

# Scale Cloud Services using Orchestration



Submitted for the course:

*NETV 379 - Cloud Computing*

November 2024

Ryufath Alief Adhyaksa Putera Soepeno (23792097)

Information Systems

Faculty of Engineering & Technology

Sampoerna University

&

Applied Computing

College of Applied Science & Technology

University of Arizona

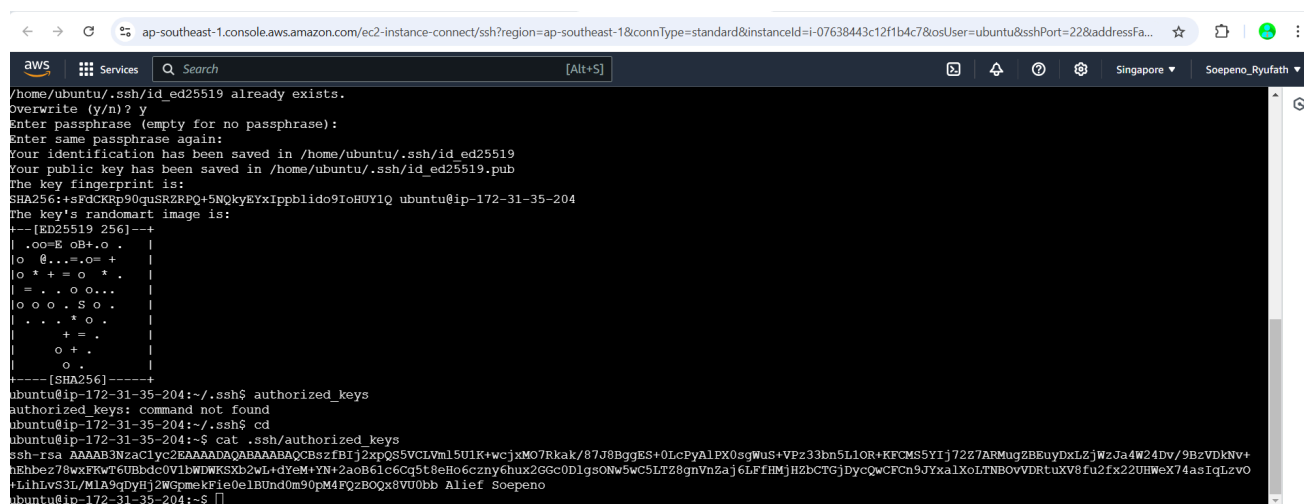
## 1. Introduction

This project aims to utilize automation and orchestration tools to enhance the scalability and efficiency of web services in a cloud environment. Specifically, it involves configuring a pre-existing AWS instance to support Ansible-based configuration management, enabling streamlined and consistent setup across multiple cloud resources. Through this approach, the project aims to demonstrate how automation can simplify large-scale web service management, reduce manual tasks, and promote high-availability deployments.

## 2. Discussion

### Part 1: Initial Actions

To begin, connect to the AWS instance created in Lab 1, running on Ubuntu 24.04 LTS. The writer generated a key pair named “*alief soepeno*” and add the required details, then select the instance to connect and run it. After connecting to the instance’s Linux terminal, update the system and check if the Linux version is up to date. The instance’s output will display after these initial steps.



```

aws
Services Search [Alt+S]
ap-southeast-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-southeast-1&connType=standard&instanceId=i-07638443c12f1b4c7&osUser=ubuntu&sshPort=22&addressFa...
/home/ubuntu/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_ed25519
Your public key has been saved in /home/ubuntu/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:+sFdCKRp90quSRZRPq+5NqkyEYxIppblido9IoHUV1Q ubuntu@ip-172-31-35-204
The key's randomart image is:
+--[ED25519 256]--+
| .oo=E oB+.o . |
|o @...=.o= + |
|o * + = o * |
| = . . o o... |
|o o o . S o . |
| . . . * o . |
| . + = . |
| o + . |
| o . |
+-----[SHA256]-----+
ubuntu@ip-172-31-35-204:~/.ssh$ authorized_keys
authorized_keys: command not found
ubuntu@ip-172-31-35-204:~/.ssh$ cd
ubuntu@ip-172-31-35-204:~/$ cat .ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCSzfBIj2xpQ85VCLVml5U1K+wcjxMO7Rkak/87J8BggES+0LcPyAlFX0sgWuS+VPz33bn5L1OR+KFCMS5YIj72Z7ARMugZBEuyDxLZjWzJa4W24Dv/9BzVDkNV+
nEhbez78wxFRt6UBBdc0V1bWDRK5Xb2wL+dYeM+YN+2aoB61c6Cq5t8eHo6czny6hux2GGc0Dlgs0Nw5wC5LT28gnVnZaj6LFfHMjHZbCTGjDycQwCFcn9JYxa1XoLTNBovVDRtuXV8fu2fx22UHWex74asIqLzvo
+LihlVS3L/MlA9qdyHj2WgpmekFie0elBUnd0m90pM4FQzBOQx8VU0bb Alief Soepeno
ubuntu@ip-172-31-35-204:~/$

