AWS CloudFormation StackSets 활용

엄기성 / AWSKRUG 판교 소모임 / 2019.03.13

Speaker

- 이름: 엄기성(GiSeong Eom)
- 회사: 판교 K모 게임회사
- 업무: Cloud/Infra Operation (a.k.a. Ops)
- 취미: https://github.com/giseongeom
- 활동
  - AWSKRUG CLI 소모임
  - AWSKRUG 판교 소모임

Disclaimer

이 슬라이드의 내용은 전적으로 **발표자 개인 의견**입니다.

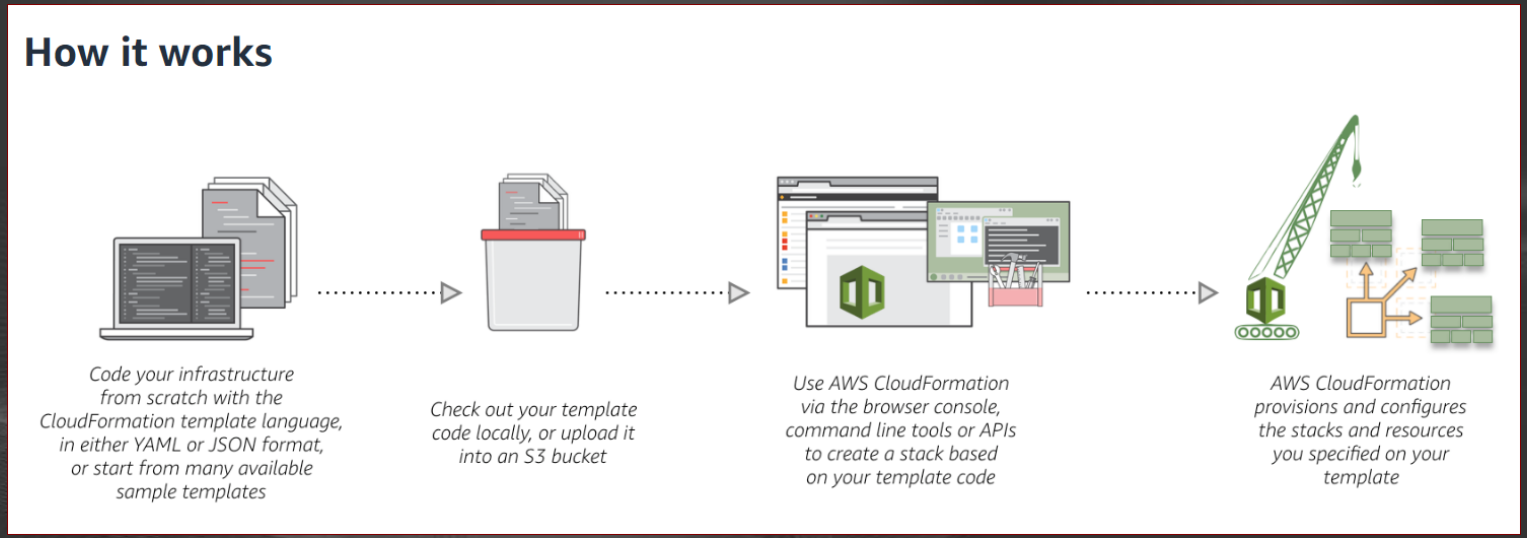
고용주/부서의 정책, 의견과 무관함을 미리 밝혀 둡니다.

Agenda

- AWS CloudFormation 소개
- AWS CloudFormation StackSet 소개
- AWS CloudFormation StackSet 정리

