

IP制限のかかっている
GitHub Organizationで
権限がほとんどないのに
GitHub連携関連の仕事をした話

高橋隆太



自己紹介

仕事

- インフラエンジニア(AWSが得意でIaC業務多めでサービスとしてはECSが得意)
- 過去にフロントエンドもバックエンドも浅くやってきたので、人手不足の際は協力も結構している

趣味

- 2次元イラストが好き(コミケ行ってきた) →
- たまに音ゲー(太ってるけどDance Dance Revolution が好き)

一言

- プレゼン資料全然書かないから優しくしてください
- これもグーグルテンプレの背景



LTテーマを一気に

エンジニア/PMになったきっかけ

- 大学時代卒業研究でIT系になったから(他の研究室行きたくなかったなので消去法でも今は好き)

こんな技術好き、知ってほしい (個人的)

- IaC が好きでアプリケーション系のエンジニアよりコードが短いのに(私の主観)
書いた内容が表現されるのが好き

生成AIの活用事例

- 時代と逆行して公式ドキュメントが多い
- たまに詰まった時にアイデア出しや、たまにコードとかめんどくさい時に出力してもらって修正など

個人開発について

- 経験なし

IP制限のかかったGitHub連携を頑張ったぞ

仕事上の立ち位置

- メンバー(プロパーじゃない)
- GitHubのAdminはプロパーの方のみ

経緯と状況

1. 前の案件でGitHub連携関係の仕事をAdmin権限持ちプロパーに持っていかれた
→ やって見たらそれが正しいことだったと思った
2. AWS Amplify とか CICDとかやってみたいから立候補した (事前準備もしてた)
→ リーダー層含めて他のメンバーはあまり知見がなかった

IP制限のかかったGitHub連携を頑張ったぞ

用語説明

- ホステッドランナー
→ GitHub Actionsが動作するサーバーのような環境(ubuntuなど)
- AWS Amplify
→ フロントエンドアプリケーションを簡単にビルド・デプロイ
(バックエンド系の認証とかもあるけど今回はこれだけ)
- GitHub Apps (AWSの)
→ GitHubリポジトリと連携し、CI/CDパイプラインや自動化タスクとよしなに連携してくれる

作るべきもの

- GitHub Apps利用したAWS との連携
- GitHubで動作するホステッドランナーの作成
- Amplify を利用したフロントエンドをデプロイ

GitHub Appsを利用したAWSとの連携

できなかったこと

- AWS CodeConnectionsとGitHubとの連携(Admin権限がなかった)
 - AWSとGitHubの接続はWEB画面じゃないとできない
 - アプリのインストールはメンバー権限じゃ無理...

 **AWS Connector for GitHub**
🕒 Installed 2 months ago 👤 Developed by [aws](#) 🔗 <https://docs.aws.amazon.com/dtconsole/latest/userguide/welcome>

Enables you to connect GitHub with AWS

Permissions

- ✓ **Read** access to actions, deployments, environments, issues, and metadata

- ✓ **Read and write** access to administration, code, commit statuses, pull requests, and repository hooks

GitHub に接続する

GitHub 接続設定 ⓘ

接続名

アプリインストール - オプション
GitHub App をインストールしてボットとして接続します。または、自分のままにして GitHub ユーザーとして接続し、AWS CodeBuild プロジェクトで使用することもできます。

または

▼ タグ - オプション

GitHub Appsを利用したAWSとの連携

できたけど本当は必要と推測する設定 (IP制限が原因) ※ すでに実施されていた

- AWSのサイトのIP許可リストをGitHub に登録する(忙しいプロパーの方に登録をお願い)

許可リストに追加する IP アドレス

[↓ PDF](#)[↓ RSS](#)☐ フォーカスモード

① 翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。

IP フィルタリングを実装するか、Amazon EC2 インスタンスで特定の IP アドレスを許可する場合は、以下の IP アドレスを許可リストに追加します。これにより、GitHub や Bitbucket などのプロバイダーへの接続が可能になります。

次の表に、デベロッパーツールコンソールの接続用の IP アドレスを AWS リージョン別に一覧表示します

① 注記

欧州 (ミラノ) リージョンの場合、このリージョンを使用する前にリージョンを有効にする必要があります。詳細については、「[リージョンの有効化](#)」を参照してください。

リージョン	IP アドレス
アジアパシフィック (東京) (ap-northeast-1)	52.196.132.231、54.95.133.227、18.181.13.91

GitHub Appsを利用したAWSとの連携

やったこと

- AWS CodeConnectionsとGitHubとの連携を忙しいプロパーの方におすすめする

GitHubで動作するホステッドランナーの作成

できなかったこと

- GitHub Actionsでランナー部分 runs-onでubuntu-latestを指定する(IP制限が原因)

name: Hello World

on: [push]

jobs:

 Hello-World-Job:

 runs-on:

- ubuntu-latest

 steps:

- run: echo "Hello World"

GitHubで動作するホステッドランナーの作成

やったこと1

- CodeBuildでランナーを作成して runs-onでCodeBuildを指定する

name: Hello World

on: [push]

jobs:

 Hello-World-Job:

 runs-on:

 - codebuild-!t-0911-\${{ github.run_id }}-\${{ github.run_attempt }}

 steps:

 - run: echo "Hello World"

GitHubで動作するホステッドランナーの作成

設定内容

1. CodeBuild の画面からランナープロジェクトを選んで選択する
(IP制限ない場合はここまでOK IP制限ある場合はエラーになる)

→ GitHub画面で設定へ

プロジェクトの設定

プロジェクト名

tk-0909

プロジェクト名は 2〜255 文字にする必要があります。アルファベット (a〜z、A〜Z)、番号 (0〜9)、特殊文字 (- と _) を含めることができます。

プロジェクトタイプ

作成するプロジェクトのタイプを選択します。 [情報](#)

☐ デフォルトプロジェクト
カスタム CodeBuild プロジェクトを作成します。

☒ ランナープロジェクト
GitHub Actions、GitHub Enterprise Actions、GitHub、または BuildKit (オプション) に CodeBuild マネージドランナーを作成します。

▶ 追加設定
説明、パブリックビルドアクセス、ビルドパック、開発ビルド制限、タグ

2. 設定内容で追加設定部分の手動作成にチェックを入れて登録する

▼ 追加設定

手動作成, ウェブフックイベントフィルター

手動作成 - オプション [情報](#)

☐ このリポジトリのウェブフックを GitHub コンソールで手動で作成します。

GitHubで動作するホステッドランナーの作成

設定内容

3. 取得した設定をGitHubに登録する

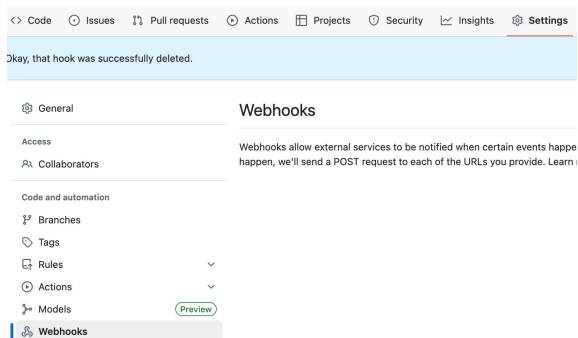
ウェブフックの作成

GitHub リポジトリのウェブフックを作成する必要があります。

ペイロードの URL

`https://codebuild.ap-northeast-1.amazonaws.com/webhooks?t=ε`

シークレット



GitHubで動作するホステッドランナーの作成

やったこと2

- 忙しいプロパーの方にWebhooks登録をお願いする

Amplifyを利用したフロントエンドのデプロイ

できなかったこと

- 下記のようなamplify.yml を利用したGitHubから直接のビルド・デプロイ(IP制限が原因)

```
version: 1
frontend:
  phases:
    preBuild:
      commands:
        - npm install
    build:
      commands:
        - npm run build
  artifacts:
    baseDirectory: dist
    files:
      - '**/*'
  cache:
    paths:
      - node_modules/**/*
```

Amplifyを利用したフロントエンドのデプロイ

やってみたけど失敗したこと

- 一旦IPで0.0.0.0/0を許可 → 実行 → Webhooksが登録される → 0.0.0.0/0塞ぎ → 再実行
→ CodeBuildのランナーと違ってどうやら毎回GitHubと接続しに行っていたらしい
※もちろんここの忙しいプロパーに依頼
※なぜかこのエラーはIAM関連の権限不足みたいなエラーが出る、、、(ここで半日時間潰した)
- GitHub のログからIPを確認登録 → 失敗 → AWS へ問い合わせ
結論としてAmplify からのIPは公開しておらず、ログで見つけても将来変える可能性があるとのこと

Amplifyを利用したフロントエンドのデプロイ

やったこと

- CodeBuildホステッドランナーから CICDパイプラインを構築して Amplifyでデプロイ
※AmplifyはGitHubがなくてもS3にアーティファクト dist/ みたいなのをおけばデプロイできた
※なおS3はKMS暗号化したらエラーになった(原因不明、開発中だから後で調査します)

下記のようなCICDパイプライン を作成

1. ビルド (npm run build など)
2. S3 sync (aws s3 sync dist/ s3://0911lt/front)
3. Amplify job (aws amplify start-deployment)

Admin権限仕事をお願いする時の心掛け!!

お願いする時

- 簡潔に何をなんのためにやるかを説明する
(急ぎの時はこれができないことへの影響も細かく伝える)

優先度を考える

- 期限を守ることが一番大事なので頻繁にお願いするときは事前連絡やカレンダーの空きに通話しながら一気にやるなどを考える

普段から

- この人がいうなら・・・みたいな人間になろう

所感

- コミュニケーションコストがかなりあるので、
可能ならAdmin権限を持った方にやってもらったほうがいい(マジで)
- どうしてもGitHub系の業務をしたいなら、
普段からAdmin権限持ちの人にいい顔しておこう
- 実はまだこの業務残っているからこれからもペコペコするかも
- でも楽しかった!!