CPD Lab 1 – Introduction to AWS

Introduction

There are many ways to access the Amazon Web Services (AWS). In this lab, you will explore the various methods of interacting with AWS, that are: (1) AWS Management Console (2) AWS Command Line Interface (CLI) (3) AWS Software Development Kit (SDK) for Python-Boto3. It is important to realise that all actions even those through AWS console get executed as API calls to AWS services.

The instructions below are for Windows but the 'links' at the end of this document provide details for other operating systems.

You will also complete a free lab on Qwiklabs (Introduction to AWS Identity and Access Management (IAM)).

Tasks:

- Task 1: Interacting with AWS - Console
- Task 2: Qwiklab-Introduction to Identity and Access Management (IAM)
- Task 3: Creating a User for programmatic access to AWS
- Task 4: Interacting with AWS – CLI
- Task 5: Interacting with AWS – SDK for Python - Boto3
- Task 6: Releasing the allocated resources

Task 1: Interacting with AWS - AWS Management Console

You can access the Management console by logging in at
https://aws.amazon.com/console/

The following helpful links are on the menu bar:

Services:

You can select or search for a particular service and start configuring or creating a service. For example, you can search for IoT under services.

A nice feature is that you can pin the services that you select on to the menu bar for easy access. Simply click the pin, select and drag the service icon to the menu bar.

You can also see the History showing recently used services.

Regions:

The region can be changed but you could use eu-west-2 (London) based on proximity.

Search for IAM under 'Services'. Note that the region changes to 'Global'. This would be the case for services which are not region-specific, such as Identity and Access Management (IAM).

Support:

This drop-down has links to forums, Documentation, and Training besides others.

Hands-on

- Go to service, S3, which is Simple Storage Service
- Click 'create bucket', provide a name e.g. *mybucket* (the name must be globally unique), and click 'create'
- The bucket gets created.
- Click the bucket name, *mybucket*
- Click Upload, and then 'Add files'
- Choose a file and click 'Upload'
- Navigate to *mybucket* and download the file
- Delete the bucket

Task 2: Qwiklab - Introduction to Identity and Access Management (IAM)

Qwiklabs provide you with an access to AWS environment using temporary credentials to learn concepts on a particular cloud platform.
Log in to Qwiklab and search for the above lab title. This lab has an estimated time of 45 minutes and is free. Completing this Qwiklab will help you to complete task 3 below to create users.

Task 3: Creating a User for programmatic access to AWS

The user account that you have used to login to AWS is the root user account with complete access to the resources in your account.
https://docs.aws.amazon.com/IAM/latest/UserGuide/id_root-user.html.

It is therefore a good idea to create IAM users and grant them administrator access rather than use the root account. Programmatic access to AWS services through CLI or SDK requires the use of access keys.

In this task we will create a user with programmatic access, generate and download the access keys. The keys will be used for Task 4 and 5 to access AWS services.

You can also refer to Link 1 under 'Links' below for further details.

Hands-on

1. Create a user with programmatic access on IAM
    - Search for IAM in services
    - Select 'Users' on the dashboard
    - Specify a name, and check 'Programmatic Access' and click next
    - Under 'Set permissions' click 'Attach existing policies directly'
    - Under 'Policy name' select 'Administrator Access' and click next

- Leave ‘Add tags’ for now and click Next
- Click ‘Create User’

2. Download .csv file containing the access keys
    - You can now download the .csv file that contains the access keys and other useful information

Task 4: Interacting with AWS – AWS CLI

You can also refer to Link 2 under ‘Links’ below for further help.
You would need to install the latest version of Python 3.X.X from
https://www.python.org/downloads/

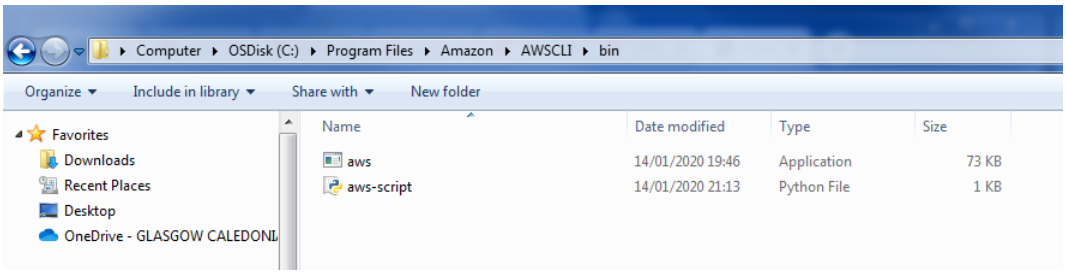
On running the installer, you could customise the installation to set path, download pip etc.