CloudFormation

- AWS에서 제공하는 리소스 배포 / 관리도구
- 일단 사용하면 Infrastructure As a Code 비젼을 체감



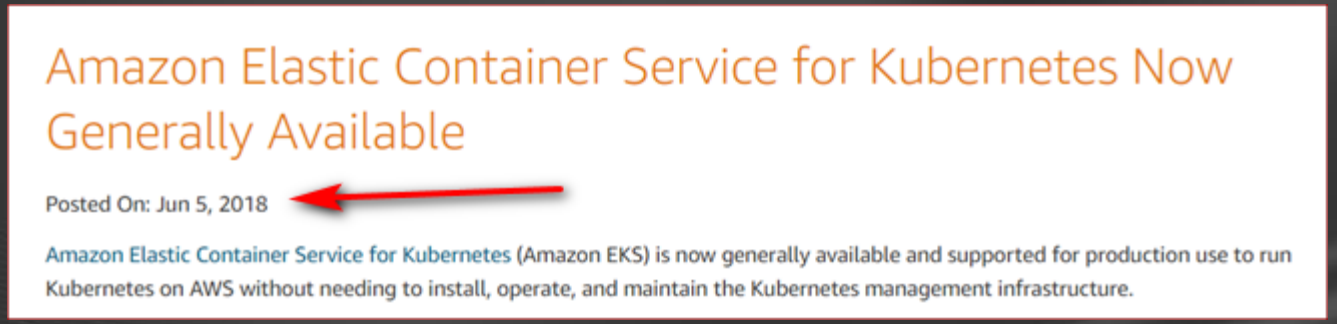
이미지 출처: https://aws.amazon.com/cloudformation/

CloudFormation - Features

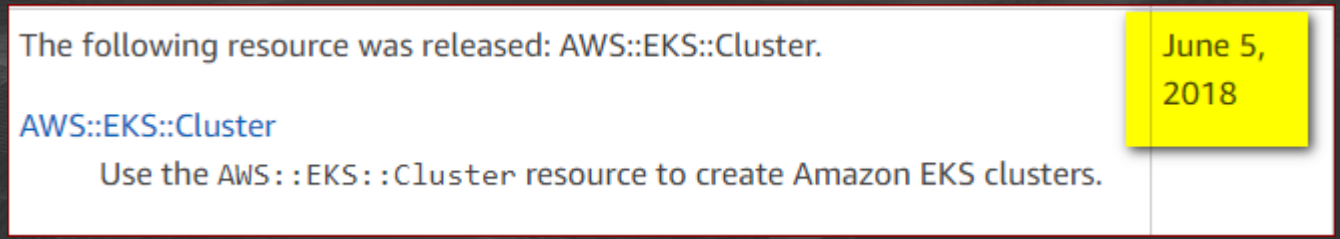
- Authoring with JSON/YAML
- Safety Controls
- Preview Changes To Your Environment (ChangeSet)
- Dependency Management
- Cross Account And Cross-Region Management (**StackSet**)
- Extensibility

CloudFormation - Support

- AWS의 새로운 제품이 발표되면 동시에 CloudFormation 지원도 추가된다.



이미지 출처: https://aws.amazon.com/about-aws/whats-new/2018/06/amazon-elastic-container-service-for-kubernetes-eks-now-ga/



이미지 출처: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/ReleaseHistory.html

CloudFormation - Competitors

- 유사한 제품
  - Azure Resource Manager Template (JSON)
  - Hashicorp Terraform (HCL)
  - Google Cloud Deployment Manager (YAML)

AWS CloudFormation

```yaml
AWSTemplateFormatVersion: "2010-09-09"

Description: S3 Storages

Parameters:
  EnvironmentId:
    Type: String

  ProductId:
    Type: String

Resources:
  MyS3bucket:
    Type: AWS::S3::Bucket
    Properties:
      BucketName: !Sub "${AWS::StackName}-${AWS::AccountId}-${AWS::Region}"
      Tags:
        - Key: Env
          Value: !Ref EnvironmentId
        - Key: Product
          Value: !Ref ProductId
```

Azure Resource Manager Template

```
{
  "$schema": "http://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "storageNamePrefix": {
      "type": "string",
      "maxLength": 11
    }
  },
  "variables": {
    "storageName": "[concat(toLower(parameters('storageNamePrefix')), uniqueString(resourceGroup().id))]"
  },
  "resources": [
    {
      "name": "[variables('storageName')]",
      "type": "Microsoft.Storage/storageAccounts",
      "apiVersion": "2016-01-01",
      "sku": {
        "name": "Standard_LRS"
      },
      "kind": "Storage",
      "location": "[resourceGroup().location]",
      "tags": {},
      "properties": {}
    }
  ]
}
```

