Practical: Installing and using the AWS SDK (Python)

Table of Contents

The AWS SDK for Python (`boto3`) is the main way of interacting with AWS services programmatically in Python. It can be used for managing services (e.g., launching EC2 instances) and also for interacting with them (e.g., writing to an S3 bucket). In this practical we'll see how to get started using the SDK and do some basic operations.

Prerequisites and references

- Boto3 documentation (https://boto3.amazonaws.com/v1/documentation/api/latest/index.html)
- Boto3 code samples (https://github.com/boto/boto3)

Creating a basic app

1. Create an EC2 instance

If you already have an EC2 instance up and running that you want to use then you can skip this step.

- Create an EC2 instance and make sure to add the `CAB432-Instance-Role` IAM role to the instance.
- Log in with the Session Manager or SSH to get to the command line for this EC2 instance. If using the Session manager to log in, change to the `ubuntu` user first: `sudo -iu ubuntu`

2. Create a skeleton Python app

- Create a new folder called `sdkdemo` and change to it:

```
mkdir sdkdemo
```

```
cd sdkdemo
```

- (Recommended) Set up a virtual environment:

```
python3 -m venv venv
source venv/bin/activate
```

3. Install the SDK with pip

- Install boto3:

```
pip install boto3
```

4. Create the demo app

We'll create a simple app that grabs the list of all S3 buckets on the QUT AWS account.

- Create `demo.py` with the following contents:

```
import boto3
from botocore.exceptions import NoCredentialsError, ClientError

def main():
    # Creating a client for S3
    client = boto3.client('s3', region_name='ap-southeast-2')
    try:
        response = client.list_buckets()
        print(response)
    except NoCredentialsError:
        print("Credentials not available.")
    except ClientError as e:
        print(f"Client error: {e}")

if __name__ == "__main__":
    main()
```

5. Run the app

- Run the app:

```
python demo.py
```

If all goes well this will print out the response from the SDK, which should include a list of all S3 buckets on the QUT account. If you run into problems with authentication, check that your EC2 instance has the correct IAM role or check out the alternative methods for authentication below.

Authentication

Provided that you have set the IAM role correctly for the EC2 instance, boto3 will automatically fetch the required credentials from the cloud. However, in some situations (for example, developing locally on your own computer) this approach will not work. Additional methods of

authentication are discussed in **Authenticating to the AWS SDK (https://canvas.qut.edu.au/courses/20367/pages/authenticating-to-the-aws-sdk)**