Amazon Web Services (AWS) Boto3 Client Service Communications

Fan Wang

2020-09-13

Contents

1 AWS Boto3

Go to the **RMD**, **PDF**, or **HTML** version of this file. Go back to Python Code Examples Repository (bookdown site) or the pyfan Package (API).

1.1 Basics

Create local .aws folder under user for example that has credential information, this will be useful for AWS command line operations.

```
# IN C:\Users\fan\.aws
# config file
[default]
region = us-east-1
output = json
# credentials file
[default]
aws_access_key_id = XKIXXXGSXXXBZXX43XXX
aws_secret_access_key = xxTgp9r0f4XXXXXXX1XXlG1vTy07wydxXXXXXX11
```

Additionally, or alternatively, for boto3 operations, store in for example a yml file, so that appropriate value could be obtained.

```
- main_aws_id: 710673677961,
  aws_access_key_id: XKIXXXGSXXXBZXX43XXX
  aws_secret_access_key: xxTgp9r0f4XXXXXXX1XXlG1vTy07wydxXXXXXX11
  region: us-east-1
  main_ec2_instance_id: i-YYYxYYYYYYx2619xx
  main_ec2_linux_ami: ami-0xYYYYYxx95x71x9
  main_ec2_public_subnet: subnet-d9xxxxYY
  fargate_vpc_name: FanCluster
  fargate_vpc_id: vpc-xxx5xYYY
  fargate_public_subnet: subnet-e3dYYYxx
  fargate_security_group: sg-17xxxxYx
  fargate_task_executionRoleArn: ecsTaskExecutionRole
```

```
  batch_task_executionRoleArn: ecsExecutionRole
  fargate_route_table: rtb-5xxxYx25
  date_start: 20180701
```

1.2 Start Client Service

For the various AWS services, could use Boto3 to access and use programmatically. To use any particular service, first start the client for that service: boto3 client.

We load AWS access key and secret acess key etc in from a yaml file to start boto3 client. We then start the client for AWS Batch. And then describe a compute environment.

```
import boto3
import yaml
import pprint

# Load YAML file
son_aws_yml = "C:/Users/fan/fanwangecon.github.io/_data/aws.yml"
fl_yaml = open(son_aws_yml)
ls_dict_yml = yaml.load(fl_yaml, Loader=yaml.BaseLoader)
# Get the first element of the yml list of dicts
aws_yml_dict_yml = ls_dict_yml[0]

# Use AWS Personal Access Keys etc to start boto3 client
aws_batch = boto3.client('batch',
  aws_access_key_id=aws_yml_dict_yml['aws_access_key_id'],
  aws_secret_access_key=aws_yml_dict_yml['aws_secret_access_key'],
  region_name=aws_yml_dict_yml['region'])

# Show a compute environment Delete some Personal Information
ob_response = aws_batch.describe_compute_environments(computeEnvironments=["SpotEnv2560"])
ob_response['ResponseMetadata'] = ''
ob_response['computeEnvironments'][0]['ecsClusterArn'] = ''
ob_response['computeEnvironments'][0]['serviceRole'] = ''
ob_response['computeEnvironments'][0]['computeResources']['instanceRole'] = ''
pprint.pprint(ob_response, width=1)
```

```
## {'ResponseMetadata': '',
##  'computeEnvironments': [{'computeEnvironmentArn': 'arn:aws:batch:us-east-1:710673677961:compute-envi
##                           'computeEnvironmentName': 'SpotEnv2560',
##                           'computeResources': {'desiredvCpus': 0,
##                                                'ec2KeyPair': 'fan_wang-key-pair-us_east_nv',
##                                                'instanceRole': '',
##                                                'instanceTypes': ['optimal'],
##                                                'maxvCpus': 2560,
##                                                'minvCpus': 0,
##                                                'securityGroupIds': ['sg-e6642399'],
##                                                'spotIamFleetRole': 'arn:aws:iam::710673677961:role/A
##                                                'subnets': ['subnet-d9abbe82'],
##                                                'tags': {},
##                                                'type': 'SPOT'},
##                           'ecsClusterArn': '',
##                           'serviceRole': '',
##                           'state': 'ENABLED',
##                           'status': 'VALID',
```

```
##                           'statusReason': 'ComputeEnvironment '
##                                           'Healthy',
##                           'tags': {},
##                           'type': 'MANAGED'}]}
```