

# Homework 1

Daniel Tan

## Basic Commands

## Print and Change Directories

The `pwd` command prints the current location of the user and the `cd` command allows you to change your location to a desired directory

```
1 [ec2-user@ip-172-31-16-171 ~]$ pwd  
2 /home/ec2-user
```

Figure 1: `pwd` Command

```
3 [ec2-user@ip-172-31-16-171 ~]$ cd /var/log  
4 [ec2-user@ip-172-31-16-171 log]$ cd ..  
5 [ec2-user@ip-172-31-16-171 var]$ pwd  
6 /var
```

Figure 2: `cd` Command

# Listing Items

The `ls` command lists the items in the current directory. By default, it only prints the names of the items but can be configured by adding flags

```
[ec2-user@ip-172-31-16-171 var]$ ls -a
. . .
[ec2-user@ip-172-31-16-171 var]$ ls -l
total 4
11 drwxr-xr-x 2 root root 19 Apr 28 19:54 account
12 drwxr-xr-x 2 root root 6 Apr 9 2019 adm
13 drwxr-xr-x 6 root root 63 Apr 28 19:54 cache
14 drwxr-xr-x 3 root root 18 Apr 28 19:54 db
15 drwxr-xr-x 3 root root 18 Apr 28 19:54 empty
16 drwxr-xr-x 2 root root 6 Apr 9 2019 games
17 drwxr-xr-x 2 root root 6 Apr 9 2019 gopher
18 drwxr-xr-x 3 root root 18 Apr 28 19:54 kerberos
19 drwxr-xr-x 30 root root 4096 Apr 28 19:55 lib
20 drwxr-xr-x 2 root root 6 Apr 9 2019 local
21 lrwxrwxrwx 1 root root 11 Apr 28 19:55 lock -> ../run/lock
22 drwxr-xr-x 7 root root 320 May 9 11:59 log
23 lrwxrwxrwx 1 root root 10 Apr 28 19:53 mail -> spool/mail
24 drwxr-xr-x 2 root root 6 Apr 9 2019 nis
25 drwxr-xr-x 2 root root 6 Apr 9 2019 opt
26 drwxr-xr-x 2 root root 6 Apr 9 2019 preserve
27 lrwxrwxrwx 1 root root 6 Apr 28 19:53 run -> ../run
28 drwxr-xr-x 9 root root 97 Apr 28 19:54 spool
29 drwxr-xr-x 3 root root 85 May 9 11:59 tmp
30 drwxr-xr-x 2 root root 6 Apr 9 2019 yp
```

Figure 3: Basic `ls` command and longlisting format

# Listing Items

The `ls` command can be used to sort items by time, size and other attributes, in both ascending and descending order

```
1 [ec2-user@ip-172-31-16-171 var]$ ls -lrt
2 total 4
3 drwxr-xr-x 2 root root 6 Apr  9 2019 yp
4 drwxr-xr-x 2 root root 6 Apr  9 2019 preserve
5 drwxr-xr-x 2 root root 6 Apr  9 2019 opt
6 drwxr-xr-x 2 root root 6 Apr  9 2019 nis
7 drwxr-xr-x 2 root root 6 Apr  9 2019 local
8 drwxr-xr-x 2 root root 6 Apr  9 2019 gopher
9 drwxr-xr-x 2 root root 6 Apr  9 2019 games
10 drwxr-xr-x 2 root root 6 Apr  9 2019 adm
11 lrwxrwxrwx 1 root root 6 Apr 28 19:53 run -> ../run
12 lrwxrwxrwx 1 root root 11 Apr 28 19:53 lock -> ../run/lock
13 lrwxrwxrwx 1 root root 10 Apr 28 19:53 mail -> spool/mail
14 drwxr-xr-x 3 root root 18 Apr 28 19:54 kerberos
15 drwxr-xr-x 3 root root 18 Apr 28 19:54 db
16 drwxr-xr-x 3 root root 18 Apr 28 19:54 empty
17 drwxr-xr-x 2 root root 19 Apr 28 19:54 account
18 drwxr-xr-x 9 root root 97 Apr 28 19:54 spool
19 drwxr-xr-x 6 root root 63 Apr 28 19:54 cache
20 drwxr-xr-x 30 root root 4096 Apr 28 19:55 lib
21 drwxr-xr-x 7 root root 320 May  9 11:59 log
22 drwxrwxrwt 3 root root 85 May  9 11:59 tmp
```

Figure 4: `ls` longlisting sorted by time in reverse order

## Absolute and Relative Paths

By default, / by itself is the root directory and ~ is the current user's directory. These are absolute paths within Linux.

```
23 [ec2-user@ip-172-31-16-171 var]$ cd ~  
24 [ec2-user@ip-172-31-16-171 ~]$ cd /home  
25 [ec2-user@ip-172-31-16-171 home]$ cd ..
```

Figure 5: Absolute and relative paths

Meanwhile .. goes back one directory from your current location, and is a relative path.

# Man Pages (Help Pages)

If ever in doubt of a command, calling `man` on the command usually brings up a help page.

```
18 [ec2-user@ip-172-31-16-171 /]$ man ls                               User Commands                                         LS(1)
19
20 NAME
21   ls - list directory contents
22
23 SYNOPSIS
24   ls [OPTION]... [FILE]...
25
26 DESCRIPTION
27   List information about the FILEs (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.
28
29   Mandatory arguments to long options are mandatory for short options too.
30
31   -a, --all
32     do not ignore entries starting with .
33
34   -A, --almost-all
35     do not list implied . and ..
36
37   --author
38     with -l, print the author of each file
39
40   -b, --escape
41     print C-style escapes for nongraphic characters
42
43   --block-size=SIZE
44     scale sizes by SIZE before printing them; e.g., '--block-size=M' prints sizes in units of 1,048,576 bytes; see SIZE format below
45
46   -B, --ignore-backups
47     do not list implied entries ending with ~
48
49   -c      with -lt: sort by, and show, ctime (time of last modification of file status information); with -l: show ctime and sort by name; otherwise: sort by ctime, newest first
50
51   -C      list entries by columns
52
53   --color[=WHEN]
54     colorize the output; WHEN can be 'never', 'auto', or 'always' (the default); more info below
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
```

