Exercise 2: How to integrate with Amazon S3 service

How to Migrate your Integration Engine to the Cloud, IUC2019

Amandeep Aujla

Integration Engineer

A. Table of Contents

A. Introduction

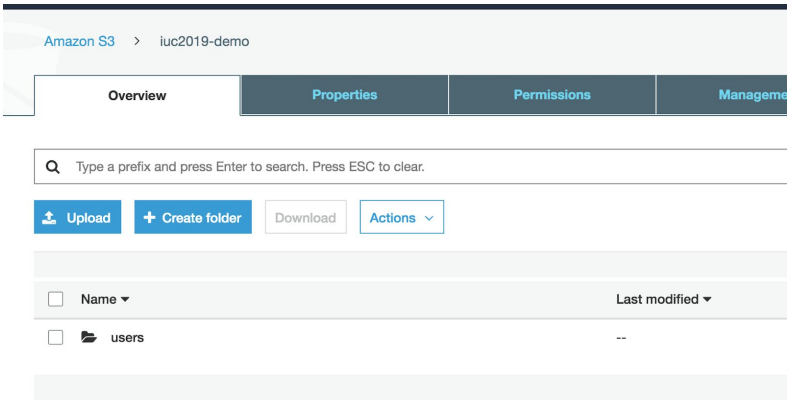
The **AWS S3 Adapter** module allows seamless integration with increasingly popular AWS S3 service. This module is designed to post files into an S3 bucket and to read the content of previously stored files.

Reference: : https://help.interfaceware.com/v6/aws-s3-adapter

B. Background

To work with AWS S3, following steps have already been done prior to demo which includes creating a user in AWS console to access bucket. S3 service parameters in aws/configuration.lua are pre-configured to reflect bucket properties: **bucket name**, **region**, **access Key ID**, **access Key**

1. Create an IAM user and assign the "AmazonS3FullAccess" permission
    - https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users_manage.html?icmpid=docs_iam_console
2. Create IAM user Access Key and update "Access Key" and "Access Key ID" in aws/configuration.lua
3. Create S3 bucket in AWS and update S3 bucket name and region in aws/configuration.lua
    - https://docs.aws.amazon.com/AmazonS3/latest/gsg/CreatingABucket.html



C. Exercise

1. Sign in AWS Console:

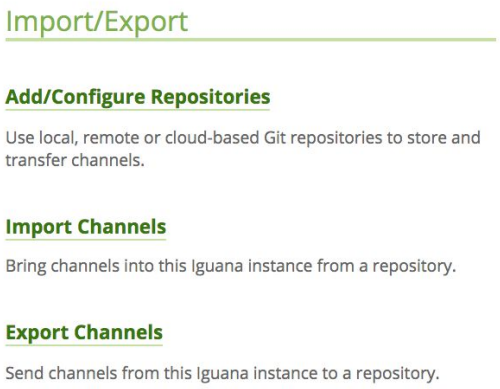
   https://aws.amazon.com/console/

2. Use following credentials to sign-in
   a. Account Id: <your account id>
   b. Username: <your username>
   c. Password:<your password>

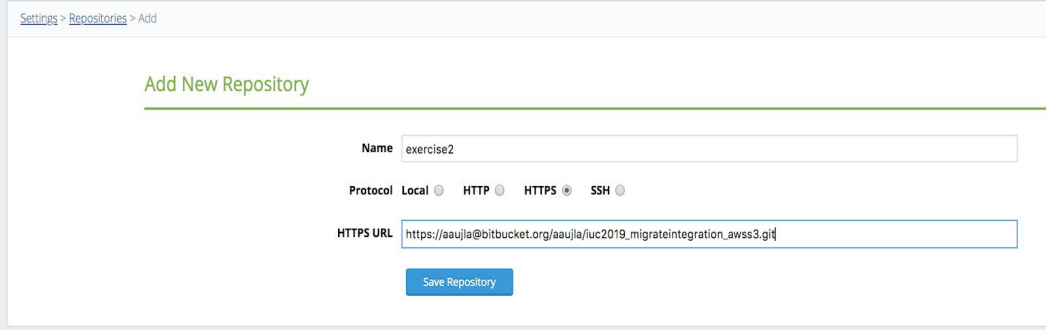
3. Import code in Iguana from following repository

   Link to repo: https://github.com/InterfacewareCS/IUC2019_Exercises.git

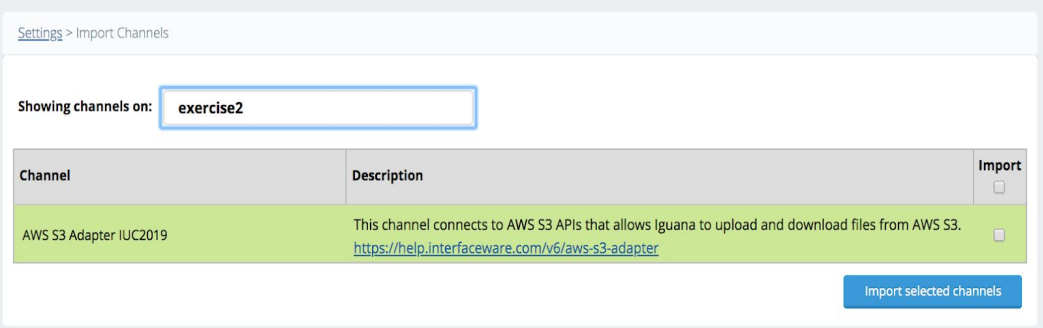
   - Go to Iguana > Settings > Add/Configure Repositories



- Click on new Repository
- Go back to Settings > Import channels



- Select the repository that we just created

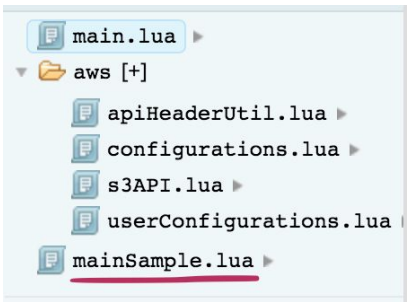


- Select the channel from the checkbox and then click on import
- Enter any message in commit dialogue and press OK

4. Navigate to channel **AWS S3 Adaptor IUC2019**

   and click on **Trans** component of channel and click on Edit Script

5. Start writing code in main.lua to connect connect to Amazon S3 bucket. Note: *Sample code is present in mainSample.lua*.



6. Define S3 bucket absolute file path - path of your bucket to store data.

```
-- 4) Define S3 bucket absolute file path
local fileName = 'yourfirstname'
local folderName = 'yourlastname'
local canonicalendpoint = '/'..folderName..'/'..fileName..'.txt'
```

7. Call s3 API to upload a file

```
-- 5) Put data in S3
s3API.uploadFile(Data, canonicalendpoint)
```

8. Go back to AWS console and a new file with name**[*fileName]*** should be present. Congratulations, you have successfully uploaded files into AWS S3 bucket.
9. [This is optional] Call s3 API in translator to read the contents of file in Iguana

```
-- 6) Get data in S3
local file = s3API.readFile(canonicalendpoint)
```

10. [This is optional] Call s3 API in translator to delete recently uploaded file in Iguana

```
-- 7) Delete data in S3
s3API.deleteFile(canonicalendpoint)
```

Hope you had fun integrating with S3 - Amandeep Aujla