Cloud Server Project & Wideo Explainer ICT171 Assignment 2

Table of Contents

GitHub Repository Link	3
Server Link	3
Instance Server Launch	3
SSH Connect client	5

GitHub Repository Link

https://github.com/NEELLACHWANI/ICT171

Server Link

http://3.27.234.91/

Instance Server Launch

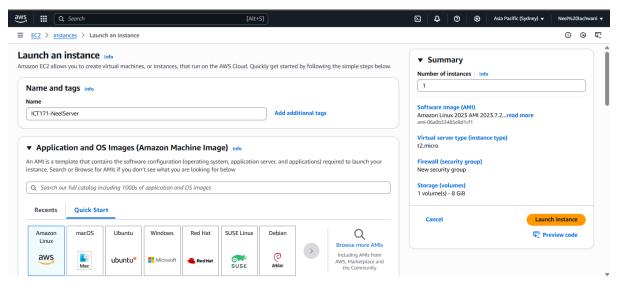


Figure 1: Instance configurations

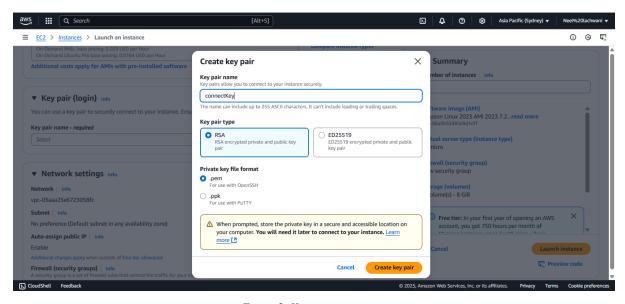


Figure 2: Key pair generation

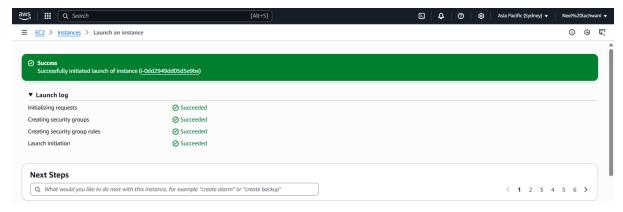


Figure 3: Instance launched

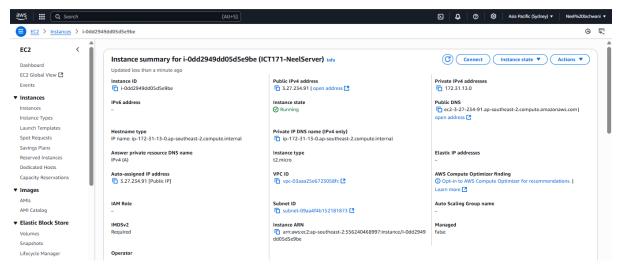


Figure 4: Instance summary

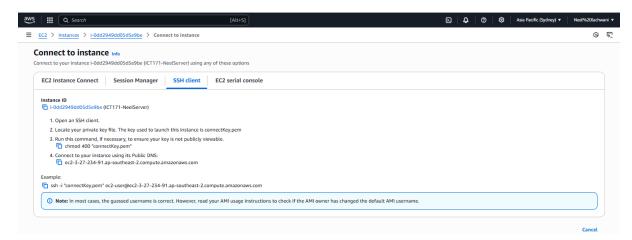


Figure 5: SSH Connect client option

SSH Connect client

Figure 6: Connected to SSH Server

Command:

"ssh -i "connectKey.pem" ec2-user@ec2-3-27-234-91.ap-southeast-

2.compute.amazonaws.com"

Description:

It allows you to safely access an AWS EC2 instance via this SSH command. The private key file is given to '-i' in this flag such as 'connectKey.pem'. By default, a user named 'ec2-user' will be created on Amazon Linux. The hostname is a short form that takes you to the public DNS of the target EC2 in the Asia Pacific (Sydney) region. Using this method, communication is protected using keys, eliminating the need for passwords.

======================================	Architecture	Version	Repository	Size
======================================				
httpd	x86_64	2.4.62-1.amzn2023	amazonlinux	48 1
nstalling dependencies:				
apr	x86_64	1.7.5-1.amzn2023.0.4	amazonlinux	129 k
pr-util	x86_64	1.6.3-1.amzn2023.0.1	amazonlinux	98
eneric-logos-httpd	noarch	18.0.0-12.amzn2023.0.3	amazonlinux	19
nttpd-core	x86_64	2.4.62-1.amzn2023	amazonlinux	1.4 [
ttpd-filesystem	noarch	2.4.62-1.amzn2023	amazonlinux	14
ttpd-tools	x86_64	2.4.62-1.amzn2023	amazonlinux	81
ibbrotli	x86_64	1.0.9-4.amzn2023.0.2	amazonlinux	315
ailcap	noarch	2.1.49-3.amzn2023.0.3	amazonlinux	33
stalling weak dependencies:				
pr-util-openssl	x86_64	1.6.3-1.amzn2023.0.1	amazonlinux	17
iod_http2	x86_64	2.0.27-1.amzn2023.0.3	amazonlinux	166
nod_lua	x86_64	2.4.62-1.amzn2023	amazonlinux	61
ransaction Summary				
nstall 12 Packages				
otal download size: 2.3 M				
stalled size: 6.9 M				
ownloading Packages:				
/12): apr-util-openssl-1.6.3	314 kB/s 17 ki			
/12): apr-1.7.5-1.amzn2023.6	2.1 MB/s 129 ki			
/12): apr-util-1.6.3-1.amzn2	1.4 MB/s 98 ki			
1/12): generic-logos-httpd-18	881 kB/s 19 ki			
5/12): httpd-2.4.62-1.amzn202	2.2 MB/s 48 ki			
(6/12): httpd-filesystem-2.4.62-1.amzn2023.noarch.rpm			510 kB/s 14 ki	B 00:00

Figure 7: Download Apache service on EC2 Server

```
| State| | S
```

Figure 8: Installed

Command:

"sudo yum install httpd -y"

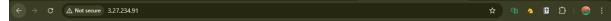
Description:

This install the Apache HTTP server on Amazon Linux or CentOS using the command sudo yum install httpd -y. The sudo prefix is needed for installing software because it needs permission from the superuser. That command installs the Apache web server, so the system can serve website content. The '-y' flag makes sure that any prompts are automatically approved, so the entire installation can take place without needing user input.

```
[ec<sup>2</sup>-user@ip-172-31-13-0 ~]$ sudo systemctl start httpd
[ec<sup>2</sup>-user@ip-172-31-13-0 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ec<sup>2</sup>-user@ip-172-31-13-0 ~]$
```

Figure 9: Start and enable the Apache Server

Figure 10: Replace the var index.html file



ICT171 Server is ON! Hello from server (Neel)

Figure 11: Public IP webhosting and hello from server

The screenshot proves, deploying a web server on an AWS EC2 instance completed successfully. You set up a virtual machine by using Amazon EC2 and manually accessed it through SSH with your private key. Once inside the server, you installed Apache HTTP server by using the command 'sudo yum install httpd -y' which permitted the server to provide web content. Then, you launched and enabled the Apache service, so it starts together with your system. You should then edit the default web page, '/var/www/html/index.html' and write your custom message "ICT171 Server is ON!" Greetings from server (Neel)" Besides, you allowed HTTP traffic on port 80 in the EC2 security group which lets the web page be seen from any browser because message is showing in the browser, you know that your server is running successfully.