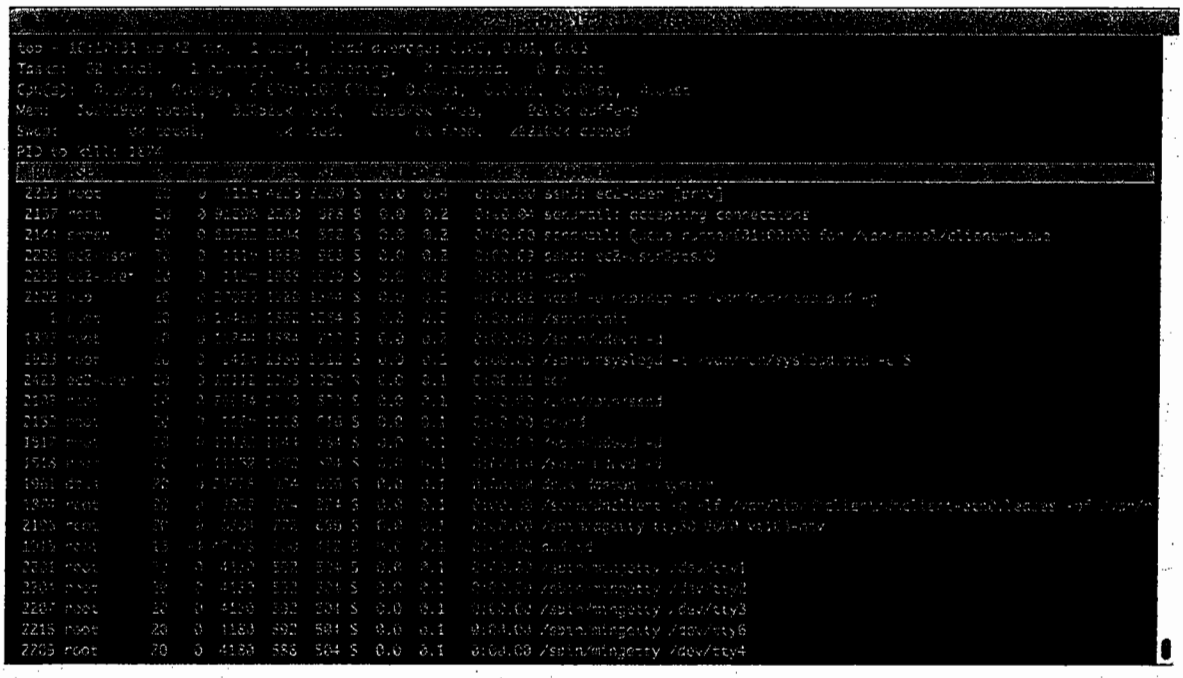
```
1. ec2-user@ip-172-31-16-169:~ (ssh)
top - 16:16:17 up 41 min,  1 user,  load average: 0.00, 0.01, 0.03
Tasks:  62 total,   1 running,  61 sleeping,   0 stopped,   0 zombie
Cpu(s):  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   1020196k total,   310068k used,   710128k free,     9132k buffers
Swap:        0k total,        0k used,        0k free,   200016k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
    1 root      20   0 10480 1592 1284 S  0.0  0.2   0:00.45 init
    2 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kthreadd
    3 root      20   0     0    0    0 S  0.0  0.0   0:00.00 ksoftirqd/0
    5 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kworker/0:0H
    6 root      20   0     0    0    0 S  0.0  0.0   0:00.04 kworker/u30:0
    7 root      20   0     0    0    0 S  0.0  0.0   0:00.51 rcu_preed
    8 root      20   0     0    0    0 S  0.0  0.0   0:00.00 rcu_bh
    9 root      RT   0     0    0    0 S  0.0  0.0   0:00.00 migration/0
   10 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 khelper
   11 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kdevtmpfs
   12 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 netns
   13 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kworker/u30:1
   19 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kswapd0
   20 root      20   0     0    0    0 S  0.0  0.0   0:00.01 ksmd
   21 root      10   0     0    0    0 S  0.0  0.0   0:00.10 khugepaged
  115 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 writeback
  118 root      25   5     0    0    0 S  0.0  0.0   0:00.00 ksmd
  119 root      39  19     0    0    0 S  0.0  0.0   0:00.00 khugepaged
  120 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kintegrityd
  121 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 bioset
  122 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 crypto
  124 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kblockd
  473 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 md
  598 root      20   0     0    0    0 S  0.0  0.0   0:00.00 khungtaskd
  603 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kswapd0
  671 root      20   0     0    0    0 S  0.0  0.0   0:00.00 fsnotify_mark
  685 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 ktnrotld
  692 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kvved
  740 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 deferwq
 1360 root      20   0     0    0    0 S  0.0  0.0   0:00.00 jbd2/xvda1-8
 1361 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 ext4-rsv-conver
 1398 root      20   0 11244 1304  712 S  0.0  0.2   0:00.08 udevd
 1517 root      20   0 11152 1144  384 S  0.0  0.1   0:00.00 udevd
 1518 root      20   0 11152 1072  324 S  0.0  0.1   0:00.00 udevd
 1531 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kpsmoused
 1549 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kworker/0:2
 1703 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kauditd
 1746 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 ipv6_addrconf
 1874 root      20   0  9128  804  704 S  0.0  0.1   0:00.00 dhclient
 1915 root      18  -4 15108  700  332 S  0.0  0.1   0:00.02 auditd
 1933 root      20   0  2417 1336 1012 S  0.0  0.1   0:00.00 rsyslogd
 1949 root      20   0  4180  364  324 S  0.0  0.0   0:00.01 rngd
 1961 dbus      20   0 21616  924  696 S  0.0  0.1   0:00.10 dbus-daemon
 2085 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kworker/0:1H
 2105 root      20   0 78904 1700  520 S  0.0  0.1   0:00.00 sshd
 2122 ntp       20   0 32956 1620 1344 S  0.0  0.2   0:00.02 ntpd
 2137 root      20   0 91296 2280  588 S  0.0  0.2   0:00.04 sendmail
 2144 smmsp     20   0 83752 2044  532 S  0.0  0.2   0:00.00 sendmail
 2152 root      20   0  1184 1168  316 S  0.0  0.1   0:00.00 crond
 2162 root      20   0 10908  372  220 S  0.0  0.0   0:00.00 atd
 2198 root      20   0  5304  772  656 S  0.0  0.1   0:00.00 agetty
 2201 root      20   0  4180  592  504 S  0.0  0.1   0:00.00 mingetty
 2204 root      20   0  4180  592  504 S  0.0  0.1   0:00.00 mingetty
```

