

透過 AWS CDK 建立自動化部署機制

Automate a CI/CD pipeline with SlackBot Interaction with AWS CDK

Github: @ookangzheng

Taiwan CDK Meetup #2

Who am I

- Wizardamigos Taipei Meetup Co-organizer
- Open Culture Foundation volunteer
- Work at **CoolBitX**



Edward



<https://www.coolwallet.io>

SYGNA[®]

<https://sygna.io>

WE BUILD SECURE AND EASY-TO-USE BLOCKCHAIN SOLUTIONS
TO HELP FOSTER THE MASS ADOPTION OF VIRTUAL ASSETS

Outline

- Where we begin
- Codepipeline/Codebuild
- Approval + Slackbot
- Recap
- Q&A

Where we begin

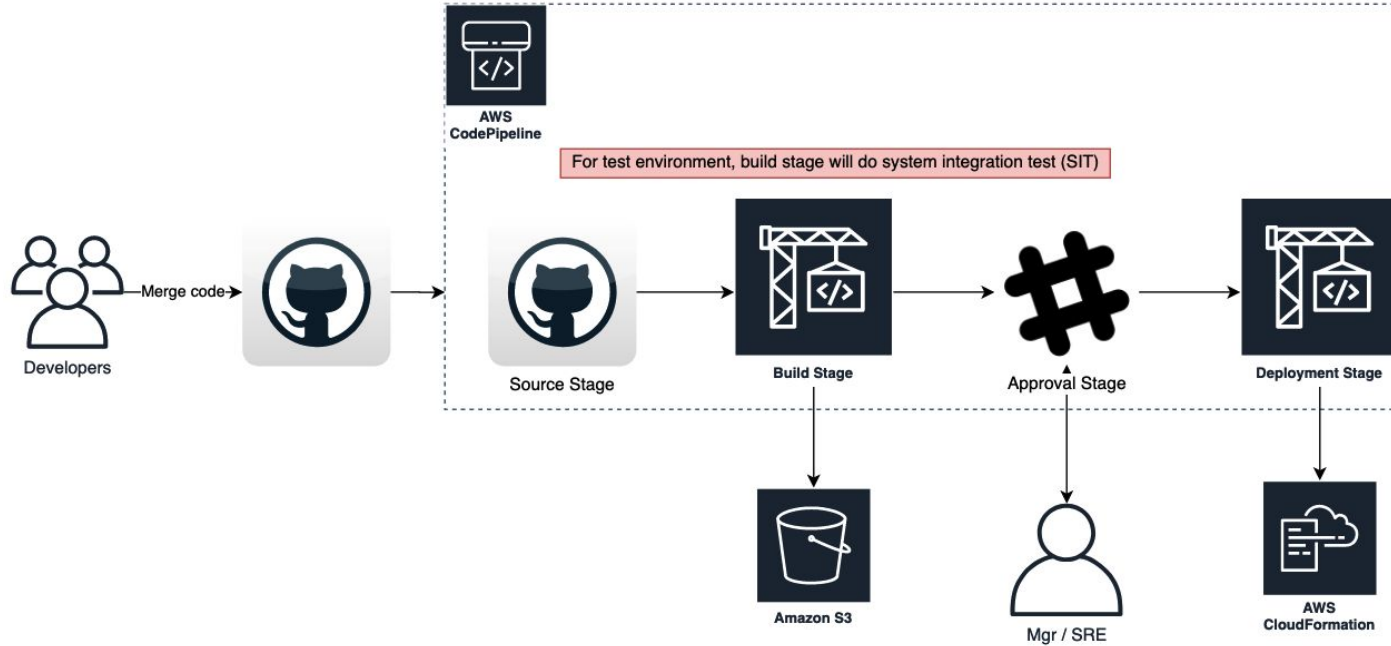
Why CDK

1. IAC
2. CDK enables developers to describe their infrastructure in code using an object model, then synthesize it into Cloudformation templates.
3. I feel a lot more productive with CDK, I am writing less code and producing more secure infrastructure.



