Show Submission Credentials

P0. Amazon Web Services APIs Amazon Web Services APIs

✔ Amazon Web Services API

✔ AWS SDK for Java

✔ AWS SDK for Python

Amazon Web Services API

Using Amazon Web Service APIs to create EC2 Instance

APIs and SDK access for AWS

The Amazon Web Services SDK includes different kinds of API packages which developers can use to create and manage resources and applications running on AWS. This allows developers to programmatically automate management tasks on AWS. Most functionality in AWS is accessible via the AWS Command Line Interface (CLI), APIs or SDK. We will go through some of the most popular AWS tools now.

AWS SDK for Java

AWS SDK for Java

The AWS SDK is available for multiple languages, including, Java, .NET, node.js, Ruby etc. The AWS SDK for Java (http://aws.amazon.com/sdk-for-java/) has been developed by Amazon and is a fully featured SDK, which includes:

- **The AWS Java Library:** Java packages, classes and methods (i.e. an API) that allow you to program AWS using Java. The API hides much of the complexity that is otherwise involved in using a REST/SOAP-based HTTP interface including authentication, request retries and error handling.
- **Code Samples:** Code samples that demonstrate the use of an API to perform various tasks on AWS.
- **IDE support:** Includes plugins for common IDEs like IntelliJ, Visual Studio Code and Eclispe that allow developers to create Java apps that work with AWS from within the IDE.

The following video provides a brief introduction to AWS SDKs as well as an example of the Java SDK:

Video 1: Introduction to AWS SDKs and Java SDK Example

Amazon EC2 Java API

You are already quite familiar with Amazon EC2 through the AWS Management Console. The AWS SDK exposes the EC2 management functionality provided by the AWS Management Console and the command line tools via programming interfaces in different programming languages. You can use the EC2 API and methods to:

- Create an EC2 key pair
- Create Security Groups for instances and open ports (also known as authorize security group ingress)
- Create, launch, stop, reboot and terminate instances on EC2.

We provide you with a worked example Maven project to provision EC2 instances with Java.

Information

The example code assumes that you have configured your AWS credentials in a manner supported by DefaultAWSCredentialsProviderChain (http://docs.aws.amazon.com/AWSJavaSDK/latest/javadoc/com/amazonaws/auth/DefaultAWSCredentialsProviderChain.html) and have already created an EC2 keypair and security group. Make sure to never expose your credentials in any code segment. Never put your credentials on any public resource such as Git or others.

If you are running this sample project on your local machine, ensure that you have configured **your AWS credentials** and **the default region name** to `us-east-1` by running `aws configure`.

Replace `KEY_NAME` and `SECURITY_GROUP` with your key pair name and existing security group name.

Many of the Java samples in this course will be provided as Maven projects; you should review the Maven primer to learn more about dependency management and project development with Maven. Running the Maven commands the first time will download the artifacts required by the project. These will **NOT** have to be downloaded every time.

To run this sample, execute the following commands:

```
$ mkdir aws_intro && cd aws_intro

$ wget https://clouddeveloper.blob.core.windows.net/aws-apis/aws-apis-vm-java.tgz -O aws-apis-vm
-java.tgz

$ tar -xvzf aws-apis-vm-java.tgz

$ export KEY_NAME='<your_key_pair>'
$ export SECURITY_GROUP_NAME='<your_security_group_name>'

$ mvn compile && mvn exec:java -Dexec.mainClass="edu.cmu.cs.cloud.samples.aws.LaunchEC2Instance"
 ...
[INFO] --- exec-maven-plugin:1.6.0:java (default-cli) @ aws-apis-samples ---
Launched instance with Instance Id: [i-0b8c7ed1909c63a90]!
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
...

$ ssh -i "your_key_pair.pem" ubuntu@<vm-public-dns>
ubuntu@ip-172-31-9-237:~$
```

In addition, the Amazon EC2 API provides classes such as `DescribeInstanceStatusRequest` to make requests that check an instance's status:

- Running
- Pending
- Shutting Down, etc...

AWS provides many utility classes to support common functionality. We suggest you review the AmazonEC2Waiters (http://docs.aws.amazon.com/AWSJavaSDK/latest/javadoc/com/amazonaws/services/ec2/waiters/AmazonEC2Waiters.html) and other waiter classes to help with event polling. Waiters make it easier to wait for a resource transition into a desired state in tasks that are eventually consistent

For more information on the Amazon EC2 API in Java, please refer to the AWS SDK for Java Documentation (http://aws.amazon.com/documentation/sdk-for-java/) and the AWS Java API Reference (http://docs.aws.amazon.com/AWSJavaSDK/latest/javadoc/index.html)

AWS SDK for Python

AWS SDK for Python (Boto3)

For Python developers, AWS supports a third-party SDK called boto3 (https://boto3.readthedocs.io/en/latest/), which can be used to make API requests to AWS from within Python. To install boto on the your machine, follow the steps outlined in the Boto3 Quickstart (https://boto3.readthedocs.io/en/latest/guide/quickstart.html). The following video provides more information about boto:

Video 2: AWS Boto3 SDK for python

Amazon EC2 Python API

You are already quite familiar with Amazon EC2 through the AWS Management Console. The AWS SDK exposes the EC2 management functionality provided by the AWS Management Console and the command line tools via programming interfaces in different programming languages. You can use the EC2 API and methods to:

- Create an EC2 key pair
- Create Security Groups for instances and open ports (also known as authorize security group ingress)
- Create, launch, stop, reboot and terminate instances on EC2.

We provide you with a worked example to provision EC2 instances with Python.

Information

Note: The snippets below assume that you have configured your AWS credentials in accordance with the Boto3 configuration (http://boto3.readthedocs.io/en/latest/guide/configuration.html) and have already created an EC2 keypair and security group. Make sure to never expose your credentials in any code segment. Never put your credentials on any public resource such as Git or others.

If you are running this sample project on your local machine, ensure that you have configured **your AWS credentials** and **the default region name** to `us-east-1` by running `aws configure`.

Replace `KEY_NAME` and `SECURITY_GROUP` with your key pair name and existing security group name.

Note that you should execute most Python examples in this course with Python 3 and a virtualenv (https://virtualenv.pypa.io/en/stable/). Virtualenv is a project that enables creation of isolated Python environments.

To run the example code, execute the following steps:

```
$ wget https://clouddeveloper.blob.core.windows.net/aws-apis/aws-apis-vm-python.tgz -O aws-apis-vm-python.tgz

$ tar -xvzf aws-apis-vm-python.tgz

$ export KEY_NAME='<your_key_pair>'
$ export SECURITY_GROUP_NAME='<your_security_group_name>'

$ virtualenv env

$ source env/bin/activate

(env)$ pip install -r requirements.txt

(env)$ python launch_ec2_instance.py
Launched instance with Instance Id: [i-0f227cdd6a25bfa17]!

$ ssh -i "your_key_pair.pem" ubuntu@<vm-public-dns>
ubuntu@ip-172-31-9-237:~$
```

The code provided above does not wait for the instance to enter the `Ready` state. Boto provides Waiters (http://boto3.readthedocs.io/en/latest/guide/resources.html#waiters) that simplify the process of polling for asynchronous events to complete.

In addition, the Amazon EC2 API provides classes such as `DescribeInstanceStatusRequest` to make requests that check an instance's status:

- Running
- Pending
- Shutting Down, etc...

For more information on the Amazon SDK in python, please refer to the AWS SDK for Python documentation (https://aws.amazon.com/sdk-for-python/) and the Boto3 (https://boto3.readthedocs.io/en/latest/index.html) project.