

AWS Serverless Application Model(AWS SAM)

- AWS Serverless Application Model(AWS SAM)은 코드형 인프라(IaC)를 사용하여 서버리스 애플리케이션을 빌드하기 위한 오픈 소스 프레임워크입니다.
- 개발자는 AWS SAM의 간편 구문을 사용하여 배포 중에 인프라로 변환되는 AWS CloudFormation 리소스와 특수한 서버리스 리소스를 선언합니다.

주요 기능

- 더 적은 코드를 사용하여 애플리케이션 인프라 코드 빠르게 정의하기
 - AWS SAM 템플릿을 작성하여 서버리스 애플리케이션 인프라 코드를 정의합니다.
 - AWS CloudFormation에 템플릿을 직접 배포하여 리소스를 프로비저닝합니다.
- 전체 개발 수명 주기에 걸쳐 서버리스 애플리케이션을 관리합니다.
 - 개발 수명 주기의 작성, 구축, 배포, 테스트 및 모니터링 단계에 걸쳐 서버리스 애플리케이션을 관리하려면 AWS SAM CLI를 사용합니다.

- AWS SAM 커넥터를 사용하여 리소스 간에 권한을 신속하게 프로비저닝
 - AWS SAM 템플릿의 AWS SAM 커넥터를 사용하여 AWS 리소스 간 권한을 정의합니다.
 - AWS SAM은 코드를 의도를 달성하는 데 필요한 IAM 권한으로 변환합니다.
- 개발 과정에서 로컬 변경 사항을 클라우드에 지속적으로 동기화
 - AWS SAM CLI `sam sync` 명령을 사용하면 로컬 변경 사항을 클라우드에 자동으로 동기화하여 개발 및 클라우드 테스트 워크플로의 속도를 높일 수 있습니다.

주요 명령어

애플리케이션 빌드 명령어

```
$ sam build # 빌드
...
Running PythonPipBuilder:CleanUp
Running PythonPipBuilder:ResolveDependencies
Running PythonPipBuilder:CopySource
Running PythonPipBuilder:CopySource

Build Succeeded
...
```

buildspec.yml

```
...  
  build:  
    commands:  
      - echo building Source by SAM...  
      - sam build --template sns/slack/template-lambda.yml  
...
```

EXPLORER

4. aws cloudformation.md U buildspec-lambda.yml X

sns > slack > buildspec-lambda.yml > {} phases > {} build > [] commands

12

1. slack.md

2. slackbot.md

3. slackbot with python.md

6. slackbot with eventbridg... U

7. slackbot with sns.md U

alarm.py

buildspec-eventbridge.yml

buildspec-lambda.yml

buildspec-sns.yml

event.py

README.md

template-eventbridge.yml 1

```
7
8 phases:
9   install:
10     runtime-versions:
11       python: 3.11
12
13   pre_build:
14     commands:
15       - echo Build started on `date`
16
17   build:
18     commands:
19       - echo building Source by SAM...
20       - sam build --template sns/slack/template-lambda.yml
21
22   post_build:
```

AWS CloudFormation

- AWS CloudFormation 은 AWS 리소스를 모델링하고 설정하여 리소스 관리 시간을 줄이고 AWS에서 실행되는 애플리케이션에 더 많은 시간을 사용하도록 해 주는 서비스입니다.

구성요소



작동 방식

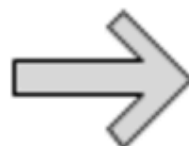
1. 구성하고자 하는 인프라에 대한 리소스를 JSON 또는 YAML 형식의 템플릿으로 작성한다.
2. 작성한 템플릿을 S3 혹은 로컬에 저장한다.
3. 템플릿 파일의 위치를 지정하여 스택을 생성하고 리소스를 구성한다.



1 Create or use an existing template



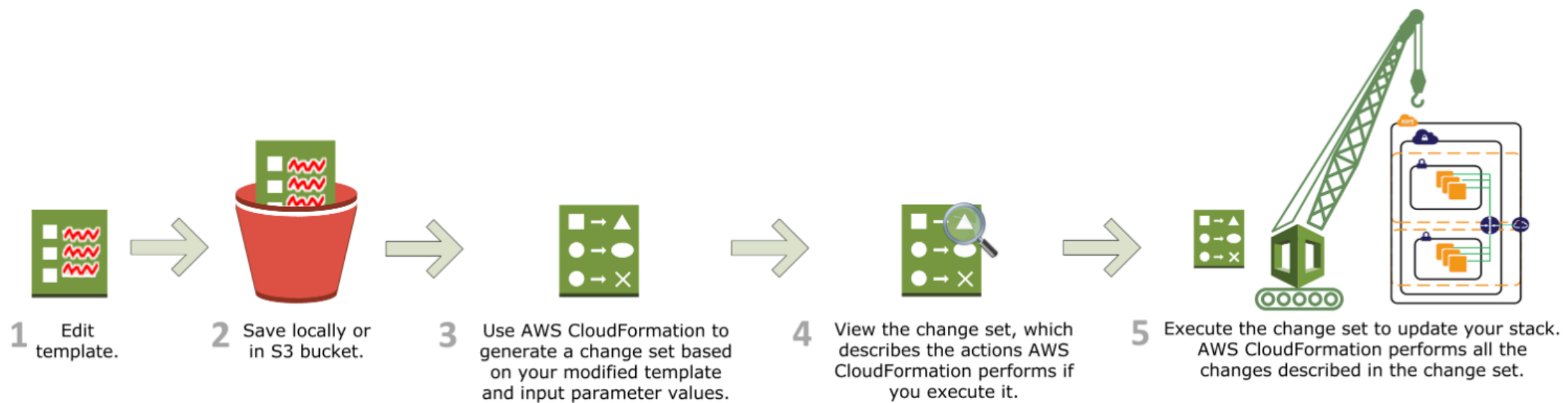
2 Save locally or in S3 bucket



3 Use AWS CloudFormation to create a stack based on your template. It constructs and configures your stack resources.

변경 세트로 스택 업데이트

1. 템플릿을 수정한다.
2. 수정된 템플릿을 S3 혹은 로컬에 저장한다.
3. CloudFormation은 수정된 템플릿 혹은 파라미터를 기반으로 변경 세트(Change Set)을 생성한다.
4. 변경 세트를 통해 변경 사항을 확인 후 변경 세트를 실행시킨다.
5. 실행된 변경 세트는 스택을 업데이트한다.