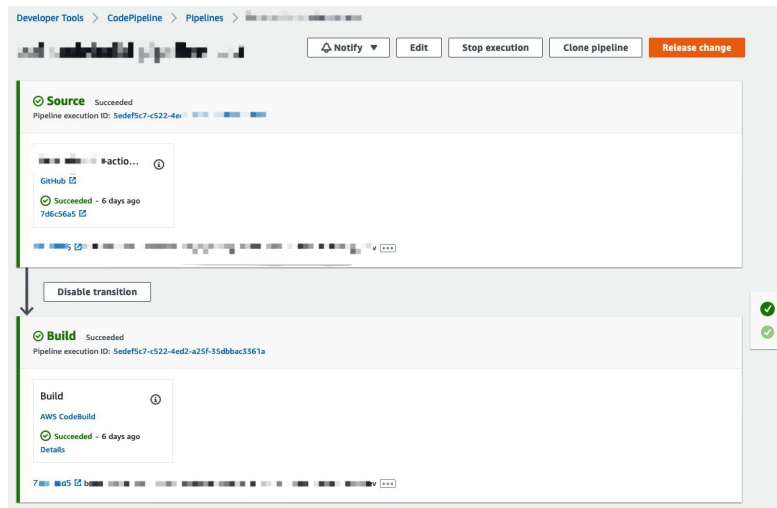
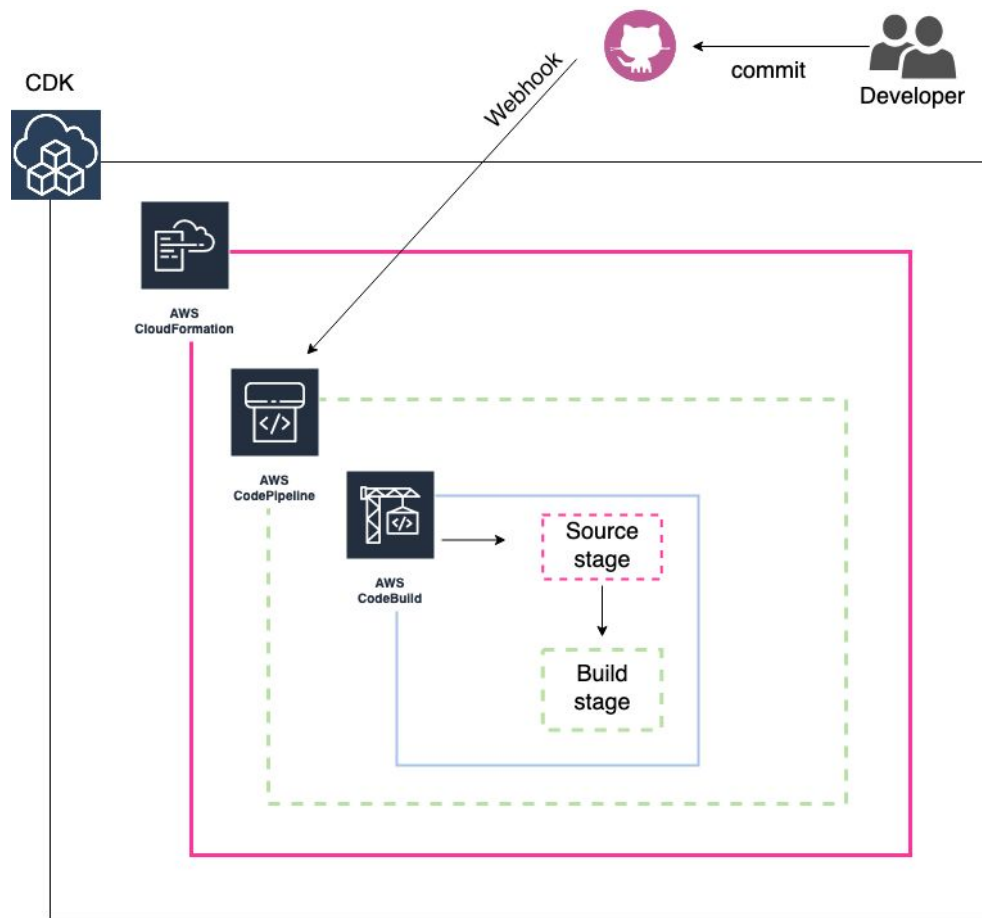
Where we begin

