



Internship Druva

Plan of approach

Bachelor in Electronics-ICT

Graduation subject: Cloud & Cybersecurity

Year 2022-2023

Campus: Thomas More Geel

Milan Sterkens

PREFACE

My internship was at the software company Druva in Letterkenny, Ireland. Formerly called Cloudranger. It was acquired by Druva in 2018. The Cloudranger project offers simple and scalable AWS backup & recovery. It is the main product under development at the Letterkenny site.

The first weeks of the internship taught me a lot about how a modern software-company operates and, specific to Cloudranger, how their cloud-environment works.

As a Cloud & Cybersecurity student, I made it my mission to tighten the gap between developers and operations. Keeping this in mind, I did a couple of things to improve the company's pipeline and keeping their secrets from leaking.

TABLE OF CONTENTS

Preface.....	2
Jest pipeline integration + Grafana dashboard.....	3
Plan of approach.....	3
Lambda function: Secrets & PII Obfuscation.....	4
Plan of approach.....	4
General conclusion.....	5

JEST PIPELINE INTEGRATION + GRAFANA DASHBOARD

The idea is to automatically do a Jest test for the repository CloudRanger_app_2_processing during the Codebuild process so that a notification can be displayed in a channel on Slack. The pipeline should fail if a unit test fails and/or if the coverage percentage went down. The Slack message should contain links to the reports and explain why the build failed. Additionally, we want to have a graphical overview of the reports in Grafana.

This should eventually be applied to all repositories & environments so the code must be flexible.

Plan of approach

This was the first project I've got. I divided the project into smaller, more feasible tasks:

1. Create a Slack bot in a private channel and post a message to the webhook url via curl
2. Execute jest tests locally on code
3. Implement the jest tests into the pipeline
4. send a message to Slack using the pipeline
5. use the jest report to see if tests fail and send different messages to Slack depending on its status
6. do the same thing for code coverage. If the coverage goes down → stop pipeline
7. Install Grafana on an EC2 instance
8. show the jest report and code coverage to the grafana dashboard
9. make everything suitable to work on multiple repositories and environments.
10. Create hyperlinks in Slack message to Grafana dashboard, jest report and code coverage.
11. Create documentation to hand over the project to Druva

LAMBDA FUNCTION: SECRETS & PII OBFUSCATION

For the second project of my internship, my goal was to create a Lambda function that detects secrets and PII (Personal Identifiable Information) in a given string of text to avoid leaks to an external API. After a potential secret or PII is detected, it is replaced with a placeholder to hide the original value.

The company wanted to create a chatbot in Slack that works with OpenAI's API. In order to prevent leaks of sensitive information to the API, I designed this Lambda function to be used as a middleware for the chatbot.

To create the function, I used AWS SAM to easily deploy and make changes to the project. It runs Python code and I wrote my own class to filter PII.

Plan of approach

This was the second project I've got. I divided the project into smaller, more feasible tasks:

1. Do some research about Lambda functions since I've never worked with the technology before.
2. Start coding the Python code locally
3. start with detecting secrets with the detect-secrets Python package from Github
4. use Scrubadub Python package to detect PII*
5. if everything works. Deploy the code in a Lambda function using AWS SAM after researching how it works.
6. Change the code to make it work in a Lambda function

* I had to create my own Python class instead of using the Scrubadub Python package since it uses too much space for my Lambda function (max 250MB)

GENERAL CONCLUSION

I feel like I made a difference at Druva. The unit test integration is going to be implemented in all their repositories and the Lambda function is also running in production. I think that it makes the life of the developers easier & safer. It was also nice to be able to help another intern and that I could help my other colleagues on certain tasks.

I learned a lot about AWS in general. During my internship, I had the opportunity to use AWS courses on Udemy to learn more about the different services they offer. The Codebuild part took most of my time with the Dash script for the buildspec file. The funny thing is that I thought that the Amazon Linux AMI image used Bash. Apparently, it uses Dash which meant I had to rewrite the script to be POSIX compliant.

Apart from the technical side of things, I also got better at speaking English which was one of the reasons for me to choose an internship in Ireland.