Figure 6: Man Page for `ls`

# Unlocking Power

Once you've familiarized yourself with a command, you can do a lot with simple commands

```
7 [ec2-user@ip-172-31-16-171 ~]$ cd /var
8 [ec2-user@ip-172-31-16-171 var]$ ls -lrsah
9 total 8.0K
10 drwxr-xr-x 2 root root 6 Apr  9 2019 yp
11 lrwxrwxrwx 1 root root 6 Apr 28 19:53 run -> ../run
12 drwxr-xr-x 2 root root 6 Apr  9 2019 preserve
13 drwxr-xr-x 2 root root 6 Apr  9 2019 apt
14 drwxr-xr-x 2 root root 6 Apr  9 2019 nis
15 drwxr-xr-x 2 root root 6 Apr  9 2019 local
16 drwxr-xr-x 2 root root 6 Apr  9 2019 gopher
17 drwxr-xr-x 2 root root 6 Apr  9 2019 games
18 drwxr-xr-x 2 root root 6 Apr  9 2019 adm
19 lrwxrwxrwx 1 root root 18 Apr 28 19:53 mail -> spool/mail
20 lrwxrwxrwx 1 root root 11 Apr 28 19:53 lock -> ./run/lock
21 drwxr-xr-x 3 root root 18 Apr 28 19:54 kerberos
22 drwxr-xr-x 3 root root 18 Apr 28 19:54 empty
23 drwxr-xr-x 3 root root 18 Apr 28 19:54 db
24 drwxr-xr-x 2 root root 19 Apr 28 19:54 account
25 drwxr-xr-x 6 root root 63 Apr 28 19:54 cache
26 drwxrwxrwt 3 root root 85 May  9 11:59 tmp
27 drwxr-xr-x 9 root root 97 Apr 28 19:54 spool
28 +rw-r--r-- 1 root root 163 Apr 28 19:53 .updated
29 dr-xr-xr-x 18 root root 257 May  9 11:59 ..
30 drwxr-xr-x 19 root root 269 May  9 11:59 .
31 drwxr-xr-x 7 root root 328 May  9 11:59 log
32 drwxr-xr-x 39 root root 4.0K Apr 28 19:55 lib
```

Figure 7: ls sorting by size in reverse, longlisting format with all files and human readable sizes

# Knowing Your OS Version

Sometimes, it is necessary to know your OS version whether to debug or to search for help. `uname` prints out details about your current machine's operating system

```
33 [ec2-user@ip-172-31-16-171 var]$ uname -a
34 Linux ip-172-31-16-171.ec2.internal 5.10.109-104.500.amzn2.x86_64 #1 SMP Wed Apr 13 20:31:43 UTC 2022 x86_64 x86_64 x86_64 GNU/Linux
```

Figure 8: `uname -a` printing out all information about the machine

# Assignment 1

# Logging Into AWS

Once a non-root user is created by the server administrator, the user can log in based on the email they are provided or the link <https://console.aws.amazon.com/console/home> with their 12 digits account ID provided

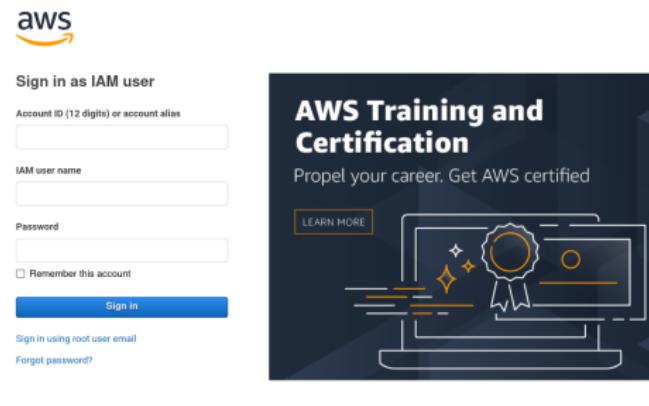


Figure 9: AWS Login Page

# EC2 Instance Creation

Once logged in, you can create an EC2 instance by searching EC2 on the search bar, or navigating to the creation menu

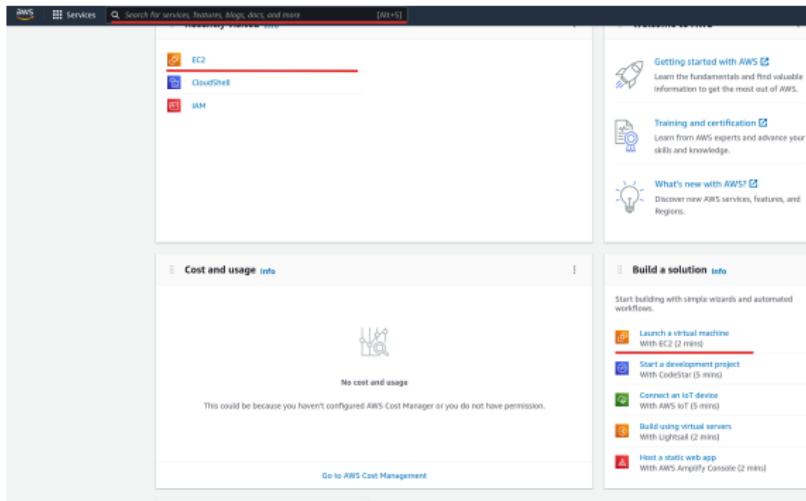


Figure 10: Methods to Navigate to EC2

# EC2 Instance Creation

Select the default settings for setting up an EC2 instance and for the key-pair, create a new key-pair or existing key-pair. Once you're done, you will see your instance loaded.

