National College of Ireland

Simulation of Compliance-as-Code for Kubernetes Using OPA, Terraform, and Conftest

MSc Research Project
MSc in Cloud Computing

Snehal Prataprao Kolhe
Student ID: x23339438

School of Computing
National College of Ireland

Supervisor: Ahemad Makki

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Snehal Prataprao Kolhe
Student ID:	x23339438
Programme:	MSc in Cloud Computing
Year:	2025
Module:	MSc Research Project
Supervisor:	Ahemad Makki
Submission Due Date:	11/08/2025
Project Title:	Simulation of Compliance-as-Code for Kubernetes Using OPA, Terraform, and Conftest
Word Count:	388
Page Count:	5

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Snehal Prataprao Kolhe
Date:	11th August 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Simulation of Compliance-as-Code for Kubernetes Using OPA, Terraform, and Conftest

Snehal Prataprao Kolhe
x23339438

1 Introduction

This configuration manual outlines the steps to run the research project in the **AWS Academy Learner Lab** environment. The project uses **JupyterLab** for code execution and analysis, a **FastAPI** service on port **8000**, and a **Gradio** interface on port **7861**. **Matplotlib** is used for generating analytical charts and visualisations.

The environment runs on an **AWS EC2 instance** provided by the AWS Academy Learner Lab, with all executions performed inside a pre-configured Python virtual environment located at `source/venv`.

2 Steps to Run the Project

Step 1 — Open AWS academy learner lab and instance

Once the AWS Academy Learner Lab is started, open the instance i.e. 'opa-terraform-simulation' created inside it to connect the DNS

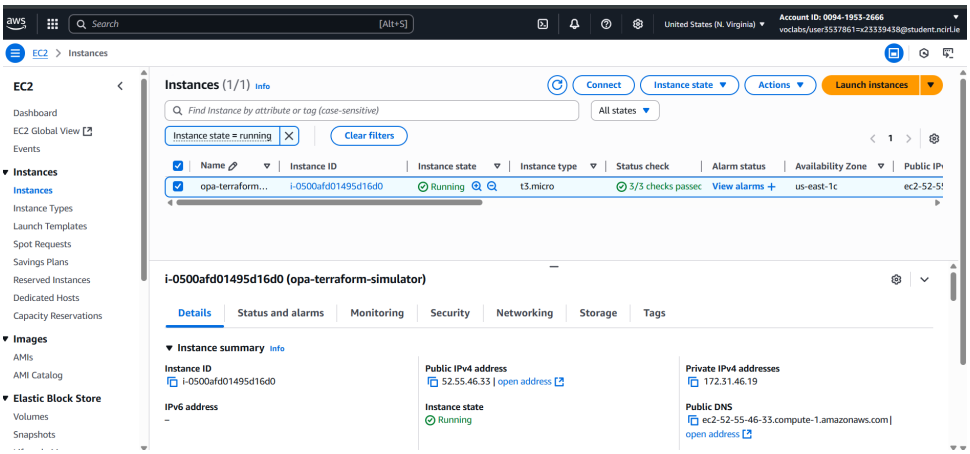

Figure 1: Instance Created in AWS lab

Step 2 — Connect to EC2 Instance

From the instance inside the lab, retrieve the **EC2 public DNS** from the lab dashboard and connect using an SSH client:

```
ssh -i /path/to/key.pem ubuntu@<EC2_PUBLIC_DNS>
```

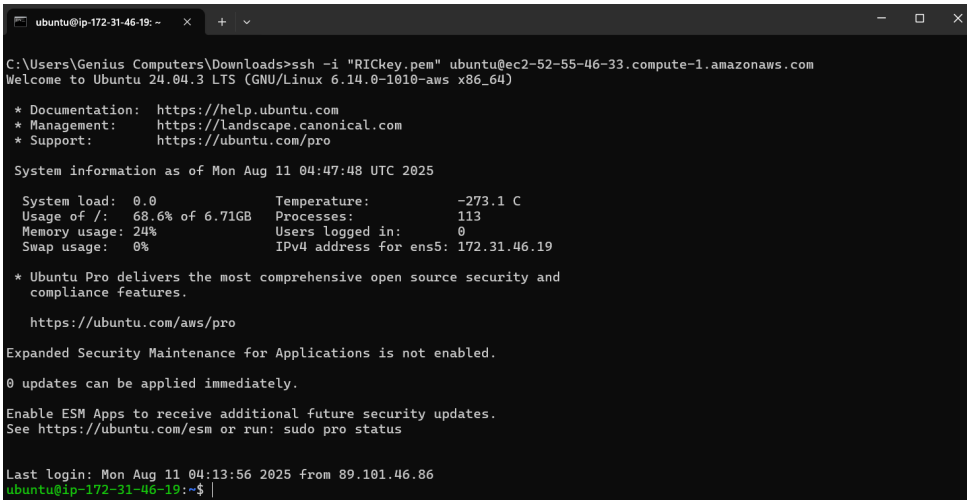

Figure 2: AWS Academy EC2 SSH connection terminal

Step 3 — Activate the Python Virtual Environment

Navigate to the project directory and activate the virtual environment:

```
source source/venv/bin/activate
```

