cfm-reslib

Release 0.1

CloudSnorkel

Mar 26, 2021

CONTENTS:

CloudFormation Resource Library: a collection of useful custom resources that are missing from CloudFormation.

CHAPTER

ONE

INSTRUCTIONS

1.1 Installation

cfm-reslib is delivered as a single CloudFormation template that exports a single output called `cfm-reslib`. To use it you must first install it in the account and region where it will be used.

1.1.1 Install

Installation is a simple one-liner. Make sure you have AWS CLI installed and configured.

```
aws cloudformation create-stack --stack-name cfm-reslib --template-url https://s3.
↪amazonaws.com/cfm-reslib/cfm-reslib-latest.template --capabilities CAPABILITY_IAM
```

You can also download the template and manually install it using AWS Console.

1.1.2 Update

If you've already installed this library before, you need to run a different command to update to the latest version.

```
aws cloudformation update-stack --stack-name cfm-reslib --template-url https://s3.
↪amazonaws.com/cfm-reslib/cfm-reslib-latest.template --capabilities CAPABILITY_IAM
```

1.2 Usage

Once installed cfm-reslib can be used by defining a custom resource with `ServiceToken` set to the exported value.

See *Available Custom Resources* for a list of supported custom resource types.

1.2.1 YAML

```
Resources:
  SomeCustomResource:
    Type: Custom::SomeCustomResourceType
    Properties:
      ServiceToken: !ImportValue cfm-reslib
      SomeParameter: some value
```

1.2.2 JSON

```
{
  "Resources": {
    "SomeCustomResource": {
      "Type": "Custom::SomeCustomResourceType",
      "Properties": {
        "ServiceToken": {"Fn::ImportValue": "cfm-reslib"},
        "SomeParameter": "some value"
      }
    }
  }
}
```

CHAPTER

TWO

AVAILABLE CUSTOM RESOURCES

2.1 Custom::ElasticTranscoderPipeline

The `Custom::ElasticTranscoderPipeline` resource creates an Elastic Transcoder pipeline.

2.1.1 Syntax

JSON

```
{
  "Type" : "Custom::ElasticTranscoderPipeline",
  "Properties" : {
    "ServiceToken" : {"Fn::ImportValue": "cfm-reslib"},
    "Name" : string,
    "InputBucket" : string,
    "OutputBucket" : string,
    "Role" : string,
    "AwsKmsKeyArn" : string,
    "Notifications" : Notifications,
    "ContentConfig" : PipelineOutputConfig,
    "ThumbnailConfig" : PipelineOutputConfig
  }
}
```

YAML

```
Type: Custom::ElasticTranscoderPipeline
Properties :
  ServiceToken : !ImportValue cfm-reslib
  Name : string
  InputBucket : string
  OutputBucket : string
  Role : string
  AwsKmsKeyArn : string
  Notifications :
    Notifications
  ContentConfig :
    PipelineOutputConfig
  ThumbnailConfig :
```

PipelineOutputConfig

2.1.2 Properties

Name

Required: Yes

Type: string

Update requires: No interruption

InputBucket

Required: Yes

Type: string

Update requires: No interruption

OutputBucket

Required: Yes

Type: string

Update requires: Replacement

Role

Required: Yes

Type: string

Update requires: No interruption

AwsKmsKeyArn

Required: Yes

Type: string

Update requires: No interruption

Notifications

Required: Yes

Type: *Notifications*

Update requires: No interruption

ContentConfig

Required: Yes

Type: *PipelineOutputConfig*

Update requires: No interruption

ThumbnailConfig

Required: Yes

Type: *PipelineOutputConfig*

Update requires: No interruption

Notifications

Syntax

JSON

```
{
  "Progressing" : string,
  "Completed" : string,
  "Warning" : string,
  "Error" : string
}
```

YAML

```
Progressing : string
Completed : string
Warning : string
Error : string
```

Properties

Progressing

Required: No

Type: string

Update requires: No interruption

Completed

Required: No

Type: string

Update requires: No interruption

Warning

Required: No

Type: string

Update requires: No interruption

Error

Required: No

Type: string

Update requires: No interruption

PipelineOutputConfig

Syntax

JSON

```
{
  "Bucket" : string,
  "StorageClass" : string,
  "Permissions" : [ Permission, ... ]
}
```