Currently we at ...



Codepipeline Codebuild

Codepipeline / Codebuild

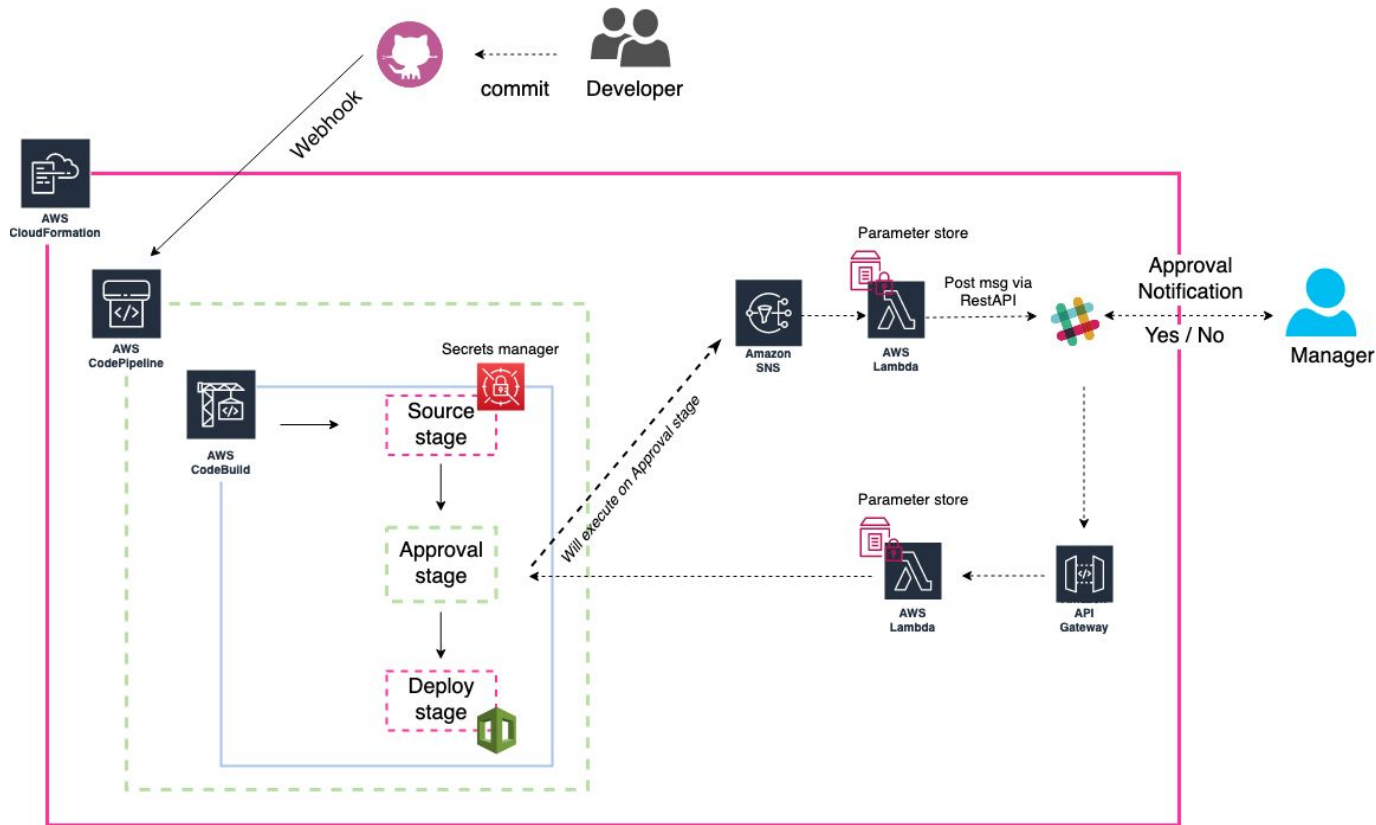


Codepipeline / Codebuild

```
1 // Init pipeline
2 const codePipeline = new codepipeline.Pipeline(this, 'Pipeline', {
3   pipelineName: 'pipeline',
4 });
5
6 const oauthSecret = cdk.SecretValue.secretsManager('bridge-githubAccessToken');
7
8 // Init Source stage
9 const sourceOutput = new codepipeline.Artifact('sourceOutput');
10 const sourceAction = new cpaction.GitHubSourceAction({
11   actionName: 'codebuild-action',
12   owner: environment.owner,
13   repo: environment.repo,
14   oauthToken: oauthSecret,
15   branch: `${environment.branch}`,
16   trigger: cpaction.GitHubTrigger.WEBHOOK,
17   output: sourceOutput,
18 });
19 sourceStage.addAction(sourceAction);
20 // Deploy stage
21 const codeBuildDeploy = new codebuild.PipelineProject(
22   this,
23   'Codebuild-Deploy',
24   {
25     role: buildRole,
26     environment: {
27       buildImage: goldenImage,
28       privileged: true,
29     },
30     buildSpec: codebuild.BuildSpec.fromSourceFilename('buildspecDeploy.yml'),
31   },
32 );
33 const deployAction = new cpaction.CodeBuildAction({
34   actionName: 'Deploy',
35   input: sourceOutput,
36   project: codeBuildDeploy,
37 });
38 deployStage.addAction(deployAction);
```

**Codepipeline
Codebuild
+
Approval stage
Slackbot**

Approval + Slackbot (Demo)



pipeline-dev

✓ **Source** Succeeded
Pipeline execution ID: d84da3c8-990a-4d60-9401-b81b22d39ea0

codebuild-action-dev ①

GitHub

✓ Succeeded - 14 hours ago

9a0fddfa

9a0fddfa codebuild-action-dev: init

Disable transition

✓ **Approval** Succeeded
Pipeline execution ID: d84da3c8-990a-4d60-9401-b81b22d39ea0

ApprovalOrDeny ①

Manual approval

✓ Approved - 14 hours ago

Details

9a0fddfa codebuild-action-dev: init

Disable transition

✓ **Deploy** Succeeded
Pipeline execution ID: d84da3c8-990a-4d60-9401-b81b22d39ea0