Hashicorp Terraform (HCL)

```
terraform {
  required_version = ">= 0.11.0"
}

provider "aws" {
  version = "~> 0.1"
}

resource "aws_s3_bucket" "example" {
  bucket = "${var.build_backup_bucket_name}"
  acl    = "private"
}
```

{ CloudFormation }

초간단 CFN Stack 생성 / 변경 / 삭제
demo1-simple-cfn-stack.ps1 참고

simple-cfn-stackV1.yml

```yaml
Resources:
  MyApplication:
    Type: "AWS::CodeDeploy::Application"
    Properties:
      ApplicationName: !Sub "${AWS::StackName}-${AWS::AccountId}-${AWS::Region}"
```

simple-cfn-stackV2.yml

```yaml
Resources:
  MyApplication:
    Type: "AWS::CodeDeploy::Application"
    Properties:
      ApplicationName: !Sub "${AWS::StackName}-${AWS::AccountId}-${AWS::Region}"

  MyCodeDeploybucket:
    Type: AWS::S3::Bucket
    Properties:
      BucketName: !Sub "${AWS::StackName}-${AWS::AccountId}-${AWS::Region}"
```

Agenda

- AWS CloudFormation 소개
- AWS CloudFormation StackSet 소개
- AWS CloudFormation StackSet 정리

CloudFormation StackSet

- 2017.07.25 CloudFormation StackSets 발표 (발표자 생일 전날)

Use CloudFormation StackSets to Provision Resources Across Multiple AWS Accounts and Regions
https://aws.amazon.com/blogs/aws/use-cloudformation-stacksets-to-provision-resources-across-multiple-aws-accounts-and-regions/

CloudFormation StackSet – History

2017

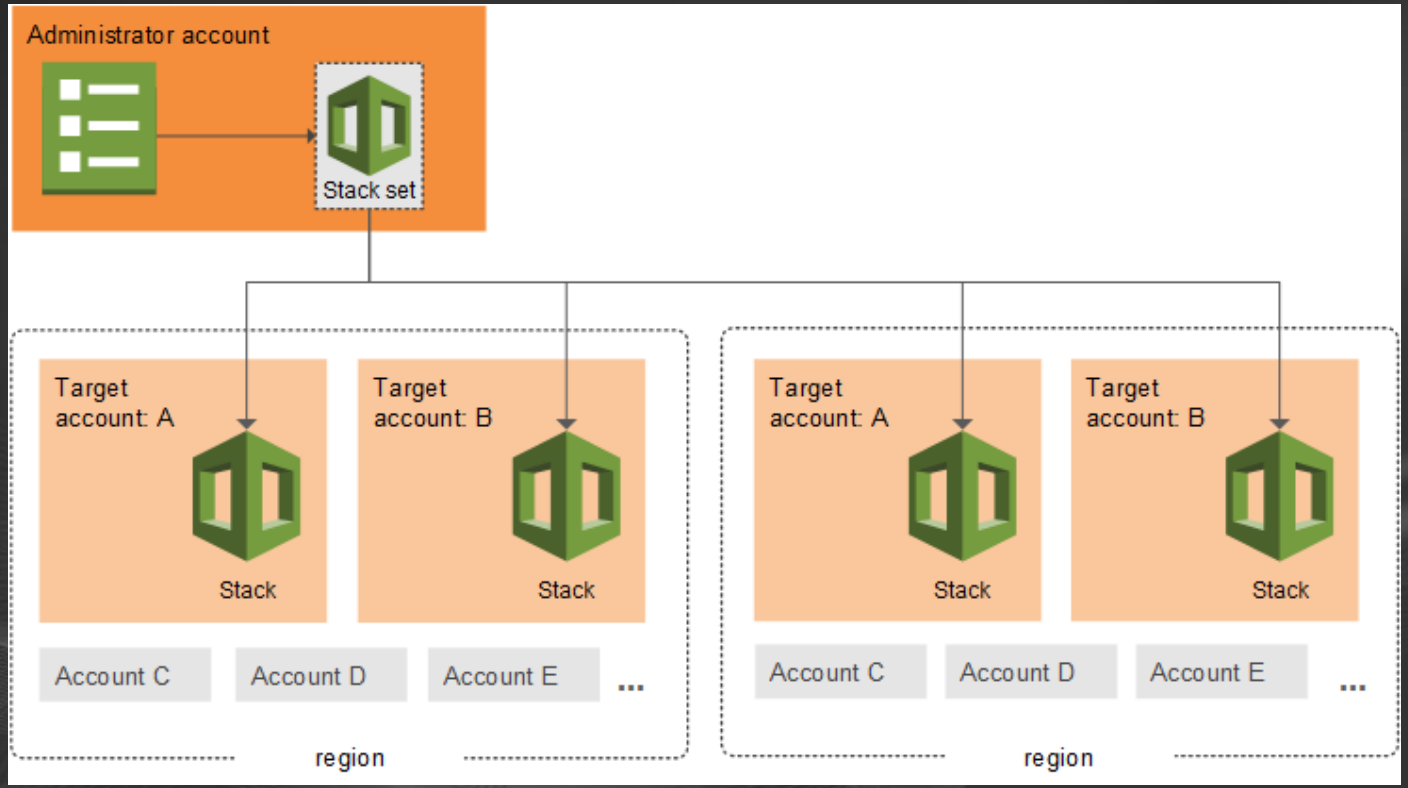
- 2017.07.25 CloudFormation StackSet 발표
- 2017.11.06 Maximum 500 stack instances per stack set
- 2017.11.17 Stack instance overrides added for stack sets