YAML

```
Bucket : string
StorageClass : string
Permissions :
  - Permission
```

Properties

Bucket

Required: No

Type: string

Update requires: No interruption

StorageClass

Required: No

Type: string

Update requires: No interruption

Permissions

Required: No

Type: List of *Permission*

Update requires: No interruption

Permission

Syntax

JSON

```
{
  "GranteeType" : string,
  "Grantee" : string,
  "Access" : [ string, ... ]
}
```

YAML

```
GranteeType : string
Grantee : string
Access :
  - string
```

Properties

GranteeType

Required: No

Type: string

Update requires: No interruption

Grantee

Required: No

Type: string

Update requires: No interruption

Access

Required: No

Type: List of string

Update requires: No interruption

2.2 Custom::FindAMI

The `Custom::FindAMI` resource finds an AMI by owner, name and architecture. The result can thenbe used with Ref

2.2.1 Syntax

JSON

```
{
  "Type" : "Custom::FindAMI",
  "Properties" : {
    "ServiceToken" : {"Fn::ImportValue": "cfm-reslib"},
    "Owner" : string,
    "Name" : string,
    "Architecture" : string
  }
}
```

YAML

```
Type: Custom::FindAMI
Properties :
  ServiceToken : !ImportValue cfm-reslib
  Owner : string
  Name : string
  Architecture : string
```

2.2.2 Properties

Owner

Required: No

Type: string

Update requires: Replacement

Name

Required: No

Type: string

Update requires: Replacement

Architecture

Required: No

Type: string

Update requires: Replacement

2.2.3 Examples

Create EC2 Instance With Latest Ubuntu

The following example searches for the latest version of Ubuntu 16.04 AMI and creates a newEC2 instance with this image.

JSON

```
{
  "UbuntuAMI": {
    "Type": "Custom::FindAMI",
    "Properties": {
      "ServiceToken": {
        "Fn::ImportValue": "cfm-reslib"
      },
      "Owner": "099720109477",
      "Name": "ubuntu/images/hvm-ssd/ubuntu-xenial-16.04*",
      "Architecture": "x86_64"
    }
  },
  "UbuntuInstance": {
    "Type": "AWS::EC2::Instance",
    "Properties": {
      "InstanceType": "t2.micro",
      "ImageId": {
        "Ref": "UbuntuAMI"
      }
    }
  }
}
```

YAML

```
UbuntuAMI:
  Properties:
    Architecture: x86_64
    Name: ubuntu/images/hvm-ssd/ubuntu-xenial-16.04*
    Owner: "099720109477"
    ServiceToken:
      Fn::ImportValue: cfm-reslib
  Type: Custom::FindAMI
UbuntuInstance:
```

(continues on next page)

(continued from previous page)

```
  Properties:
    ImageId:
      Ref: UbuntuAMI
    InstanceType: t2.micro
  Type: AWS::EC2::Instance
```

2.3 Custom::KafkaCluster

The `Custom::KafkaCluster` resource creates a Kafka Cluster (MSK). Now offificially available in CloudFormation with AWS::MSK::Cluster.

2.3.1 Syntax

JSON

```
{
  "Type" : "Custom::KafkaCluster",
  "Properties" : {
    "ServiceToken" : {"Fn::ImportValue": "cfm-reslib"},
    "BrokerNodeGroupInfo" : BrokerNodeGroupInfo,
    "ClientAuthentication" : ClientAuthentication,
    "ClusterName" : string,
    "ConfigurationInfo" : ConfigurationInfo,
    "EncryptionInfo" : EncryptionInfo,
    "EnhancedMonitoring" : string,
    "OpenMonitoring" : OpenMonitoringInfo,
    "KafkaVersion" : string,
    "LoggingInfo" : LoggingInfo,
    "NumberOfBrokerNodes" : integer,
    "Tags" : map
  }
}
```