Instances (1) <a href="#">Info</a>				
<input type="checkbox"/> <a href="#">Search</a>				
Name	Instance ID	Instance state	Instance type	Status check
<input type="checkbox"/> Assignment1	<a href="#">i-0611e5a5d2069d3d8</a>	<span>Running</span>	t2.micro	-

Figure 11: Loaded Instance

Click on the blue highlighted link under 'Instance ID' to enter your id settings

# EC2 Instance Screen

You will see the following screen. Take note of your Public IPv4 address as you will need it to log into your instance.

The screenshot shows the AWS EC2 Instances page with the following details for the instance **i-0611e5a5d2069d3d8 (Assignment1)**:

- Public IPv4 address:** 100.26.61.244 | [Open address](#)
- Private IPv4 DNS:** ec2-100-26-61-244.compute-1.amazonaws.com | [Open address](#)
- Private IP address:** 172.31.27.105
- Instance state:** Pending
- Private IP DNS name (IPv4 only):** ip-172-31-27-105.ec2.internal
- Elastic IP address:** -
- Auto-assigned IP address:** 100.26.61.244 (Public IP)
- AMI ID:** ami-0611e5a5d2069d3d8
- Subnet ID:** subnet-0399fcfbf16041381
- Auto Scaling Group name:** -

**Details Tab:** Instance details, Info, Security, Networking, Storage, Status checks, Monitoring, Tags

**Platform:** Amazon Linux (Virtex)

**Platform details:** Linux/UNIX

**Launch time:** Tue May 10 2022 15:49:06 GMT+0800 (Singapore Standard Time) (1 minute)

**Monitoring:** disabled

**Termination protection:** Disabled

**Instance auto-recovery:** Default

**AWS Compute Optimizer Finding:** User: arn:aws:iam::707905225948:user/XBm-Restict is not authorized to perform: correlate-optimizer:GetEnvironmentStatus on resource: \* because no identity-based policy allows the compute-optimizer:GetEnvironmentStatus action. (HTTP)

Figure 12: Instance Page

# Logging In

Once your instance is running, you should be able to click on the connect button (if not, refresh and check again)

The screenshot shows the AWS EC2 Instances page for an instance named 'i-0011e5a5d2069d3d8 (Assignment1)'. The instance is currently in a 'Pending' state. A red arrow points to the 'Connect' button located in the top right corner of the instance's detail card.

**Instance summary for i-0011e5a5d2069d3d8 (Assignment1)**

Updated less than a minute ago

**Instance ID:** i-0011e5a5d2069d3d8 (Assignment1)

**IPv4 address:** -

**Hostname type:** IP name: ip-172-31-27-105.ec2.internal

**Instance type:** t2.micro

**VPC ID:** vpc-0e5a04ea205207881

**Subnet ID:** subnet-0299cfa0f16041591

**Data** | **Security** | **Networking** | **Storage** | **Status checks** | **Monitoring** | **Tags**

**Platform:** Amazon Linux (Verified)

**Platform details:** Linux/UNIX

**Launch time:** Fri May 10 2022 15:49:06 GMT+0800 (Singapore Standard Time) (1 minute)

**Public IPv4 address:** 100.26.61.244 | [Open address](#)

**Private IPv4 address:** 172.31.27.105

**Public IPv6 DNS:** ec2-100-26-61-244.compute-1.amazonaws.com | [Open address](#)

**Private IPv6 DNS:** ip-172-31-27-105.ec2.internal

**Elastic IP addresses:** -

**Auto Assigned IP address:** 100.26.61.244 (Public IP)

**IAW (IP Address:** -)

**AWS Compute Optimizer findings:**

Amazon Compute Optimizer is not authorized to perform: compute-optimizer:GetEnvironmentStatus on resource: \* because no identity-based policy allows the compute-optimizer:GetEnvironmentStatus action

**Auto Scaling Group name:** -

**AMI ID:** ami-0e123f778911c1b640

**AMI name:** amzn2-ami-kernel-5.10-hvm-2.0.20220426.0-sd8\_64-gp2

**AMI location:** amzn2/amzn2-ami-kernel-5.10-hvm-2.0.20220426.0-sd8\_64-gp2

**Monitoring:** disabled

**Termination protection:** disabled

**Instance auto-recovery:** default

Figure 13: Connect Button

# Logging In

Once you have clicked on the connect button, you will be redirected to a page like this. Click to the 'SSH client' tab and copy the sample command

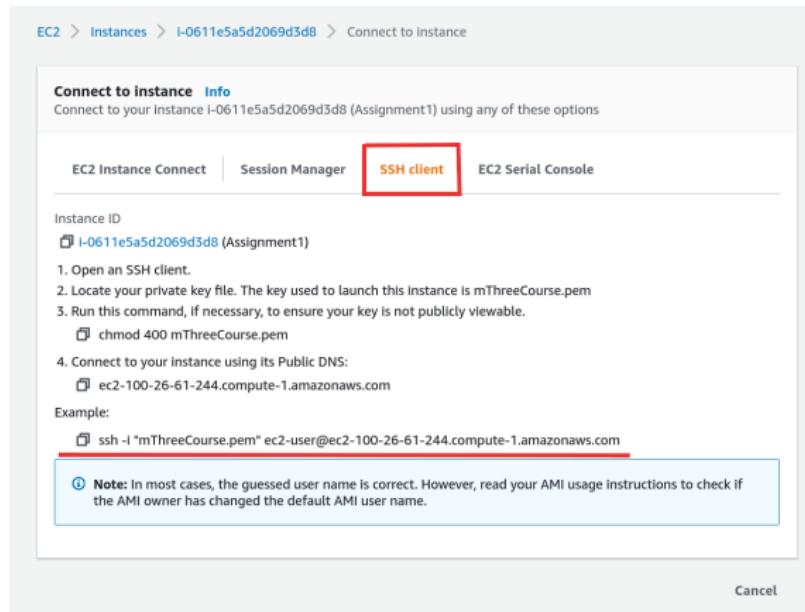


Figure 14: Sample Commands

## Logging In

On first creating a new key-pair, you should have downloaded a .pem or .cer file. Move that over to a secure folder location (e.g., .ssh folder or a project folder).