```

Next, generate an RSA key pair and append the public key to `authorized\_keys` without adding arguments for simplicity. Return to the home directory and verify that the public key has

been correctly added to ``authorized_keys``. Finally, connect to localhost via SSH to complete the setup.

```

aws
Services Search [Alt+S]
Usage of /: 44.5% of 6.71GB Users logged in: 0
Memory usage: 61% IPv4 address for enx0: 172.31.35.204
Swap usage: 0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.
  https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

19 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

*** System restart required ***
Last login: Sat Nov 2 06:52:49 2024 from 3.0.5.36
ubuntu@ip-172-31-35-204:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ED25519 key fingerprint is SHA256:5nBo944bUky5DrYrx+16UHS11WKR9sWiH/QRr203PM.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ED25519) to the list of known hosts.

ubuntu@ip-172-31-13-222:~/.ssh$ cat id_rsa.pub >> authorized_keys
ubuntu@ip-172-31-13-222:~/.ssh$ cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCA66yFjhbEm0Yr4UIMPvSyBrwbl/gWnbYuxxKnKb17uBpCpJDCMxcM6BTmIya1DZv+L/9yzkTYmG7xkY9inpu56nOx4A3nibTa/C3Hh8oKHdS
JEXVcqyVJGjdtlob21BR+xi7KVpp2uBqWuleWduFReYc8I5dQWk35Gysv+5BvAxLvBks/v7retbEdd9pHRieG51APLUHnG3qGwvq4Xc0TisVmGXu4L5f2cq6QCPiLLysYnHvQciH3W+HaPpkA
45FFgVAST6T//bI8CYvHvthGab2wBP9j4qJusWHhA9zAof5FS1xOBtTzUSsjy/rhokgS9QbBYgJVBW0YXgkI03L NETV379
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCA66yFjhbEm0Yr4UIMPvSyBrwbl/gWnbYuxxKnKb17uBpCpJDCMxcM6BTmIya1DZv+L/9yzkTYmG7xkY9inpu56nOx4A3nibTa/C3Hh8oKHdS
2wdUkCM5Be45xIJfQ58InywiI6F5OSEnmUjz7h66xaAzJ902VwoeuyoPUi/+E132aFRBvt/FxQpMcRz241j/LMWVIk2Ou09h3EBfOGxE2RSdgulVHHVIOESam5yK/Hz1Uqf6wZeDtA5yK4K4
oCJ4Y8bvx3s8zXzgWt075rgYoEXXS/XLycWHyJAjjz7FO+C0aXKxVr2cYUeanYR/qQpUuHnF11zYmLBj7tgzfyG111+WTd3U2dI6IPOwExAnqklpxr9xT6idc01c2Ajd2J4MLH1kgPTh5VKhF
UQo6RnDUvo21J5no01iupZ1LAHvBbf3VKbz0ZyELdp6TYaKojMu7OZ6NPh1TLdV1EgzjUi5CkXzWuhLadFUS11kCu/hcyaIvHwQX51TL/QliTU57J0= ubuntu@ip-172-31-13-222

```

## Part II: Ansible

First, start by updating the instance and its properties using the ``sudo apt-get update`` command. Then, add the Ansible PPA to allow for easy installation of Ansible and its components. Next, create a custom Ansible inventory file by opening the SSH window and using ``sudo nano /etc/ansible/hosts``. Add the server name in a structured format, replacing any spaces with underscores, and write it in the format "Server\_name Localhost" for clarity.

To check if Ansible is set up correctly, try running a basic command, such as ``ansible all -m ping``, which will verify connectivity. Also, install the ``cowsay`` package to test Ansible's functionality further.

```
ubuntu@ip-172-31-13-222:~$ cowsay 'Hello World'
```

```
< Hello World >
```

```

      ^  ^
     (oo)\_____
    (__)\       )\/\
    ||----w |
    ||     ||