YAML

```
Type: Custom::KafkaCluster
Properties :
  ServiceToken : !ImportValue cfm-reslib
  BrokerNodeGroupInfo :
    BrokerNodeGroupInfo
  ClientAuthentication :
    ClientAuthentication
  ClusterName : string
  ConfigurationInfo :
    ConfigurationInfo
  EncryptionInfo :
    EncryptionInfo
  EnhancedMonitoring : string
  OpenMonitoring :
```

```
    OpenMonitoringInfo
  KafkaVersion : string
  LoggingInfo :
    LoggingInfo
  NumberOfBrokerNodes : integer
  Tags : map
```

2.3.2 Properties

BrokerNodeGroupInfo

Required: Yes

Type: *BrokerNodeGroupInfo*

Update requires: Replacement

ClientAuthentication

Required: Yes

Type: *ClientAuthentication*

Update requires: Replacement

ClusterName

Required: Yes

Type: string

Update requires: Replacement

ConfigurationInfo

Required: Yes

Type: *ConfigurationInfo*

Update requires: Replacement

EncryptionInfo

Required: Yes

Type: *EncryptionInfo*

Update requires: Replacement

EnhancedMonitoring

Required: Yes

Type: string

Update requires: Replacement

OpenMonitoring

Required: Yes

Type: *OpenMonitoringInfo*

Update requires: Replacement

KafkaVersion

Required: Yes

Type: string

Update requires: Replacement

LoggingInfo

|desc51|

Required: Yes

Type: *LoggingInfo*

Update requires: Replacement

NumberOfBrokerNodes

Required: Yes

Type: integer

Update requires: Replacement

Tags

Required: Yes

Type: map

Update requires: Replacement

BrokerNodeGroupInfo

Syntax

JSON

```
{
  "BrokerAZDistribution" : string,
  "ClientSubnets" : [ string, ... ],
  "InstanceType" : string,
  "SecurityGroups" : [ string, ... ],
  "StorageInfo" : StorageInfo
}
```

YAML

```
BrokerAZDistribution : string
ClientSubnets :
  - string
InstanceType : string
SecurityGroups :
  - string
StorageInfo :
  StorageInfo
```

Properties

BrokerAZDistribution

Required: Yes

Type: string

Update requires: No interruption

ClientSubnets

Required: Yes

Type: List of string

Update requires: No interruption

InstanceType

Required: Yes

Type: string

Update requires: No interruption

SecurityGroups

Required: Yes

Type: List of string

Update requires: No interruption

StorageInfo

Required: Yes

Type: *StorageInfo*

Update requires: No interruption

StorageInfo

Syntax

JSON

```
{
  "EbsStorageInfo" : EBSStorageInfo
}
```

YAML

```
EbsStorageInfo :
  EBSStorageInfo
```

Properties

EbsStorageInfo

Required: No

Type: *EBSStorageInfo*

Update requires: No interruption

EBSStorageInfo

Syntax

JSON

```
{
  "VolumeSize" : integer
}
```

YAML

```
VolumeSize : integer
```

Properties

VolumeSize

Required: No

Type: integer

Update requires: No interruption

ClientAuthentication

Syntax

JSON

```
{
  "Sasl" : Sasl,
  "Tls" : Tls
}
```

YAML

```
Sasl :
  Sasl
Tls :
  Tls
```

Properties

Sasl

Required: No

Type: *Sasl*

Update requires: No interruption

Tls

Required: No

Type: *Tls*

Update requires: No interruption

Sasl

Syntax

JSON

```
{
  "Scram" : Scram
}
```

YAML

```
Scram :
  Scram
```

Properties

Scram

Required: No

Type: *Scram*

Update requires: No interruption

Scram

Syntax

JSON

```
{
  "Enabled" : boolean
}
```

YAML

```
Enabled : boolean
```

Properties

Enabled

Required: No

Type: boolean

Update requires: No interruption

Tls

Syntax

JSON

```
{
  "CertificateAuthorityArnList" : [ string, ... ]
}
```

YAML

```
CertificateAuthorityArnList :
  - string
```

Properties

CertificateAuthorityArnList

Required: No

Type: List of string

Update requires: No interruption

ConfigurationInfo

Syntax

JSON

```
{
  "Arn" : string,
  "Revision" : long
}
```

YAML

```
Arn : string
Revision : long
```

Properties

Arn

Required: Yes

Type: string

Update requires: No interruption

Revision

Required: Yes

Type: long

Update requires: No interruption

EncryptionInfo

Syntax

JSON

```
{
  "EncryptionAtRest" : EncryptionAtRest,
  "EncryptionInTransit" : EncryptionInTransit
}
```

