AWS CloudFormation Tutorial

Flux7

About Flux7

Flux7: Cloud and DevOps Solutions

Founded in 2013
Team of 35+
Headquartered in Austin, Texas

Achievements

AWS DevOps, Migration, Healthcare, and Life Sciences Competencies

TechTarget's "**Impact Best AWS Consulting Partner**" two years in a row (2015 & 2016)

Partner Recognition Award by AWS at reInvent 2015

Customers featured on stage at AWS re:Invent three years in a row

Docker Foundation and authorized consulting partner

150+ happy customers through word of mouth

> *"[Flux7] taught us how to do 10x the work in 1/10th the time" - Patrick K, AWS Re:invent'14, CTO's Keynote*

Flux7

Pre-reqs

A laptop with web browser, a text editor, and Wifi

AWS account with PowerUser privileges

Outcomes

You will be able to:

- ☑ Understand the anatomy of CloudFormation
- ☑ Be able to read and modify CloudFormation
- ☑ Deploy a CloudFormation stack
- ☑ Update a CloudFormation stack
- ☑ Access a web app running on the servers behind an ELB

Plan

Present

- ➔ Provide a big picture view of CloudFormation
- ➔ CloudFormation concepts
- ➔ Walkthrough a sample CloudFormation stack

Hands On

- ➔ Deploy a simple CloudFormaitonStack
- ➔ Deploy a complex Stack
- ➔ Extend a CloudFormation stack

What is CloudFormation?

An “Infrastructure as code” description of AWS resources desired by the user

What is CloudFormation?

✓ Resources are described in JSON

```
"WebSg": {
  "Type" : "AWS::EC2::SecurityGroup",
  "Properties" : {
    "GroupDescription" : " Security group attached to the webservers ",
    "SecurityGroupIngress" : [
       {
          "CidrIp":"10.0.0.0/8",
          "FromPort":"80",
          "ToPort":"80",
          "IpProtocol":"tcp"
       }
    ],
    "VpcId" : "vpc-2664fb41"
  }
},
```

What can CF provision?

- **Network:** VPC, subnets, routing tables, gateways
- **Infrastructure:** Instances, Load balancers
- **IAM:** User accounts, permissions, groups, and privileges
- **Custom:** It is possible to declare customer resources which can be inside or outside AWS
- **... and even Software:** to run on EC2 instances

CloudFormation Features

Handles ordered creation and deletion of resources

Allows for a smart update of stacks:
Updates will create new resource before destruction of the old one

Includes automated rollback on failure

Leaves an audit trail of changes applied

Taxonomy: Nouns

Template:

A JSON file containing a description of the architecture

Stacks:

Instantiation of a CloudFormation template

Parameters:

The input parameters provided when creating a stack

Resources:

The resources that make up a stack

Events:

The events that take place while a CF operation (e.g., creation, update, deletion, or rollback) is taking place

Flux7

Taxonomy: Verbs

Create:
The operation of creating a new CloudFormation stack using a template

Update:
The action to update an existing stack by making changes to the template

Delete:

The action to delete a CloudFormation stack

Rollback:

The actions triggered when a Stack creation or update fails mid-way. Purpose is to undo any changes made by the operation stack

Flux7

Taxonomy: Adjectives

Created:

A stack creation operation has successfully completed

Updated:

A stack update operation has successfully completed

Deleted:

A stack update operation has successfully completed and the stack is no longer listed as a stack on the AWS console

Corrupted:

A stack is considered corrupted if the rollback operation fails

Flux7

Anatomy of a Template

Parameters

Declare parameters that the user must specify when they want to create a stack

Examples: Name of an instance, number of instances, instance types, etc

Resources

Declare resources to be created when a stack is created

Examples: Instances, ELBs, VPCs, EBS volume, S3 buckets, etc

Outputs

The outputs to be displayed to the user once the stack creation/update has completed

Examples: Endpoint of the ELB

Attributes

The properties of a parameter, resource, and outputs. Some are defined in the template and others get populated when the stack is created

Anything else?

- **References**: It is possible to reference one resource when defining the attributes of another resource, e.g., use a reference to the routing table when creating a subnet
- **Conditionals**: It is possible to set a property based on a conditional, e.g, if a DB snapshot parameter is defined, use it otherwise to create a fresh DB
- **Depends On and Signals**: It is possible to enforce order on creation of resources, e.g., that application servers be created *after* the DB has been created
- **Joins**: To help with creating names of resources, you can concatenate two or more strings
- **Mappings:** To match a key with a corresponding set of name values

Anything else?

- **API Version**: API Version of CF template you can specify in each template
- **Signals**: Mechanism to signal CloudFormation when a particular activity is complete

How do I:

? **Write a template**
- ◎ Text editor (VS, Eclipse)

? **Catch syntactic errors**
- ◎ CF validate

? **Catch logical errors**
- ◎ ChangeSets

? **Provision**
- ◎ Create stack

? **Access outputs**
- ◎ Review the console

? **Update an existing stack**
- ◎ Update stack

? **Debug errors**
- ◎ Review error logs

Resources

Starter Code:
https://github.com/Flux7Labs/aws-devops-tutorial

Thank You