template.yml

```
AWSTemplateFormatVersion: '2010-09-09'  
Transform: AWS::Serverless-2016-10-31  
Description: |  
    설명글  
  
Parameters: # Resources에서 사용할 변수선언  
    ...  
  
Resources: # aws에 구축할 리소스 정의  
    ...
```

EXPLORER

▼ COURSE_AWS

- ▼ sns
 - ▼ slack
 - 1. slack.md
 - 2. slackbot.md
 - 3. slackbot with python.md
 - 6. slackbot with eventbrid... U
 - 7. slackbot with sns.md U
 - alarm.py
 - buildspec-eventbridge.yml
 - buildspec-lambda.yml
 - buildspec-sns.yml
 - event.py
 - README.md
 - template-eventbridge.yml 1
 - template-lambda.yml 1
 - template-sns.yml 4

sns > slack > template-lambda.yml > Serverless IDE > {} Resources > {} AlarmL

Visualize with Infrastructure Composer

```
1 AWSTemplateFormatVersion: '2010-09-09'
2 Transform: AWS::Serverless-2016-10-31
3 Description: |
4   SNS Slack Lambda Sample Template https://docs.aws.amazon.com/ko\_kr/AWSCloudF
6   UserGuide/AWS\_Events.html
7 #####
8 # Parameters
9 # https://docs.aws.amazon.com/ko\_kr/AWSCloudFormation/latest/UserGuide/
11 parameters-section-structure.html
12 #####
13 Parameters:
14   DefaultName:
15     Type: String
16     Default: sns-slack
17   ServiceType:
18     Type: String
19     Default: tutorial
20   PythonVersion:
```

AWS Cloudformation Role 생성

단계1: IAM > Create role

The screenshot shows the AWS IAM console interface. The left sidebar is titled 'Identity and Access Management (IAM)' and contains a search bar and a list of navigation items: 'Dashboard', 'Access management' (expanded), 'User groups', 'Users', 'Roles' (highlighted with a red arrow), 'Policies', and 'Identity providers'. The main content area is titled 'Roles (6)' and includes a description: 'An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.' Below this is a search bar and a table of roles. The table has columns for 'Role name', 'Trusted entities', and 'Last activity'. A red arrow points to the 'Create role' button in the top right corner of the main content area.

Role name	Trusted entities	Last activity
AWSCodePipelineServiceRole-ap-northeast-2-investment-crypto-pip	AWS Service: codepipeline	24 hours ago
AWSServiceRoleForCostOptimizationHub	AWS Service: cost-optimization-hub	24 hours ago
AWSServiceRoleForSupport	AWS Service: support (Service-Linker)	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service-Linker)	-
codebuild-investment-crypto-service-role	AWS Service: codebuild	23 hours ago
investment-crypto-stack-role	AWS Service: cloudformation	23 hours ago

Trusted entity type

☒ **AWS service**

Allow AWS services like EC2, Lambda, or others to perform actions in this account.

☐ **AWS account**

Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

☐ **Web identity**

Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

☐ **SAML 2.0 federation**

Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

☐ **Custom trust policy**

Create a custom trust policy to enable others to perform actions in this account.

Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case

CloudFormation

Choose a use case for the specified service.

Use case

☒ **CloudFormation**

Allows CloudFormation to create and manage AWS stacks and resources on your behalf.

Cancel

Next

단계2: Add Permissions

```
AWSCloudFormationFullAccess  
IAMFullAccess  
AmazonEventBridgeFullAccess  
AmazonS3FullAccess  
AWSLambda_FullAccess  
CloudWatchEventsFullAccess  
CloudWatchFullAccess
```

단계3: 결과 확인

The screenshot shows the AWS IAM console interface. The top navigation bar includes the AWS logo, a search bar, and various service icons. The left sidebar displays the 'Identity and Access Management (IAM)' section with a search bar and a list of navigation items: Dashboard, Access management (expanded), User groups, Users, Roles (highlighted with a red arrow), Policies, Identity providers, Account settings, and Root access management. The main content area is titled 'Roles (7)' and includes a description: 'An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.' Below this is a search bar and a table listing the roles.

<input type="checkbox"/>	Role name	Trusted entities	Last activity
<input type="checkbox"/>	AWSCodePipelineServiceRole-ap-northeast-2-investment-crypto-pip	AWS Service: codepipeline	24 hours ago
<input type="checkbox"/>	AWSServiceRoleForCostOptimizationHub	AWS Service: cost-optimization-hub	24 hours ago
<input type="checkbox"/>	AWSServiceRoleForSupport	AWS Service: support (Service-Linker)	-
<input type="checkbox"/>	AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service-Linker)	-
<input type="checkbox"/>	codebuild-investment-crypto-service-role	AWS Service: codebuild	23 hours ago
<input type="checkbox"/>	investment-crypto-stack-role	AWS Service: cloudformation	24 hours ago
<input type="checkbox"/>	sns-slack-cloudformation-role	AWS Service: cloudformation	-