Serverless Application Model (SAM) Tutorial

Setup

- Install the AWS SAM CLI (we recommend the Command line install for "all users" for your particular Operating System)
- Harvest your AWS credentials from the Learner Lab and store in ~/.aws/credentials
- Install the AWS SAM Toolkit plugin for Visual Studio

Instructions

- Create a new folder in your local machine, open a terminal window, and navigate to your new folder.
- Create a new SAM app by typing: sam init
    - Select "2" for "Quick start template"
    - Select "1" for "Hello World Example"
    - "y" for Python and Zip
    - X-Ray = "n"
    - CloudWatch = "y"
    - Structured logging = "y"
    - Name your project like "hw11-..."
- Edit your template.yaml
    - Lines 57-60 (HelloWorldFunctionIamRole): comment these lines out
    - Lines 41-46 (ApplicationInsightsMonitoring): comment these lines out
    - Around line 24, add the following line under Resources-HelloWorldFunction-Properties: (Replace XXXXXXX with your AWS account number.)
      Role: arn:aws:iam::XXXXXXXXXXX:role/LabRole
    - (Note: we do not want a new role created because the Learner Lab does not allow for the creation of new IAM roles.)
    - Validate that your template.yaml is structurally correct: sam validate
- Build your app: sam build
- Deploy your app: sam deploy --guided
    - Accept most of the defaults by simply pressing the <Enter> key. But select "n" when prompted about creating an IAM role.
- If you receive no errors, say "Y" when prompted if you want to "deploy this changeset".
    - If you get errors (red font) try fixing the issue(s). If you are unable to deploy the new changeset after updating the template, try using the delete command listed in the table below.
- Obtain the URL for your new API: sam list endpoints --output json
- Copy/paste the url into a browser. (Or use the curl command.) If you see the message "hello world", it worked!
    - Note: you need the CloudEndpoint url with the Method url at the end. The full url should look something like this:
      https://dhdjkd3h3hd.execute-api.us-east-1.amazonaws.com/Prod**/hello**

SAM CLI Command	Description/Purpose
sam init	Create a new SAM app config
sam build	Build your app
sam validate	Validate your IAC template
sam local invoke	Invoke your lambda function locally. (Be sure to launch your local Docker before trying this command.)
sam sync --stack-name {{stack-name}} --watch	Invoke function in the cloud
sam deploy --guided	Deploy your app to the cloud. (guided switch is optional.)
sam list endpoints --output json	List all the various endpoints that were created by your IAC stack in AWS. These endpoints are important when you want to test your deployment in AWS.
sam delete --stack-name <stack name>	Delete your stack on AWS and all the resources it created.

- Test you app locally
    - Run your lambda function
        - Make sure you cd into the local folder that contains your hello-world stack
        - sam remote invoke HelloWorldFunction --stack-name <your stack name>
    -
- Git
    - Edit the .gitignore file in the subfolder that contains the hello-world SAM app you created. You need to tell git to ignore the /deps folder. Add the following at the end of .gitignore:
      /deps/
    - When you do a "git status", you should NOT see any files in that /deps folder appear in any list.

Other tutorials

- AWS SAM hello-world
- Using the SAM CLI
- SAM template snippets