Open a terminal or cd to **wherever you have stored your .pem/.cer file**. Then paste the copied command. If your file is not in the current folder, you will receive this error

```
SyncThing/mThree/Linux took 5s + ssh -i "mThreeCourse.pem" ec2-user@ec2-100-26-61-244.compute-1.amazonaws.com
Warning: Identity file mThreeCourse.pem not accessible: No such file or directory.
The authenticity of host 'ec2-100-26-61-244.compute-1.amazonaws.com (100.26.61.244)' can't be established.
ED25519 key fingerprint is SHA256:03qBNQnwIDC0NNNdwokhBw3L+ONF1qJZUPIBU0Hu13cA.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? _
```

Figure 15: File not found error

# Logging In

Once you have pasted the command, type “yes” to the prompt and press Enter. You will be logged into the instance.

If an error still occurs, type `chmod 400 xxx.pem` and it should work.

```
SyncThing/mThree/Linux took 58s + ssh -i ~/.ssh/mThreeCourse.pem ec2-user@ec2-100-26-61-244.compute-1.amazonaws.com
The authenticity of host 'ec2-100-26-61-244.compute-1.amazonaws.com (100.26.61.244)' can't be established.
ED25519 key fingerprint is SHA256:03qBNQmwIDC0NNdwokhBw3L+ONF1qJZUPIBUOHuL3cA.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-100-26-61-244.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

--| --|_ )
_| ( _ /   Amazon Linux 2 AMI
__| \__|__|  
  
https://aws.amazon.com/amazon-linux-2/
2 package(s) needed for security, out of 2 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-27-105 ~]$ 
```

Figure 16: Logged in

## Installing Packages

To install programs (also known as packages in Linux), Amazon Linux uses `yum`. The command to install a package is `yum install packagename`

In this example, we will be installing `tree`. However, you can only installing packagess as the root user. If not, you will receive the error below. (You can also use the `sudo` command if you have been given those permissions)

```
[ec2-user@ip-172-31-27-105 ~]$ yum install tree
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
You need to be root to perform this command.
```

Figure 17: Installing as Non-Root

## Installing Packages

To login as the root user, the command is `sudo -i`. Once you do, you will see that the user has changed.

```
[ec2-user@ip-172-31-27-105 ~]$ sudo -i  
[root@ip-172-31-27-105 ~]# ]
```

Figure 18: Logging In as Root

# Installing Packages

As the root user, you may now install packages.

```
[root@ip-172-31-27-105 ~]# yum install tree
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Resolving Dependencies
--> Running transaction check
---> Package tree.x86_64 0:1.6.0-10.amzn2.0.1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package           Arch      Version            Repository          Size
=====
Installing:
tree             x86_64   1.6.0-10.amzn2.0.1  amzn2-core          47 k

Transaction Summary
=====
Install 1 Package

Total download size: 47 k
Installed size: 83 k
Is this ok [y/d/N]: y
Downloading packages:
tree-1.6.0-10.amzn2.0.1.x86_64.rpm | 47 kB  00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : tree-1.6.0-10.amzn2.0.1.x86_64
  Verifying   : tree-1.6.0-10.amzn2.0.1.x86_64
                                                               1/1
                                                               1/1

Installed:
  tree.x86_64 0:1.6.0-10.amzn2.0.1

Complete!
[root@ip-172-31-27-105 ~]# _
```

Figure 19: Installing tree

# Tree

The tree package allows you to print a tree diagram of the current directory and any subdirectories. Here, we print the /opt directory.

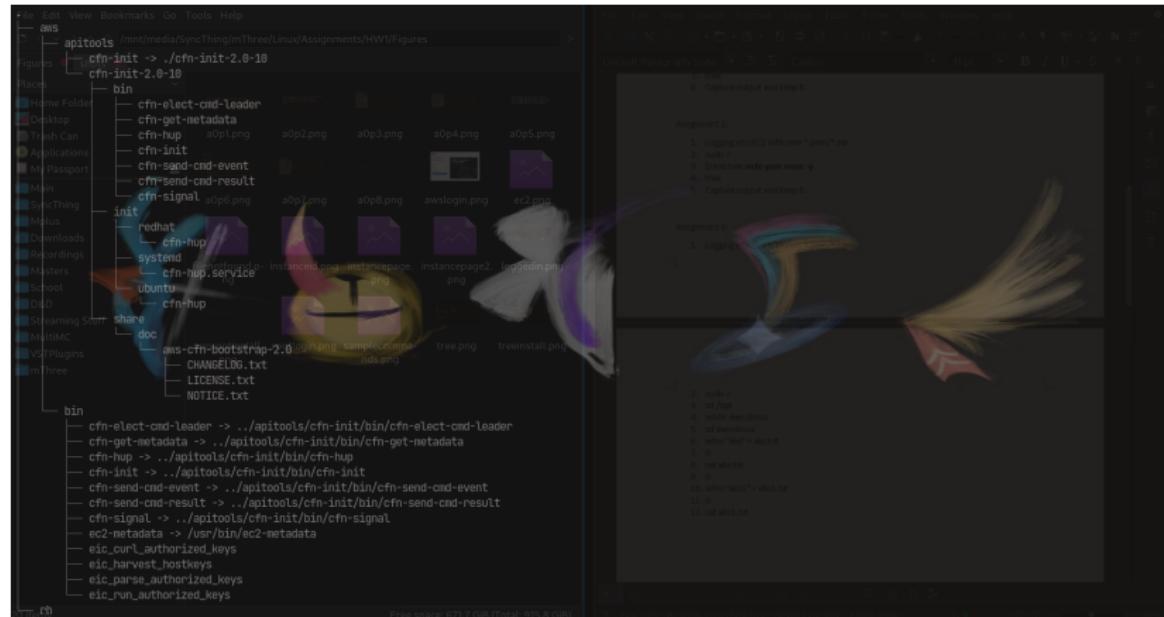


Figure 20: Tree Diagram of /opt

## Assignment 2

# Uninstalling Packages

yum is also used to uninstall packages. The command to uninstall a package is `yum erase packagename`