YAML

```
EncryptionAtRest :
  EncryptionAtRest
EncryptionInTransit :
  EncryptionInTransit
```

Properties

EncryptionAtRest

Required: No

Type: *EncryptionAtRest*

Update requires: No interruption

EncryptionInTransit

Required: No

Type: *EncryptionInTransit*

Update requires: No interruption

EncryptionAtRest

Syntax

JSON

```
{
  "DataVolumeKMSKeyId" : string
}
```

YAML

```
DataVolumeKMSKeyId : string
```

Properties

DataVolumeKMSKeyId

Required: Yes

Type: string

Update requires: No interruption

EncryptionInTransit

Syntax

JSON

```
{
  "ClientBroker" : string,
  "InCluster" : boolean
}
```

YAML

```
ClientBroker : string
InCluster : boolean
```

Properties

ClientBroker

Required: No

Type: string

Update requires: No interruption

InCluster

Required: No

Type: boolean

Update requires: No interruption

OpenMonitoringInfo

Syntax

JSON

```
{
  "Prometheus" : PrometheusInfo
}
```

YAML

```
Prometheus :
  PrometheusInfo
```

Properties

Prometheus

Required: Yes

Type: *PrometheusInfo*

Update requires: No interruption

PrometheusInfo

Syntax

JSON

```
{
  "JmxExporter" : JmxExporterInfo,
  "NodeExporter" : NodeExporterInfo
}
```

YAML

```
JmxExporter :
  JmxExporterInfo
NodeExporter :
  NodeExporterInfo
```

Properties

JmxExporter

Required: No

Type: *JmxExporterInfo*

Update requires: No interruption

NodeExporter

Required: No

Type: *NodeExporterInfo*

Update requires: No interruption

JmxExporterInfo

Syntax

JSON

```
{
  "EnabledInBroker" : boolean
}
```

YAML

```
EnabledInBroker : boolean
```

Properties

EnabledInBroker

Required: Yes

Type: boolean

Update requires: No interruption

NodeExporterInfo

Syntax

JSON

```
{
  "EnabledInBroker" : boolean
}
```

YAML

```
EnabledInBroker : boolean
```

Properties

EnabledInBroker

Required: Yes

Type: boolean

Update requires: No interruption

LoggingInfo

Syntax

JSON

```
{
  "BrokerLogs" : BrokerLogs
}
```

YAML

```
BrokerLogs :
  BrokerLogs
```

Properties

BrokerLogs

|desc52|

Required: Yes

Type: *BrokerLogs*

Update requires: No interruption

BrokerLogs

Syntax

JSON

```
{
  "CloudWatchLogs" : CloudWatchLogs,
  "Firehose" : Firehose,
  "S3" : S3
}
```

YAML

```
CloudWatchLogs :
  CloudWatchLogs
Firehose :
  Firehose
S3 :
  S3
```

Properties

CloudWatchLogs

|desc53|

Required: No

Type: *CloudWatchLogs*

Update requires: No interruption

Firehose

|desc56|

Required: No

Type: *Firehose*

Update requires: No interruption

S3

|desc59|

Required: No

Type: *S3*

Update requires: No interruption

CloudWatchLogs

Syntax

JSON

```
{
  "Enabled" : boolean,
  "LogGroup" : string
}
```

YAML

```
Enabled : boolean
LogGroup : string
```

Properties

Enabled

|desc54|

Required: Yes

Type: boolean

Update requires: No interruption

LogGroup

|desc55|

Required: Yes

Type: string

Update requires: No interruption

Firehose

Syntax

JSON

```
{
  "DeliveryStream" : string,
  "Enabled" : boolean
}
```

YAML

```
DeliveryStream : string
Enabled : boolean
```

Properties

DeliveryStream

|desc57|

Required: Yes

Type: string

Update requires: No interruption

Enabled

|desc58|

Required: Yes

Type: boolean

Update requires: No interruption

S3

Syntax

JSON

```
{
  "Bucket" : string,
  "Enabled" : boolean,
  "Prefix" : string
}
```