Deploy ①

AWS CodeBuild

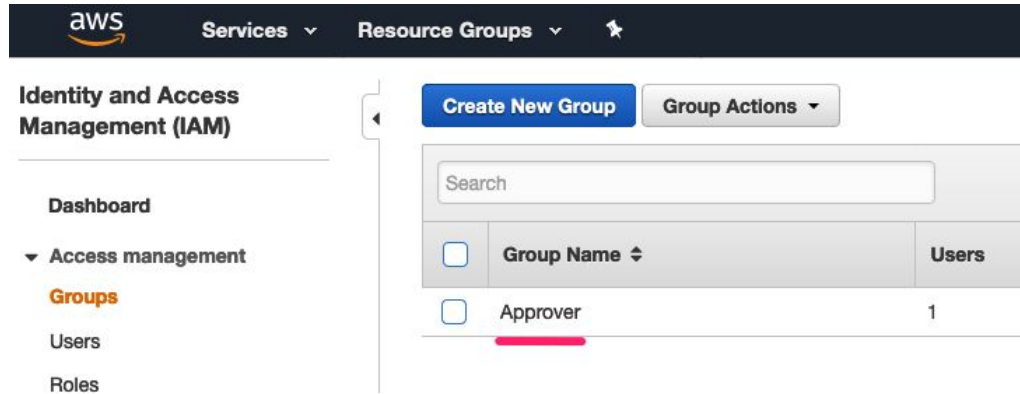
✓ Succeeded - 14 hours ago

Details

9a0fddfa codebuild-action-dev: init

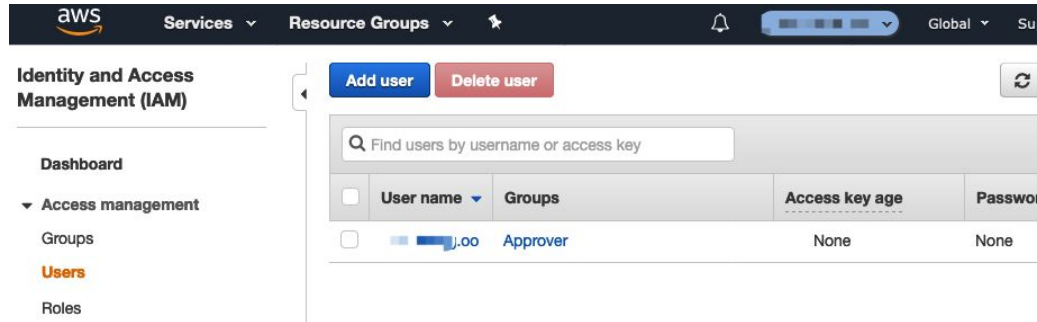
Approval

Create an IAM User name same as slack user ID, but without Access key and Secret.



The screenshot shows the AWS IAM console interface. The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', and a star icon. The left sidebar is titled 'Identity and Access Management (IAM)' and contains a menu with 'Dashboard', 'Access management' (expanded), 'Groups' (highlighted), 'Users', and 'Roles'. The main content area is titled 'Create New Group' and features a 'Create New Group' button and a 'Group Actions' dropdown. Below these is a search bar labeled 'Search'. A table lists existing groups with columns for selection, 'Group Name', and 'Users'. One group named 'Approver' is listed with 1 user.

	Group Name	Users
<input type="checkbox"/>	Approver	1



The screenshot shows the AWS IAM console interface for adding a user. The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', a bell icon, a refresh button, and 'Global'. The left sidebar is titled 'Identity and Access Management (IAM)' and contains a menu with 'Dashboard', 'Access management' (expanded), 'Groups', 'Users' (highlighted), and 'Roles'. The main content area is titled 'Add user' and features an 'Add user' button and a 'Delete user' button. Below these is a search bar labeled 'Find users by username or access key'. A table lists existing users with columns for selection, 'User name', 'Groups', 'Access key age', and 'Password'. One user named 'Approver' is listed with no access key age and no password.

	User name	Groups	Access key age	Password
<input type="checkbox"/>	Approver		None	None

Approval

Add SNS and Approval stage

```
1 const snsArn = ssm.StringParameter.fromStringParameterName(  
2     this,  
3     `cdk-arn-slackSNS`,  
4     '/CDK/arn/slack-sns',  
5 ).stringValue;  
6  
7 // Approve stage  
8 const ApprovalSNS = sns.Topic.fromTopicArn(this, `ApprovalSNS-${environment.ENV}`, snsArn);  
9 const approvalAction = new cpackaction.ManualApprovalAction({  
10     actionName: 'ApprovalOrDeny',  
11     notificationTopic: ApprovalSNS,  
12     externalEntityLink: `https://github.com/${environment.owner}  
    /${sourceAction.variables.repositoryName}/commit/${sourceAction.variables.commitId}`,  
13     additionalInformation: 'Please check Github commit before APPROVE!!',  
14 });
```