```

```
ubuntu@ip-172-31-13-222:~$
```

```
ubuntu@ip-172-31-13-222:~$ fortune | cowsay
```

```

/ Q: What do you call a WASP who doesn't \
| work for his father, isn't a          |
|                                     |
| lawyer, and believes in social causes? |
\ A: A failure.                        /

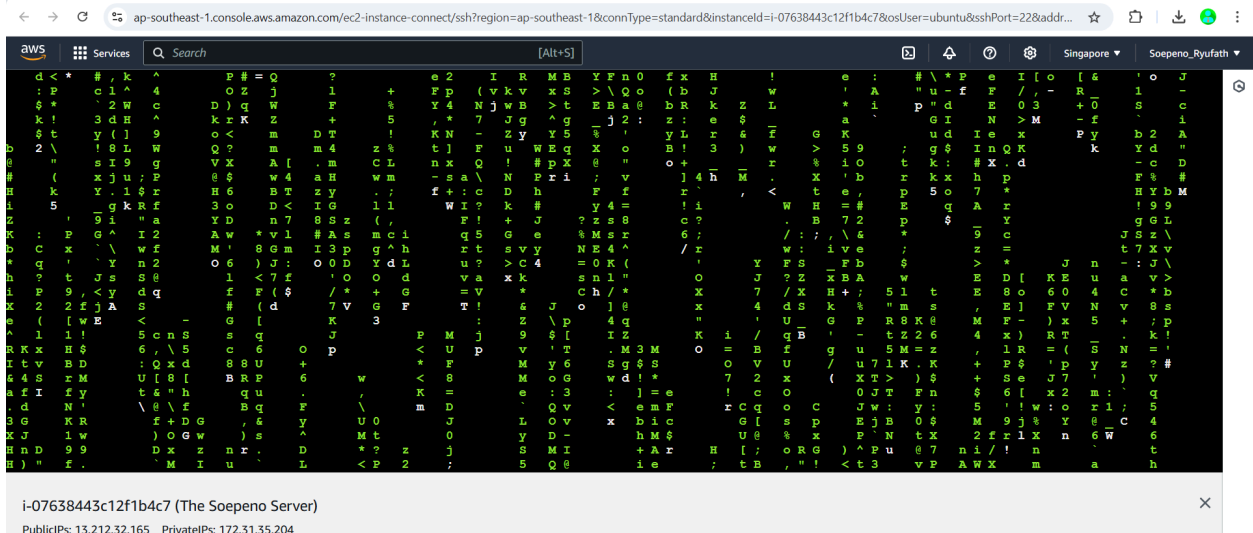
```

```

      ^  ^
     (oo)\_____
    (__)\       )\/\
    ||----w |
    ||     ||

```

Then, configure the `cmatrix.yml` file within Ansible by setting up the configurations needed for later steps. Once configurations are complete, execute the `.yml` file using `sudo` to apply the settings.



After configuring and running `cmatrix.yml`, execute `cmatrix` to see it in action. Now, add the public IP of the instance to the Ansible configuration, allowing for direct connection. Run the IP with a shell `sudo` command, and when prompted, ensure the connection continues without requiring additional fingerprint verification.

```

ubuntu@ip-172-31-35-204:~$ ansible The_Soepeno_Server -m shell -a 'free | grep Mem:'
localhost | CHANGED | rc=0 >>
Mem:          989392      644264          78604          3652          266524          182928
54.200.43.172 | CHANGED | rc=0 >>
Mem:          989392      644896          80384          3652          264112          182436

```

Finally, to monitor memory and disk usage, run an Ansible command to view a summary of memory output, enable a query in Ansible to check free and used disk space specifically within the home folder.

```

ubuntu@ip-172-31-35-204:~$ df -hT /home
Filesystem      Type  Size  Used Avail Use% Mounted on
/dev/root       ext4   6.8G  3.5G  3.3G  52% /

```

i-07638443c12f1b4c7 (The Soepeno Server)  
PublicIPs: 13.212.32.165 PrivateIPs: 172.31.35.204

### 3. Observation

This lab involves deploying and managing an AWS instance with a focus on configuration automation via Ansible, highlighting several structured key aspects of cloud resource management, security, and automation practices. This demonstrates the benefits of automating cloud infrastructure, from updating instances to verifying configurations. Using commands like `sudo apt-get update` and defining configurations in YAML enables CI/CD pipelines, crucial for agile and DevOps practices. In Ansible and AWS contexts, these practices reduce downtime and ensure system consistency across environments.

During the lab study, there were also a few errors that I faced and noticed on my behalf and I figured out solutions to overcome those errors. At first, I spent hours on the ansible inventory file, finding out what was wrong. Then I remembered that since server files are also named in file form—those used in networking as well—and all spaced names and variables should be written with an underscore, so I did.