Here, we uninstall tree. We can also pass the `-y` flag so that the `erase` command assumes the answer 'yes' automatically

```
[root@ip-172-31-27-105 opt]# yum erase tree -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package tree.x86_64 0:1.6.0-10.amzn2.0.1 will be erased
--> Finished Dependency Resolution
amzn2-core/2/x86_64
Dependencies Resolved

=====
Package           Arch      Version            Repository        Size
=====
Removing:        tree      x86_64   1.6.0-10.amzn2.0.1  @amzn2-core      83 k

Transaction Summary
=====
Remove 1 Package

Installed size: 83 k
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Erasing   : tree-1.6.0-10.amzn2.0.1.x86_64
Verifying  : tree-1.6.0-10.amzn2.0.1.x86_64
Removed:
tree.x86_64 0:1.6.0-10.amzn2.0.1
Complete!
```

Figure 21: Uninstalling tree

## Uninstalling Packages

If you try to uninstall a package that does not exist, yum will simply throw an error

```
[root@ip-172-31-27-105 opt]# yum erase tree -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No Match for argument: tree
No Packages marked for removal
```

Figure 22: Attempting to uninstall tree after it has been uninstalled

# Assignment 3

## Making Directories

Inside the terminal, you can create new folders using the `mkdir` (make directory) command.

In this example, we create a new directory called `demolinux` in the `/opt` directory. You can see that the `demolinux` directory only appears in the `ls` statement after we execute the `mkdir demolinux` statement

```
[ec2-user@ip-172-31-27-105 ~]$ sudo -i  
[root@ip-172-31-27-105 ~]# cd /opt  
[root@ip-172-31-27-105 opt]# ls  
aws rh  
[root@ip-172-31-27-105 opt]# mkdir demolinux  
[root@ip-172-31-27-105 opt]# ls  
aws demolinux rh
```

Figure 23: Making a new directory called ‘demolinux’

## Creating New Files

We can create new files by using the echo command.

Used alone, echo simply prints the input to the console. But the output can instead be redirected into a .txt file

```
[root@ip-172-31-27-105 demolinux]# echo "abc"  
abc  
[root@ip-172-31-27-105 demolinux]# ls  
[root@ip-172-31-27-105 demolinux]# echo "abc" > abc.txt  
[root@ip-172-31-27-105 demolinux]# ls  
abc.txt
```

Figure 24: Echo to create a text file with content

## Print Content from Files

We can print the content from a file using the cat command

```
[root@ip-172-31-27-105 demolinux]# cat abc.txt  
abc
```

Figure 25: Printing Content of abc.txt

## Summary Process

We can thus echo content into files and print that content for later use using cat

```
[root@ip-172-31-27-105 demolinux]# echo "abc1" > abc1.txt
[root@ip-172-31-27-105 demolinux]# ls
abc1.txt  abc.txt
[root@ip-172-31-27-105 demolinux]# cat abc1.txt
abc1
[root@ip-172-31-27-105 demolinux]# ]
```

Figure 26: Full Demonstration

# History

```
[ec2-user@ip-172-31-27-105 ~]$ sudo -i
[root@ip-172-31-27-105 ~]# cd /opt
[root@ip-172-31-27-105 opt]# ls
aws rh
[root@ip-172-31-27-105 opt]# mkdir demolinux
[root@ip-172-31-27-105 opt]# ls
aws demolinux rh
[root@ip-172-31-27-105 opt]# cd demolinux
[root@ip-172-31-27-105 demolinux]# echo "abc"
abc
[root@ip-172-31-27-105 demolinux]# ls
[root@ip-172-31-27-105 demolinux]# echo "abc" > abc.txt
[root@ip-172-31-27-105 demolinux]# ls
abc.txt
[root@ip-172-31-27-105 demolinux]# cat abc.txt
abc
[root@ip-172-31-27-105 demolinux]# ls
abc.txt
[root@ip-172-31-27-105 demolinux]# echo "abc1" > abc1.txt
[root@ip-172-31-27-105 demolinux]# ls
abc1.txt abc.txt
[root@ip-172-31-27-105 demolinux]# cat abc1.txt
abc1
[root@ip-172-31-27-105 demolinux]# _
```

Figure 27: History of Commands Used

# Assignment 4

## Assignment 4

This assignment is a test of the skills from the Basic Commands Assignment and Assignment 3

# History

```
SyncThing/mThree/Linux took 1m 22s + ssh -i ~/ssh/mThreeCourse.pem ec2-user@ec2-54-234-244-214.compute-1.amazonaws.com
Last login: Tue May 10 08:50:09 2022 from 39.109.242.62
      _\   _)
      \_ ( _/  Amazon Linux 2 AMI
      ___\_\_\_\_|
https://aws.amazon.com/amazon-linux-2/
2 package(s) needed for security, out of 2 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-21-8 ~]$ export TERM=linux
[ec2-user@ip-172-31-21-8 ~]$ sudo -i
[root@ip-172-31-21-8 ~]# cd /opt
[root@ip-172-31-21-8 opt]# mkdir demolinux
[root@ip-172-31-21-8 opt]# cd demolinux/
[root@ip-172-31-21-8 demolinux]# echo "abc" > abc.txt
[root@ip-172-31-21-8 demolinux]# ls -ltr
total 4
-rw-r--r-- 1 root root 4 May 10 08:52 abc.txt
[root@ip-172-31-21-8 demolinux]# cat abc.txt
abc
[root@ip-172-31-21-8 demolinux]# ls -lra
total 4
drwxr-xr-x 5 root root 44 May 10 08:51 ..
-rw-r--r-- 1 root root 4 May 10 08:52 abc.txt
drwxr-xr-x 2 root root 21 May 10 08:52 .
[root@ip-172-31-21-8 demolinux]# echo "abc1" > abc1.txt
[root@ip-172-31-21-8 demolinux]# ls
abc1.txt abc.txt
[root@ip-172-31-21-8 demolinux]# cat abc1.txt
abc1
[root@ip-172-31-21-8 demolinux]# _
```