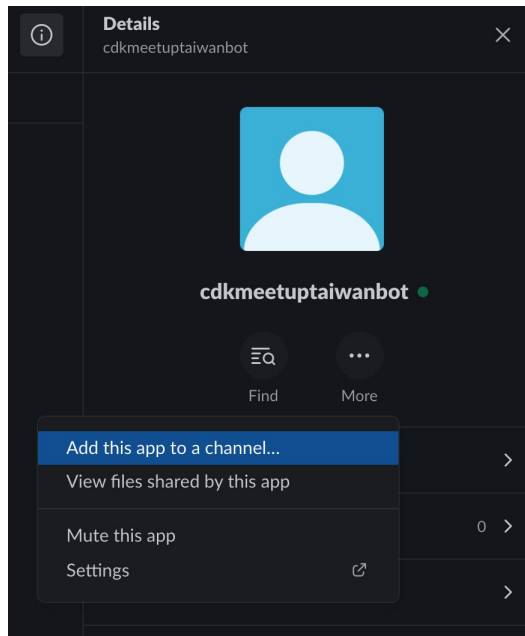
Scopes




A Slack app's capabilities and permissions are governed by the [scopes](#) it requests.

Bot Token Scopes

Scopes that govern what your app can access.

OAuth Scope	Description
channels:history	View messages and other content in public channels that CDK-Meetup-Taiwan-Bot has been added to
channels:read	View basic information about public channels in the workspace
chat:write	Send messages as @cdkmeetuptaiwanbot



Interactivity & Shortcuts

Settings

- Basic Information
- Collaborators
- Install App
- Manage Distribution
- Submit to App Direct...

Features

- App Home
- Incoming Webhooks
- Interactivity & Short...**
- Slash Commands

Interactivity

Any interactions with shortcuts, modals, or interactive components (such as buttons, select menus, and datepickers) will be sent to a URL you specify. [Learn more.](#)

Request URL

`tps://2-1.amazonaws.com/prod/bot`

Slack will send an HTTP POST request with information to this URL when users interact with a shortcut or interactive component.

Slackbot (Lambda [Send Message])

Create a lambda function to send approval message to slack and subscribe a SNS topic

```
1  const sendApprovalToSlackFunc = new lambda.Function(this, `CDK-SendApprovalToSlack-Func`, {
2    logRetention: log.RetentionDays.ONE_MONTH,
3    role: lambda.ExeRole,
4    functionName: `CDK-sendApprovalToSlack`,
5    runtime: lambda.Runtime.PYTHON_3_8,
6    code: lambda.AssetCode.fromAsset('source'),
7    handler: 'sendApprovalToSlack.handler',
8    environment: {
9      ENV: environment.ENV
10   }
11 });
12
13 // Enable a SNS for Lambda
14 const CDKApprovalSNS = new sns.Topic(this, `CDK-Approval-SNS`, {
15   displayName: `CDK-Approval-SNS`
16 })
17
18 CDKApprovalSNS.addSubscription(new subs.LambdaSubscription(sendApprovalToSlackFunc))
```

Slackbot (Lambda [Receive Message])

Create another Lambda Function to receive response from slack

```
1 const approvalTriggerFunc = new lambda.Function(this, `CDK-ApprovalTrigger-Func`, {
2     logRetention: log.RetentionDays.ONE_MONTH,
3     role: lambdaExeRole,
4     functionName: `CDK-approvalTrigger-func`,
5     runtime: lambda.Runtime.PYTHON_3_8,
6     code: lambda.AssetCode.fromAsset('source'),
7     handler: 'InteractiveMessageHandler.handler',
8     environment: {
9         ENV: environment.ENV
10    }
11 });
```


Slackbot (Apigw)