2018

- 2018.05.30 Selective updates of stack instances
- 2018.12.13 Stack instance operation limit (1500)

CloudFormation - Release History
https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/ReleaseHistory.html

CloudFormation StackSet - Architecture



이미지 출처: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/what-is-cfnstacksets.html

CloudFormation StackSet – Requirements

- Administrator Account
- Target Account(s)
- IAM Service Role
  - AWSCloudFormationStackSetAdministrationRole
  - AWSCloudFormationStackSetExecutionRole
  - Stack Set이 사용하는 Role의 이름은 정해져 있으며 변경 불가능

CloudFormation StackSet – Requirements

Administrator Account

- 이름처럼 StackSet 관리자(?) 역할
- StackSet을 생성하는 AWS Account
- AWSCloudFormationStackSetAdministrationRole 생성

Target Account(s)

- StackSet의 결과물(Stack)이 생성되는 AWS Account(s)
- AWSCloudFormationStackSetExecutionRole 생성

CloudFormation StackSet – Options

Maximum concurrent accounts

- 병렬 처리 단위 / 작업 시간에 영향
- 단위: count / percent (%)

Failure tolerance

- StackSet Operation 실패 기준
- 단위: count / percent (%)

Retain Stack

- Stack instance를 삭제할 때 Stack을 남겨두는 옵션

{ CloudFormation StackSet }

초간단 CFN StackSet 생성 / 변경 / 삭제
demo2-prepare-cfn-stackset.ps1
demo3-manage-cfn-stackset.ps1
demo4-manage-cfn-stackset-SingleAccount.ps1

AWSCloudFormationStackSetAdministrationRole.yml

```yaml
AWSTemplateFormatVersion: 2010-09-09
Description: AWSCloudFormationStackSetAdministrationRole

Resources:
  StackSetAdminRole:
    Type: AWS::IAM::Role
    Properties:
      RoleName: AWSCloudFormationStackSetAdministrationRole
      AssumeRolePolicyDocument:
        Version: 2012-10-17
        Statement:
          - Effect: Allow
            Principal:
              Service: cloudformation.amazonaws.com
            Action:
              - sts:AssumeRole
      Path: /
      Policies:
        - PolicyName: AssumeRole-AWSCloudFormationStackSetExecutionRole
          PolicyDocument:
            Version: 2012-10-17
            Statement:
              - Effect: Allow
                Action:
                  - sts:AssumeRole
                Resource:
                  - "arn:aws:iam::*:role/AWSCloudFormationStackSetExecutionRole"
```