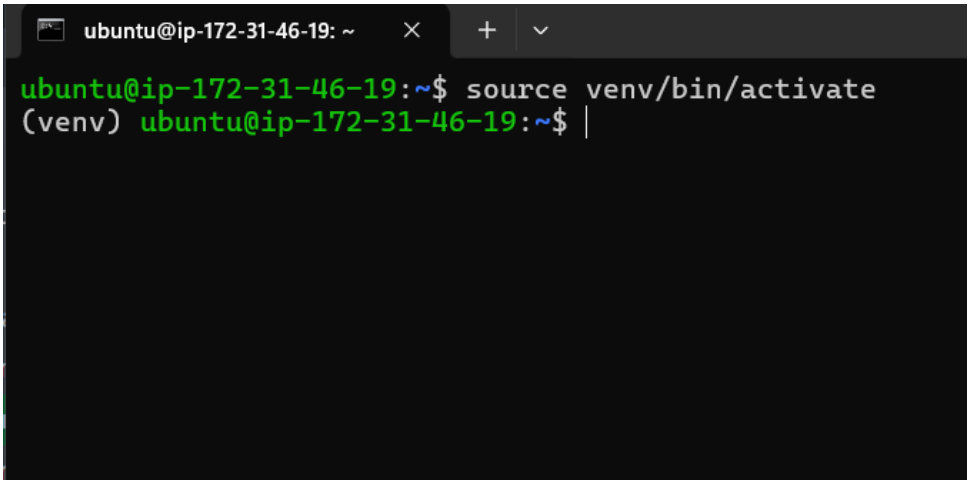

Figure 3: Terminal showing virtual environment activation

Step 4 — Start JupyterLab (Port 8888)

Launch JupyterLab with no browser and no token requirement:

```
jupyter lab --ip=0.0.0.0 --port=8888 --no-browser --NotebookApp.token=''
```

Access it in a browser:

```
http://<EC2_PUBLIC_DNS>:8888/lab
```

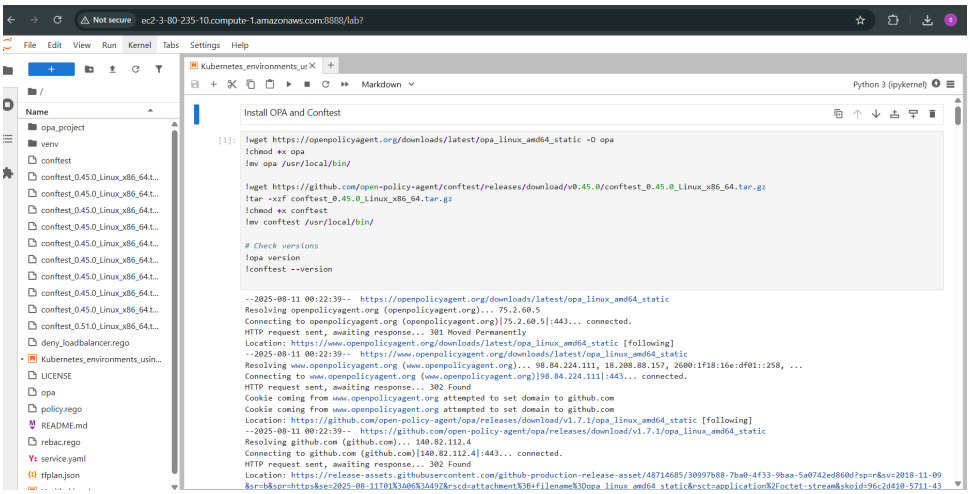

Figure 4: JupyterLab interface in the browser

Step 5 — Running FastAPI and Gradio

When the notebook is executed:

- **FastAPI** will run on port **8000**
  Access via:

  ```
  http://<EC2_PUBLIC_DNS>:8000
  ```

- **Gradio** will run on port **7861**
  Access via:

  ```
  http://<EC2_PUBLIC_DNS>:7861
  ```

Step 6 — Data Analysis with Matplotlib

Matplotlib is used within the notebook to generate visual analysis. When the relevant cells are executed:

- Graphs and charts will be displayed inline in JupyterLab.