Figure 28: Assignment 4

## ls -a

The `-a` flag in `ls` lists all files, including hidden files. In Assignment 4, it revealed the `..` and `.` directories, which are respectively the parent directory and the current directory.

```
[root@ip-172-31-21-8 demolinux]# ls -ltra
total 4
drwxr-xr-x 5 root root 44 May 10 08:51 ..
-rw-r--r-- 1 root root 4 May 10 08:52 abc.txt
drwxr-xr-x 2 root root 21 May 10 08:52 .
```

Figure 29: ls -a

# Assignment 5

## Assignment 5

In this assignment, we will create a http page using only the console/terminal and access it using the browser.

# Enable HTTP Traffic

Before we can display a http page however, we need to enable HTTP traffic for our instance. We can do so when setting up the instance as below.

▼ Network settings Edit

Network  
vpc-0d3a04ea203207633

Subnet  
No preference (Default subnet in any availability zone)

Auto-assign public IP

Enable

Security groups (Firewall) [Info](#)  
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

We'll create a new security group called '**launch-wizard-5**' with the following rules:

Allow SSH traffic from My IP  
39.109.242.62/32  
Helps you connect to your instance

Allow HTTPS traffic from the internet  
To set up an endpoint, for example when creating a web server

Allow HTTP traffic from the internet  
To receive requests from the internet directly

# Enable HTTP Traffic

Otherwise, go to the Security Tab in the Instance Page back on AWS and click on the available Security Group.

The screenshot shows the AWS Instance Summary page for an instance named 'Assignment5'. A red arrow points from the text above to the 'Security' tab in the navigation bar at the bottom of the page. The 'Security' tab is highlighted in red. Below it, the 'Details' tab is also visible. The main content area displays various instance details like Public IPv4 address (54.147.164.29), Private IPv4 address (172.31.51.91), and Auto Assigned IP address (54.147.164.29). The 'Security groups' section lists 'sg-0de6517354801395 (Launch-wizard-1)'. The 'Inbound rules' section is collapsed.

Figure 31: Security Tab on Instance Page

# Enable HTTP Traffic

Here, click on the 'Edit Inbound Rules' button.

The screenshot shows the AWS EC2 Security Groups page for a specific security group. The top navigation bar includes 'EC2 > Security Groups > sg-0ee60517354d01393 - launch-wizard-5'. The main section displays the security group details: Name (sg-0ee60517354d01393), Owner (767565525948), Security group ID (sg-0ee60517354d01393), Description (launch-wizard created 2022-05-10T00:56:12.062Z), and VPC ID (vpc-0ca04ea205207653). Below this, there are tabs for 'Inbound rules' (selected), 'Outbound rules', and 'Tags'. A message at the top says, 'You can now check network connectivity with Reachability Analyzer' with a 'Run Reachability Analyzer' button. The 'Inbound rules' table shows one entry: a single SSH rule (sgr-0596fa360b7fb5bd) allowing port 22 from 59.109.242.62/32. The table has columns for Name, Security group rule ID, IP version, Type, Protocol, Port range, Source, and Description. Buttons for 'Edit tags' and 'Edit inbound rules' are located at the bottom right of the table. A red arrow points from the text above to the 'Edit inbound rules' button.

Name	Security group rule ID	IP version	Type	Protocol	Port range	Source	Description
sgr-0596fa360b7fb5bd	IPv4	SSH	TCP	22	59.109.242.62/32	-	

Figure 32: Security Group Page

# Enable HTTP Traffic

Finally, add a rule for HTTP at the 0.0.0.0/0 address

The screenshot shows the 'Edit inbound rules' section of the AWS CloudFormation console. It displays two rules:

- Rule 1:** Type: SSH, Protocol: TCP, Port range: 22, Source: Custom (39.109.242.62/32), Description: optional.
- Rule 2:** Type: SSH, Protocol: TCP, Port range: 80, Source: Anywhere-IPv4, Destination: 0.0.0.0/0, Description: optional.

At the bottom, there are buttons for 'Cancel', 'Preview changes', and a prominent orange 'Save rules' button.

Figure 33: Adding HTTP Traffic Rule

# Install httpd Package

Back to the server, install the package httpd using yum. As httpd has a few dependencies, the output will be a bit longer.

```
[ec2-user@ip-172-31-31-91 ~]$ sudo -i  
[root@ip-172-31-31-91 ~]# export TERM=linux  
[root@ip-172-31-31-91 ~]# yum install httpd  
Loaded plugins: extras_suggestions, langpacks, priorities, update-notified  
amzn2-core  
Resolving Dependencies  
--> Running transaction check  
---> Package httpd.x86_64 0:2.4.53-1.amzn2 will be installed  
---> Processing Dependency: httpd-tools = 2.4.53-1.amzn2 for package: httpd-2.4.53-1.amzn2.x86_64  
---> Processing Dependency: httpd-filesystem = 2.4.53-1.amzn2 for package: httpd-2.4.53-1.amzn2.x86_64  
---> Processing Dependency: system-logos-httdp for package: httpd-2.4.53-1.amzn2.x86_64  
---> Processing Dependency: mod_http2 for package: httpd-2.4.53-1.amzn2.x86_64  
---> Processing Dependency: httpd-filesystem for package: httpd-2.4.53-1.amzn2.x86_64  
---> Processing Dependency: /etc/mime.types for package: httpd-2.4.53-1.amzn2.x86_64  
---> Processing Dependency: libaprutil-1.so.0() (64bit) for package: httpd-2.4.53-1.amzn2.x86_64  
---> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.53-1.amzn2.x86_64  
---> Running transaction check  
---> Package apr.x86_64 0:1.7.0-9.amzn2 will be installed  
---> Package apr-util.x86_64 0:1.6.1-5.amzn2.0.2 will be installed  
---> Processing Dependency: apr-util-bdb(x86-64) = 1.6.1-5.amzn2.0.2 for package: apr-util-1.6.1-5.amzn2.0.2.x86_64  
---> Package generic-logos-httdp.noarch 0:18.0.0-4.amzn2 will be installed  
---> Package httpd-filesystem.noarch 0:2.4.53-1.amzn2 will be installed  
---> Package httpd-tools.x86_64 0:2.4.53-1.amzn2 will be installed  
---> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed  
---> Package mod_http2.x86_64 0:1.15.19-1.amzn2.0.1 will be installed  
---> Running transaction check  
---> Package apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2 will be installed  
---> Finished Dependency Resolution
```