Create an Apigw with path `bot`

```
1 const api = new apigateway.RestApi(this, `CDK-ApprovalTrigger-Slackbot`, {
2     restApiName: `CDK-ApprovalTrigger`,
3     description: "A gateway trigger Codepipeline approval action through slack bot."
4 });
5
6 const approvalTriggerFuncLambda = new apigateway.LambdaIntegration(approvalTriggerFunc, {
7     proxy: true,
8 })
9 const rootPath = api.root.resourceForPath('/bot')
10
11 rootPath.addMethod('POST', approvalTriggerFuncLambda);
```

Slackbot (Inside Lambda)

Grab slack channel ID and Token from Parameter Store

```
1 ENV = os.environ['ENV']
2 SLACK_API_URL = "https://slack.com/api"
3 ssm = boto3.client('ssm')
4
5 def get_ssm_secret(parameter_name):
6     return ssm.get_parameter(
7         Name= parameter_name,
8         WithDecryption=False
9     )
10
11 slack_channel = get_ssm_secret("/CDK/arn/slack-channel")
12 slack_bot_token = get_ssm_secret("/CDK/arn/slack-token")
13
14 SLACK_BOT_TOKEN = str(slack_bot_token["Parameter"]["Value"])
15 SLACK_CHANNEL = str(slack_channel["Parameter"]["Value"])
```

Slackbot (Inside Lambda)

Receive message from SNS and send to Slack

```
1 def handler(event, context):
2     message = event["Records"][0]["Sns"]["Message"]
3
4     slack_message = {
5         "channel": SLACK_CHANNEL,
6         "text": "An approval is waiting for approve.",
7         "blocks": [
8             {
9                 "type": "section",
10                "text": {
11                    "type": "mrkdwn",
12                    "text": f"An approval is waiting for approve.🔴 \n Pipeline:
<{approvalConsoleLink}|{codepipeline_name}⚡ > \n Status: Waiting for approval \n Github
review: {GithubConsoleLink} \n"
13                }
14            }
15        ],
```

Slackbot (Inside Lambda)


User validation and detect Yes / No

```
1 for user in approvalUser["Users"]:
2     validUser = user["UserName"]
3     print(f'user: {user["UserName"]}')
```

4

```
5     if slackApproveState["approve"] is True and validUser == currentUser:
6
7         print(
8             f"User {currentUser} is valid approver and pressed ALLOW, {validUser}"
9         )
10        send_slack_message(slackApproveState)
11        return {
12            "isBase64Encoded": "false",
13            "statusCode": 200,
14            "body": f"Approved by {currentUser}, App is now ready to deploy \n pipeline:
{pipelineName} ",
15        }
16
17    if slackApproveState["approve"] is False and validUser == currentUser:
18        print(
19            f"User {currentUser} is valid approver and pressed REJECT, {validUser}"
20        )
21        send_slack_message(slackApproveState)
22        return {
23            "isBase64Encoded": "false",
24            "statusCode": 200,
25            "body": f"Rejected by {currentUser}, App deployment has been stopped. \n
pipeline:{pipelineName}",
26        }
```


Slackbot



cdkmeetuptaiwanbot

APP

3:10 PM


An approval is waiting for approve. 


Pipeline: [pipeline-dev](#) ⚡

Status: Waiting for approval

Github review: <https://github.com/ookangzheng/taipei-cdk-meetup-2-app/commit/9a0fddfafd3cb5856f40f0a216e8249d054e3ad3>

Yes to deploy

Yes 

No 

Are you sure?

✕

This action will deploy your app and can't be undo!!