AWSCloudFormationStackSetExecutionRole.yml

```yaml
AWSTemplateFormatVersion: 2010-09-09
Description: AWSCloudFormationStackSetExecutionRole

Parameters:
  AdministratorAccountId:
    Type: String
    Description: AWS Account Id of the administrator account (the account in which StackSets will be created).
    MaxLength: 12
    MinLength: 12

Resources:
  ExecutionRole:
    Type: AWS::IAM::Role
    Properties:
      RoleName: AWSCloudFormationStackSetExecutionRole
      AssumeRolePolicyDocument:
        Version: 2012-10-17
        Statement:
          - Effect: Allow
            Principal:
              AWS:
                - !Ref AdministratorAccountId
            Action:
              - sts:AssumeRole
      Path: /
      ManagedPolicyArns:
        - arn:aws:iam::aws:policy/AdministratorAccess
```

Agenda

- AWS CloudFormation 소개
- AWS CloudFormation StackSet 소개
- AWS CloudFormation StackSet 정리

CloudFormation StackSet – 구조적인 한계

- CFN의 집합(?)이므로 그 한계점을 그대로 상속함
  - 부족한 Intrinsic Function & Pseudo Parameters
  - Terraform HCL에 비교하면 장난 아님
  - 리소스 이름에 Random 문자열 하나 넣으려면 삽질이 많음
- 구글링하면서 발견한 "My CloudFormation wish list"

CloudFormation StackSet – 사용하면서

- StackSet 소스는 Local file 사용불가. 매번 S3에 업로드 ㅠㅠ
  (Public Access 허용된 S3를 사용할 필요는 없어서 다행)
  Azure RM template은 Public-Access 허용된 곳에 위치해야 한다. workaround는 있음

- Stack Instance의 Name/Pattern을 지정할 수 없다.
  StackName 기반의 문자열 처리 불가능함
  (예) StackSet-singleAWSAccount-Stackset-Demo-f5009c99-5cc3-4541-aa2f-2ac726fc6645

- 여러 Region에서 작업하는 경우, 기대보다 빠르지 않음

- Terminate-Protection 없음 ㅠㅠ
  개별 stack 에서는 terminate-protection이 설정할 수 있지만, stack-set 수준에서는 불가능

CloudFormation StackSet - 결론

- AWS Account / Region별 리소스 이름 중복이 없도록 디자인
- 해당 리소스가 지속적으로 배포/관리 필요한지 확인
- Monolithic 스타일의 큰 template 대신 작은 코드부터 시작하는 것이 필요
- AWS Region별 차이점도 미리 고려하자
  (모든 AWS Region != us-east-1)

References

- AWS CodeDeploy pricing https://aws.amazon.com/codedeploy/pricing/
- Working with AWS CloudFormation StackSets https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/what-is-cfnstacksets.html
- CloudFormation StackSets: automated cross-account/region deployments https://sanderknape.com/2017/07/cloudformation-stacksets-automated-cross-account-region-deployments/
- How do I reference a resource in another AWS CloudFormation stack during template creation? https://aws.amazon.com/premiumsupport/knowledge-center/cloudformation-reference-resource/

References (cont'd)

- AWS Management Tools Blog / AWS CloudFormation https://aws.amazon.com/blogs/mt/category/management-tools/aws-cloudformation/
- AWS CloudFormation: 2018 in review https://aws.amazon.com/blogs/mt/aws-cloudformation-2018-in-review/
- Deploy private Resource Manager template with SAS token and Azure PowerShell https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-manager-powershell-sas-token
- My CloudFormation wish list https://www.kencochrane.net/2017/03/25/my-cloudformation-wishlist/
- Azure Resource Manager template functions https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-template-functions

References (cont'd)

- terraform 0.11 and Older / Interpolation Syntax
  https://www.terraform.io/docs/configuration-0-11/interpolation.html
- Up your AWS CloudFormation game with Visual Studio Code
  https://hodgkins.io/up-your-cloudformation-game-with-vscode

Thank You

발표자료 / 예제소스 다운로드