Figure 34: Installing httpd

# Install httpd Package

```
Dependencies Resolved

=====
Package           Arch      Version       Repository     Size
=====
Installing:
httpd            x86_64   2.4.53-1.amzn2      amzn2-core    1.3 M
Installing for dependencies:
apr               x86_64   1.7.0-9.amzn2      amzn2-core    122 k
apr-util          x86_64   1.6.1-5.amzn2.0.2  amzn2-core    99 k
apr-util-odbc    x86_64   1.6.1-5.amzn2.0.2  amzn2-core    19 k
generic-logos-httpd  noarch  18.0.0-4.amzn2      amzn2-core    19 k
httpd-filesystem  noarch  2.4.53-1.amzn2      amzn2-core    24 k
httpd-tools        x86_64   2.4.53-1.amzn2      amzn2-core    88 k
mailcap           noarch  2.1.41-2.amzn2      amzn2-core    31 k
mod_http2          x86_64   1.15.19-1.amzn2.0.1  amzn2-core    149 k

Transaction Summary
=====
Install 1 Package (+8 Dependent packages)

Total download size: 1.9 M
Installed size: 5.2 M
Downloading packages:
(1/9): apr-util-1.6.1-5.amzn2.0.2.x86_64.rpm | 99 kB  00:00:00
(2/9): apr-1.7.0-9.amzn2.x86_64.rpm          | 122 kB  00:00:00
(3/9): apr-util-odbc-1.6.1-5.amzn2.0.2.x86_64.rpm | 19 kB  00:00:00
(4/9): generic-logos-httpd-18.0.0-4.amzn2.noarch.rpm | 19 kB  00:00:00
(5/9): httpd-filesystem-2.4.53-1.amzn2.noarch.rpm | 24 kB  00:00:00
(6/9): httpd-tools-2.4.53-1.amzn2.x86_64.rpm | 88 kB  00:00:00
(7/9): httpd-2.4.53-1.amzn2.x86_64.rpm          | 1.3 MB  00:00:00
(8/9): mailcap-2.1.41-2.amzn2.noarch.rpm        | 31 kB  00:00:00
(9/9): mod_http2-1.15.19-1.amzn2.0.1.x86_64.rpm | 149 kB  00:00:00

Total                                         8.6 MB/s | 1.9 MB  00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : apr-1.7.0-9.amzn2.x86_64

```

Figure 35: Installing httpd

# Install httpd Package

```
[4/9]: generic-logos-httpd-18.0.0-4.amzn2.noarch.rpm | 19 kB 00:00:00
[5/9]: httpd-filesystem-2.4.53-1.amzn2.noarch.rpm | 24 kB 00:00:00
[6/9]: httpd-tools-2.4.53-1.amzn2.x86_64.rpm | 88 kB 00:00:00
[7/9]: httpd-2.4.53-1.amzn2.x86_64.rpm | 1.3 MB 00:00:00
[8/9]: mailcap-2.1.41-2.amzn2.noarch.rpm | 31 kB 00:00:00
[9/9]: mod_http2-1.15.19-1.amzn2.0.1.x86_64.rpm | 149 kB 00:00:00

Total                                         8.6 MB/s | 1.9 MB 00:00:00

Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : apr-1.7.0-9.amzn2.x86_64          1/9
Installing : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64 2/9
Installing : apr-util-1.6.1-5.amzn2.0.2.x86_64 3/9
Installing : httpd-tools-2.4.53-1.amzn2.x86_64 4/9
Installing : generic-logos-httpd-18.0.0-4.amzn2.noarch 5/9
Installing : mailcap-2.1.41-2.amzn2.noarch 6/9
Installing : httpd-filesystem-2.4.53-1.amzn2.noarch 7/9
Installing : mod_http2-1.15.19-1.amzn2.0.1.x86_64 8/9
Installing : httpd-2.4.53-1.amzn2.x86_64 9/9
Verifying : apr-1.6.1-5.amzn2.0.2.x86_64
Verifying : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64
Verifying : httpd-2.4.53-1.amzn2.0.1.x86_64
Verifying : mod_http2-1.15.19-1.amzn2.0.1.x86_64
Verifying : httpd-2.4.53-1.amzn2.x86_64
Verifying : httpd-tools-2.4.53-1.amzn2.x86_64
Verifying : mailcap-2.1.41-2.amzn2.noarch
Verifying : generic-logos-httpd-18.0.0-4.amzn2.noarch
Verifying : httpd-2.4.53-1.amzn2.x86_64
Verifying : apr-1.7.0-9.amzn2.x86_64

Installed:
httpd.x86_64 0:2.4.53-1.amzn2

Dependency Installed:
apr.x86_64 0:1.7.0-9.amzn2           apr-util.x86_64 0:1.6.1-5.amzn2.0.2
httpd-filesystem.noarch 0:2.4.53-1.amzn2   httpd-tools.x86_64 0:2.4.53-1.amzn2
                                            apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2
                                            mailcap.noarch 0:2.1.41-2.amzn2
                                            generic-logos-httpd.noarch 0:18.0.0-4.amzn2
                                            mod_http2.x86_64 0:1.15.19-1.amzn2.0.1

Complete!
[root@ip-172-31-31-91 ~]# _
```