Then, there was an error on my behalf, so I overwritten the key pair again, this is the result

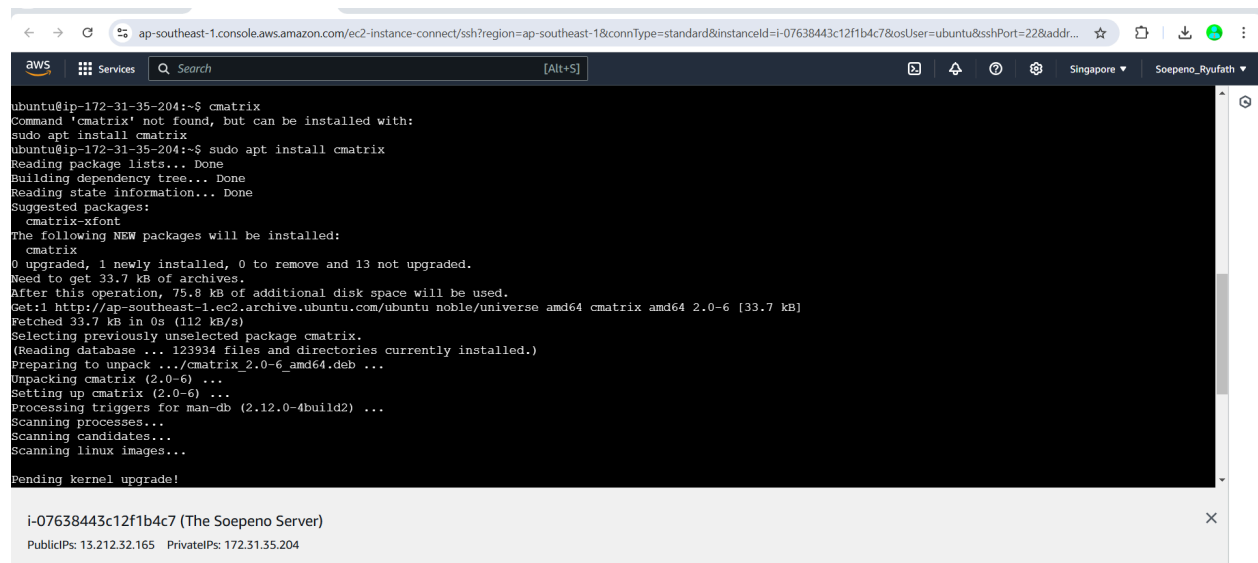
```

ubuntu@ip-172-31-35-204:~/.ssh$ ssh-keygen
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_ed25519):
/home/ubuntu/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_ed25519
Your public key has been saved in /home/ubuntu/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:+sFdCKRp90quSR2RPQ+5N0kyEYxIppblido9IoHUY1Q ubuntu@ip-172-31-35-204
The key's randomart image is:
---[ED25519 256]---
..oo=E oB+.o . |
o 0...=.o=+ |
o *+=o * . |
=. . o o... |
oo o. S o . |
. . * o . |
+ = . |
o + . |
o . |
-----[SHA256]-----
ubuntu@ip-172-31-35-204:~/.ssh$

```

i-07638443c12f1b4c7 (The Soepeno Server)  
PublicIPs: 13.212.32.165 PrivateIPs: 172.31.35.204

Another one was in my AWS instance; it seemed that I was not able to run `cmatrix` immediately, therefore, `'cmatrix'` had to be installed to the instance beforehand.



```

ap-southeast-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-southeast-1&connType=standard&instanceId=i-07638443c12f1b4c7&osUser=ubuntu&sshPort=22&addr...
aws Services Search [Alt+S] Singapore Soepeno_Ryufath
ubuntu@ip-172-31-35-204:~$ cmatrix
Command 'cmatrix' not found, but can be installed with:
sudo apt install cmatrix
ubuntu@ip-172-31-35-204:~$ sudo apt install cmatrix
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  cmatrix-xfont
The following NEW packages will be installed:
  cmatrix
0 upgraded, 1 newly installed, 0 to remove and 13 not upgraded.
Need to get 33.7 kB of archives.
After this operation, 75.8 kB of additional disk space will be used.
Get:1 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 cmatrix amd64 2.0-6 [33.7 kB]
Fetched 33.7 kB in 0s (112 kB/s)
Selecting previously unselected package cmatrix.
(Reading database ... 123934 files and directories currently installed.)
Preparing to unpack .../cmatrix_2.0-6_amd64.deb ...
Unpacking cmatrix (2.0-6) ...
Setting up cmatrix (2.0-6) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning candidates...
Scanning linux images...
Pending kernel upgrade!

i-07638443c12f1b4c7 (The Soepeno Server)
PublicIPs: 13.212.32.165 PrivateIPs: 172.31.35.204

```

## 4. Conclusion

This lab provided foundational knowledge in automating and scaling cloud environments by configuring connectivity, inventory, and YAML-based settings for streamlined management. Through hands-on practice, I learned how automating updates, verifying configurations, and managing resources can improve scalability, consistency, and reduce downtime.