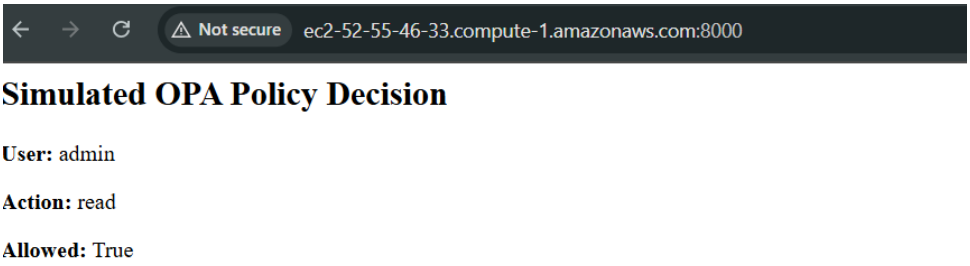

Figure 5: FastAPI Swagger UI on port 8000

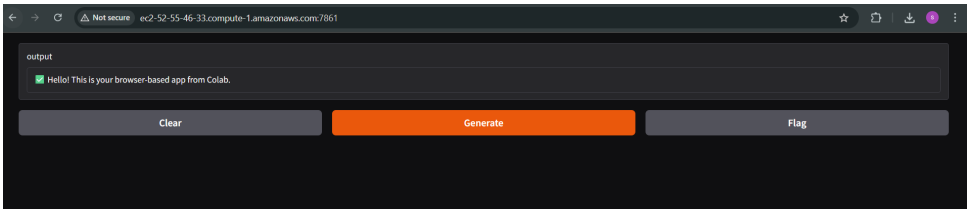

Figure 6: Gradio interface running on port 7861

- Figures can be saved as `.png` or `.jpg` for reports.

Example to save a figure:

```
import matplotlib.pyplot as plt
# plot code here
plt.savefig("analysis_output.png", dpi=150, bbox_inches="tight")
```

3 Summary of Service Ports

Service	Port	Access URL Example
JupyterLab	8888	http://EC2_PUBLIC DNS :8888/lab
FastAPI	8000	http://EC2_PUBLIC DNS :8000
Gradio	7861	http://EC2_PUBLIC DNS :7861

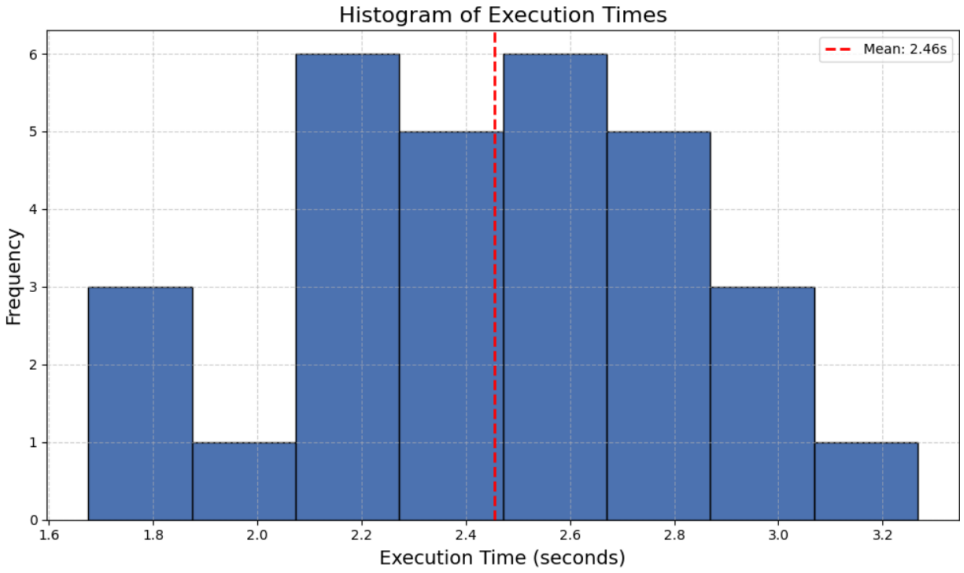

Figure 7: Example Matplotlib analysis chart

FastAPI (n.d.) *JupyterLab Documentation — JupyterLab 4.5.0a1 documentation* (n.d.) *Matplotlib documentation — Matplotlib 3.10.5 documentation* (n.d.) Team (n.d.)

References

FastAPI (n.d.).
URL: *https://fastapi.tiangolo.com/*

JupyterLab Documentation — JupyterLab 4.5.0a1 documentation (n.d.).
URL: *https://jupyterlab.readthedocs.io/en/latest/*

Matplotlib documentation — Matplotlib 3.10.5 documentation (n.d.).
URL: *https://matplotlib.org/stable/index.html*

Team, G. (n.d.). Gradio.
URL: *https://gradio.app*