If you want to order the processes per used memory, just press "M".



top also allows you to kill a process using its PID (Process ID). Press "k", insert the PID and then press "ENTER" twice.



The **df** command reports the amount of available disk space being used by file systems. Using "**df -h**" you can see all values converted in GB.

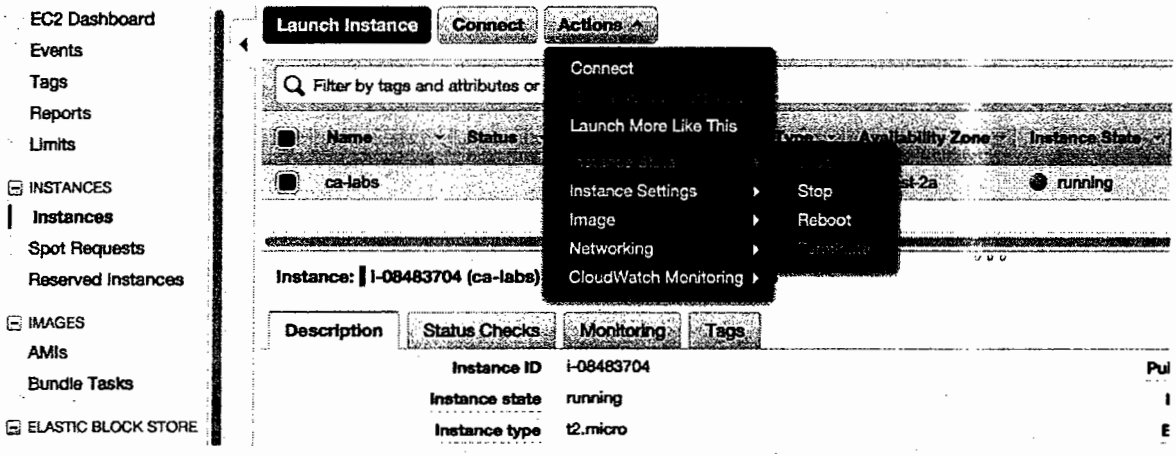
```
[ec2-user@ip-172-31-14-169 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda1      7.8G  1.1G  6.7G  14% /
devtmpfs        491M   56K  491M   1% /dev
tmpfs           499M     0  499M   0% /dev/shm
```

STEP 9: Terminate an EC2 instance

When you've decided that you no longer need an instance, you can terminate it.

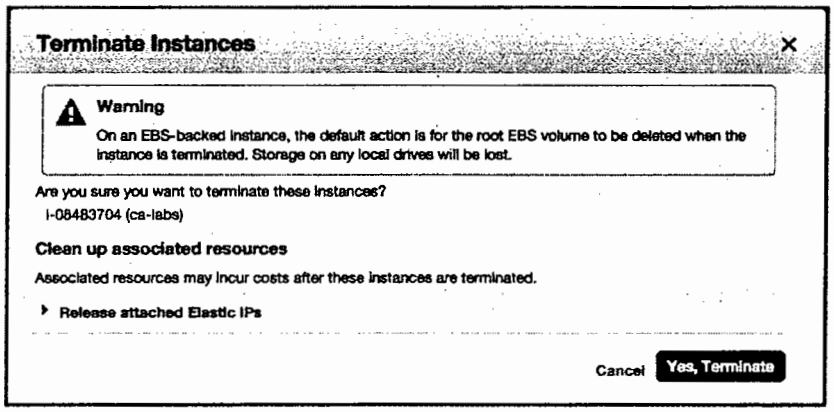
Select the EC2 service from the Management Console dashboard:

In the navigation pane, click **Instances**.



Select the instance linuxshell, click **Actions**, select **Instance State**, and then click **Terminate**.

Click **Yes, Terminate** when prompted for confirmation.



Now your instance is completely destroyed.