YAML

```
Bucket : string
Enabled : boolean
Prefix : string
```

Properties

Bucket

|desc60|

Required: Yes

Type: string

Update requires: No interruption

Enabled

|desc61|

Required: Yes

Type: boolean

Update requires: No interruption

Prefix

|desc62|

Required: Yes

Type: string

Update requires: No interruption

2.4 Custom::Route53Certificate

The `Custom::Route53Certificate` resource requests an AWS Certificate Manager (ACM) certificate that you can use to enable secure connections. For example, you can deploy an ACM certificate to an Elastic Load Balancer to enable HTTPS support. For more information, see RequestCertificate in the AWS Certificate Manager API Reference.

Unlike `AWS::CertificateManager::Certificate`, this resource automatically validates the certificate for you. This only works if you request a certificate for a domain that's hosted on Route53.

2.4.1 Syntax

JSON

```
{
  "Type" : "Custom::Route53Certificate",
  "Properties" : {
    "ServiceToken" : {"Fn::ImportValue": "cfm-reslib"},
    "DomainName" : string,
    "SubjectAlternativeNames" : [ string, ... ]
  }
}
```

YAML

```
Type: Custom::Route53Certificate
Properties :
  ServiceToken : !ImportValue cfm-reslib
  DomainName : string
  SubjectAlternativeNames :
    - string
```

2.4.2 Properties

DomainName

Required: Yes

Type: string

Update requires: Replacement

SubjectAlternativeNames

Required: Yes

Type: List of string

Update requires: Replacement

CHAPTER

THREE

DEVELOPMENT

3.1 Preparing Environment

1. Get the source code

```
git clone https://github.com/CloudSnorkel/cfm-reslib.git``
```

2. Switch to the code directory

```
cd cfm-reslib
```

3. Install requirements

```
pip install -r requirements.txt
```

4. Create a virtual environment with all of the requirements

```
poetry install
```

3.2 Run Tests

Unit tests can be executed using py.test or simply with:

```
poetry run py.test tests
```

3.3 Building

The building process creates a CloudFormation template that can be deployed and expose `cfm-reslib` to be imported by other CloudFormation stacks. This template uses Lambda and its source code needs to be uploaded to a bucket. The build script will create both a ZIP file and a template and will upload it to a given S3 bucket.

```
BUCKET=my-bucket-name
poetry run python build.py $BUCKET
```

And just like when deploying the released versions of cfm-reslib, you can deploy this with `aws` CLI tool.

```
BUCKET=my-bucket-name
aws cloudformation create-stack --stack-name cfm-reslib --template-url https://s3.
↪amazonaws.com/$BUCKET/cfm-reslib-latest.template --capabilities CAPABILITY_IAM
```

Or when updating:

```
BUCKET=my-bucket-name
aws cloudformation update-stack --stack-name cfm-reslib --template-url https://s3.
↪amazonaws.com/$BUCKET/cfm-reslib-latest.template --capabilities CAPABILITY_IAM
```

Note that you won't be able to deploy multiple stacks of cfm-reslib in the same region because the exported name has to be unique across all stacks in a certain region.

3.4 Adding Custom Resources

There are two methods to implement a new custom resource. You will need to create a class for your resource in both.

1. If the custom resource uses just one boto3 call to create, update and delete a resource, you can inherit from *cfmreslib.boto.BotoResourceHandler*. Simply override all of the constants with the names of the methods that need to be called and you're done. Check out `ElasticTranscoderPipeline` for an example.
2. If you need more control of the process, inherit from *cfmreslib.base.CustomResourceHandler*. You will have to implement some methods that will be called for requests coming from CloudFormation. Check out `Route53Certificate` for an example.

Once you've added your custom resource, make sure to add it to `ALL_RESOURCES` at the end of `resources.py`.

3.4.1 Classes

class cfmreslib.base.**CustomResourceHandler**

Abstract base class for all custom resources. Implement this class for new resources. Check the documentation for each method. Not all methods are always required.