Installation – AWS CLI

On opening Link 2 from under ‘Links‘ below, you can see a link to Windows installer on top right side. Download and run the installer.

You can check the installation by typing aws at the command prompt:

- If the path is not recognised, locate the aws installation, which could be as below:



- Add this to the system path
- Relaunch the command prompt
- The command aws should run

Configuration:

We need to setup the access keys in order to access AWS services

- Run ‘aws configure’ at the command prompt
- In response you will get prompted to enter the access key, secret key, region (eu-west-2 for London) and default output (json)
- Enter the required details from the .csv file downloaded in Task 3
- This completes the configuration, enabling execution of commands from the command prompt

The above process creates two files, `config` and `credentials` in a folder.aws in the home directory:

Hands-on

Elastic Compute Cloud (EC2) is used to create virtual machines or servers, termed instances by AWS. The aim below is to demonstrate the flexibility and strength of CLI in creating an instance. You can also refer to Link 3 under 'Links' below for further help.

1. Running a new EC2 instance on AWS
    - You can use the command below to create a key-pair required for next step

      ```
      >aws ec2 create-key-pair --key-name MyKeyPair
      ```

    - Then by running the following command, an EC2 instance is created and set to running state

      ```
      > aws ec2 run-instances --image-id ami-0089b31e09ac3fffc
      --count 1  --instance-type t2.micro --key-name MyKeyPair
      ```

    - You can check the EC2 instance by navigating to EC2 service. It would be under resources within the 'Running instance'
    - Similarly you can check the key-pair under resources

2. De-allocate resources
    - Terminate the EC2 instance by first selecting it, then by going to

      actions->Instance State->Terminate

      and confirming termination. However note that it will remain visible for some time on the console after being terminated and then the entry gets deleted
    - Delete the key-pair

Task 5: Interacting with AWS – AWS SDK for Python- Boto3

SDKs allow you to write code to interact with AWS. An SDK is helpful to programmatically automate a series of tasks to integrate AWS services.
An example could be to upload an image to S3, write data to DynamoDB etc.

You can also refer to Link 4 under 'Links' below for further help.

Setup:

- At the command prompt, enter pip install boto3
- Verify the installation, aws --version

Running a simple program:

Copy the program below from Boto3 documentation and run using any Python editor of your choice. This program creates a queue. Simple Queue Service (SQS) is used for decoupling the senders and receivers.

https://boto3.amazonaws.com/v1/documentation/api/latest/guide/sqs.html

```
# Get the service resource
sqs = boto3.resource('sqs')

# Create the queue. This returns an SQS.Queue instance
queue = sqs.create_queue(QueueName='test',
Attributes={'DelaySeconds': '5'})

# You can now access identifiers and attributes
print(queue.url)
print(queue.attributes.get('DelaySeconds'))
```

Check that the queue is created through AWS console.

Delete the SQS.

Task 6: Releasing the allocated resources

Remember to:

- Delete the bucket from Task 1
- Delete the key-pair and terminate EC2 instance from Task4
- Delete SQS from Task 5

Links

This lab requires that the students have an account on AWS (Free Tier) and Qwiklabs.

1. Creating an IAM user
   https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users_create.html
2. Installing AWS SDK (CLI) https://aws.amazon.com/cli/
3. Creating an instance on AWS through CLI
   https://docs.aws.amazon.com/cli/latest/userguide/cli-services-ec2-keypairs.html
   https://docs.aws.amazon.com/cli/latest/userguide/cli-services-ec2-instances.html
4. Installing and configuring Boto3
   https://boto3.amazonaws.com/v1/documentation/api/latest/guide/quickstart.html