No

Yes

Developer Tools

CodePipeline

Source • CodeCommit

Artifacts • CodeArtifact

Build • CodeBuild

Deploy • CodeDeploy

Pipeline • CodePipeline

Gettng started

Pipelines

Pipeline

History

Settings

Settings

Go to resource

Feedback

Developer Tools > CodePipeline > Pipelines > pipeline-dev

pipeline-dev

Notify

Edit

Source Succeeded

Pipeline execution ID: 66deaa6e-01b0-43df-b8d3-1f4f1ea9ad9b

codebuild-action-dev

GitHub

Succeeded - Just now

9a0fddfa

9a0fddfa codebuild-action-dev: init

Disable transition

Approval Pending

Pipeline execution ID: 66deaa6e-01b0-43df-b8d3-1f4f1ea9ad9b

ApprovalOrDeny

Manual approval

Waiting for approval -

Review

9a0fddfa codebuild-action-dev: init

Disable transition

Deploy Succeeded

Pipeline execution ID: d84da3c8-990a-4d60-9401-b81b22d39ea0

Deploy

AWS CodeBuild

Succeeded - 14 hours ago

Details

9a0fddfa codebuild-action-dev: init

Approval + Slackbot

Yes to deploy Yesterday ▾

Yes 🗳️ No 🗳️

V2-Test-BridgeBot APP 11:11 AM
Bridge-Backoffice-SES-Codebuild-Deploy-test:9301db45-01f1-4427-a1b7-4475e895892a
Region: ap-northeast-1
Build status: Successful
Time: 2020-08-10 11:11:56.679445+08:00 (Taipei)

cdkmeetuptaiwanbot APP 4:08 PM
was added to sre-testing by kangzheng.oo.

cdkmeetuptaiwanbot APP 5:19 PM
Approved by kangzheng.oo, App is now ready to deploy
pipeLine: pipeline-dev (edited)

CDK Meetup Approval Bot APP 6:41 PM
was added to sre-testing by kangzheng.oo, along with cdkmeetuptaiwanbot.

Today ▾

cdkmeetuptaiwanbot APP 12:00 AM
Approved by kangzheng.oo, App is now ready to deploy
pipeLine: pipeline-dev (edited)

cdkmeetuptaiwanbot APP 12:17 AM
Approved by kangzheng.oo, App is now ready to deploy
pipeLine: pipeline-dev (edited)

cdkmeetuptaiwanbot APP 12:36 AM
Approved by kangzheng.oo, App is now ready to deploy
pipeLine: pipeline-dev (edited)

cdkmeetuptaiwanbot APP 3:10 PM
An approval is waiting for approve. 🚨
Pipeline: pipeline-dev ⚡
Status: Waiting for approval
Github review: <https://github.com/ookangzheng/taipei-cdk-meetup-2-app/commit/9a0fddfa3cb5856140f0a216e8249d054e3ad3>

Yes to deploy

Yes 🗳️ No 🗳️

🔗 Message sre-testing

pipeline-dev

Notify ▾ Edit Stop execution Clone pipeline Release change

Source Succeeded
Pipeline execution ID: 66dea6e-01b0-43df-b8d3-1f4f1ea9ad9b

codebuild-action-dev ①
GitHub
Succeeded - Just now
9a0fddfa
9a0fddfa codebuild-action-dev: init

Disable transition

Approval Pending
Pipeline execution ID: 66dea6e-01b0-43df-b8d3-1f4f1ea9ad9b

ApprovalOrDeny ①
Manual approval
Waiting for approval -
Review
9a0fddfa codebuild-action-dev: init




Disable transition

Deploy Succeeded
Pipeline execution ID: d84da3c8-990a-4d60-9401-b81b22d39ea0

Deploy ①
AWS CodeBuild
Succeeded - 14 hours ago
Details

Recap

Recap

1. Get slackbot with api key ready, add it to Parameter Store via AWS CLI
2. Deploy slackbot stack with ``slackchannelID``
3. Invite slackbot into group
4. Paste in Apigw URL e.x: <https://xxx.ap-northeast-1.com/prod/bot>
5. Deploy codebuild stack with your *Github Token, Org Name, Repository Name*
6. Create a Secrets Manager via AWS CLI to store Github Personal Token
7. Wait for magic happen...   

Thanks

We're hiring !!!