NAME = '<not set>'

Custom resource name to be used in CloudFormation with `Custom::` prefix.

DESCRIPTION = '<not set>'

Resource description for automatically generated documentation.

EXAMPLES: List[Dict[str, str]] = []

Optional resource examples to be used in documentation. Each example needs "title", "description" and "template".

REPLACEMENT_REQUIRED_ATTRIBUTES = {}

set of properties that require a replacement on update value changes.

exists() → bool

Checks if the resource specified in self.physical_id exists.

- Must always be implemented

Returns *True* if the resource exists, *False* if not

ready() → bool

Checks if the resource specified in self.physical_id is ready.

- Must always be implemented
- Can just return *True* if a resource existing means it's ready

Returns *True* if the resource exists, *False* if not

data() → Optional[Dict[str, object]]
Retrieves the current data that should be returned for this resource.

- Only required if `_wait_ready()` is used

Returns resource data, can be *None*

create(*args: Dict[str, object]*) → None
Creates a new resource with supplied arguments.

- Must set *self.physical_id*
- Must call `_success()`, `_fail()` or `_wait_ready()`
- Must always be implemented

Parameters **args** – arguments as passed from CloudFormation

can_update(*old_args: Dict[str, object]*, *new_args: Dict[str, object]*, *diff: List[str]*) → bool
Checks if a resource can safely be updated or whether a new one has to be created.

- Must always be implemented, but can just return *False* if needed.

Parameters

- **old_args** – existing arguments as passed from CloudFormation for the current resource
- **new_args** – requested arguments as passed from CloudFormation
- **diff** – a list of argument names that have changed value

Returns *True* if the resource can be updated or *False* if it needs to be recreated

update(*old_args: Dict[str, object]*, *new_args: Dict[str, object]*, *diff: List[str]*) → None
Updates the resource specified in self.physical_id based on the old and new arguments.

- Must call `_success()`, `_fail()` or `_wait_ready()`
- Only required if *can_update()* ever returns *True*.

Parameters

- **old_args** – existing arguments as passed from CloudFormation for the current resource
- **new_args** – requested arguments as passed from CloudFormation
- **diff** – a list of argument names that have changed value

delete() → None
Deletes the resource specified in self.physical_id .

- Must call `_success()`, `_fail()` or `_wait_delete()`
- Must always be implemented

get_iam_actions() → List[str]
Returns a list of required IAM permissions for all operations.

- Must always be implemented

class cfmreslib.boto.**BotoResourceHandler**

NAME = None
Custom resource name to be used in CloudFormation with `Custom::` prefix.

SERVICE = None
boto3 service name that will be used to create the client (e.g. s3, acm, ec2).

CREATE_METHOD = {}
Descriptor for method used to create resource. Requires "name" with the name of the method, and "physical_id_query" used to query for the physical id of the newly created resource from the method return value.

UPDATE_METHODS = []
Optional list of descriptor for methods used to update an existing resource. Each item requires "name" with the name of the method, and "physical_id_argument" with the name of the method argument that needs to have the physical id of the updated resource.

EXISTS_METHOD = {}
Descriptor for method used to check if resource exists. Requires "name" with the name of the method, and "physical_id_argument" with the name of the method argument that needs to have the physical id of the checked resource. This method will raise the exception set in `NOT_FOUND_EXCEPTION` when the resource does not exist.

EXIST_READY_QUERY = {}
Optional descriptor of query to check against the result of `EXISTS_METHOD`. When set we will wait until the resource is ready before finishing with create and update operations. Requires "query" with the query to run over the exists method result, "expected_value" with the expected value (e.g. READY), and "failed_values" with values that denote failure and should stop the operation.

DELETE_METHOD = {}
Descriptor for method used to delete an existing resource. Requires "name" with the name of the method, and "physical_id_argument" with the name of the method argument that needs to have the physical id of the resource.

NOT_FOUND_EXCEPTION = ''
Name of exception thrown by the exists method if the resource doesn't exist.

EXTRA_PERMISSIONS = []
A list of extra permissions required by any operations for this resource. Most permissions will be deduced by method names, but sometimes extra IAM permissions are required.

INDEX