Figure 36: Installing httpd

# Creating the HTTP File

You can then create a html file using the echo command.  
Outputting the html to index.html ensures that the file is loaded whenever the server is accessed through HTTP.

```
[root@ip-172-31-31-91 ~]# echo "<html><body><h1>Welcome! <br> Design & Developed by Daniel Tan</h1></body></html>" > /var/www/html/index.html  
[root@ip-172-31-31-91 ~]# cat /var/www/html/index.html  
<html><body><h1>Welcome! <br> Design & Developed by Daniel Tan</h1></body></html>
```

Figure 37: Creating the HTTP File

## Starting httpd

Just installing a daemon or service like httpd does not automatically mean it is working in the background. To do so, you must enable and start the daemon using `service daemonname on` and `chkconfig daemonname on` in Amazon Linux

```
[root@ip-172-31-31-91 ~]# sudo service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-31-91 ~]# sudo chkconfig httpd on
Note: Forwarding request to 'systemctl enable httpd.service'.
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
```

Figure 38: Enable and Start HTTP Daemon

# Acquiring Public IP

From the Instance Page on the AWS Console, we need to get the public IP

The screenshot shows the 'Instance summary' page for an Amazon Linux 2 instance with the ID i-05cde0b82f5b039d2. The page includes sections for Instance ID, IP address, Hostname type, Instance type, VPC ID, and Subnet ID. On the right, detailed information is provided for both the Public and Private IPv4 addresses. The Public IPv4 address is listed as 54.147.164.29 with a link to 'open address'. The Private IPv4 address is listed as 172.31.31.91. The Instance State is shown as 'Running'. A note at the bottom indicates an 'AWS Compute Optimizer' error related to the compute-optimizer service.

Public IPv4 address	Private IPv4 address
54.147.164.29 [open address]	172.31.31.91
Private IPv4 DNS name (IPv4 only)	Private IPv4 DNS
ip-172-31-31-91.ec2.internal	ec2-54-147-164-29.compute-1.amazonaws.com [open address]
Elastic IP addresses	Amazon private resource DNS name
-	IPv4 (4)
Auto-assigned (IP address)	Auto-assigned (IP address)
54.147.164.29 [Public IP]	54.147.164.29 [Public IP]
IAW Role	IAW Role
-	-

Figure 39: Acquiring Public IP

# Accessing HTML Page

With the public ip, we can access the page by pasting `http://publicip`. Remember not just to paste the IP as most browsers nowadays automatically use https. You may receive an error but simply click on 'Continue to HTTP Site'

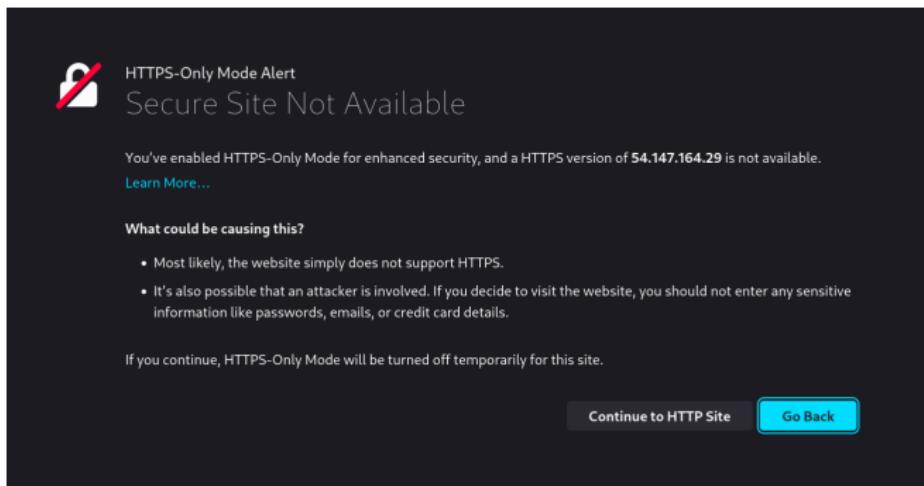


Figure 40: HTTP Error

## Accessing HTML Page

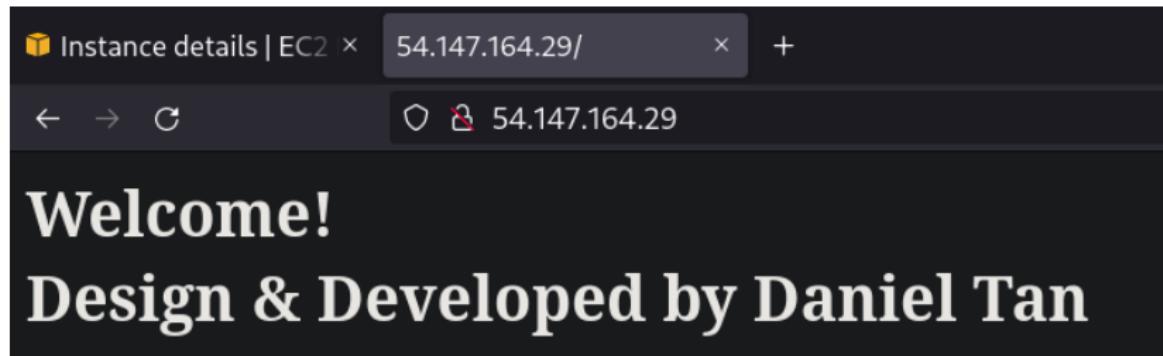


Figure 41: Rendered HTTP Page