

GitHub Flowで作る AWSインフラストラクチャ

AWS Dev Day Tokyo 2018 Lightning Talk大会

October 31, 2018

Isao Shimizu

About me

清水 勲 @isaoshimizu

株式会社ミクシィ

ヴァンテージスタジオ みてね事業部 開発グループ

- 2011年8月 中途入社、SNS mixi アプリ運用チーム
- 2014年4月 モンスターストライク サーバーエンジニア、SRE
- 2018年2月 家族アルバム「みてね」 SRE

AWS Summit Tokyo, TechLION, hbstudy, Internet Week, Think Japan IBM Code Dayなどの登壇

[Qiita](#), [Medium](#), Software Design 記事執筆

記事: [大規模インフラのSREから社内スタートアップへの挑戦。ベテランエンジニアのキャリアの作り方](#)



300万人が使う No.1 家族アルバムアプリ 「みてね」

- <https://mitene.us/>
- 特徴
 - 子供の写真や動画を共有、整理
 - 家族だけで安心して使えるアルバムアプリ
 - フォトブック、DVDの販売
- 2015年4月 サービス開始
- 2017年7月 英語版提供開始
- 2018年7月 ユーザー数 300万人突破

AWSのインフラ管理、どうやっていますか

AWSでのインフラ管理といえば

- AWS マネジメントコンソール
 - ウェブベースの管理ツール
 - [AWS Console モバイルアプリ](#)もある
- AWS CloudFormation
 - JSON/YAMLファイルの記述でAWSリソースの環境構築
- HashiCorp Terraform
 - .tfファイルの記述で様々なクラウドリソースの環境構築

マネジメントコンソールだけ使っていると起きること

- どんな手順で構成したかは操作した人しかわからない
 - Wikiやドキュメントツールなどを使って手順書を書く
 - スクリーンショットなどが無いと伝わりづらい内容もある
- 手順の確認、レビューがしづらい
 - Slackにスクリーンショットを貼って「これ実行します！」
- 同じ手順の繰り返し
 - 多くの手間

プログラマならコードで管理したい

Terraformを使う

Terraformを使う理由

- AWS以外の管理もできる
 - たとえば、Google Cloud、New Relic、GitHubの管理
 - 90以上のサービスに対応
 - <https://www.terraform.io/docs/providers/>
- <https://github.com/hashicorp/terraform>
 - スター数 14,000超え
 - 毎日のようにコミットがされ開発が活発
- "AWSだけ" ならCloudFormationを使うのはアリ
 - AWS SAM(Serverless Application Model)を使うならなおさら

Terraformの課題

Terraformをどこで実行するか

- `terraform` コマンドをどこの環境で実行するのか
 - 複数の開発者が実行するのは怖い
 - 実行の競合による先祖返り
 - 手元のPCで実行する場合、強力なアクセスキーが必要
 - EC2上で実行する場合、SSHの権限が悩ましい
 - 最近だとAWS Cloud9、AWS Systems Manager（セッションマネージャー）とかありかも？
 - しかし手作業は面倒

workspaceを使うか、ディレクトリを分けるか

- みてねでは1レポジトリでディレクトリを分ける運用
- workspaceは後述するCI/CD環境に不向き
 - 複数のworkspaceを切り替えて自動化させるのは複雑になりそう
- 更新があったtfファイルが存在するディレクトリをスクリプトで検知して、対象のディレクトリのみでTerraformを実行するように工夫
 - gitコマンドを駆使して実行対象のディレクトリを抽出して個別に実行

自動化しよう

CI/CD環境をTerraformの実行環境とする

GitHubと連携させて自動化する（GitHub Flow）

- tfファイルを書く
- Pull Requestを作る/commitをpushで `terraform plan` 実行
- レビュー、Approve
- Pull Requestマージで `terraform apply` 実行

GitHubレポジトリの設定

- ✓ `Require pull request reviews before merging`
- ✓ `Require status checks to pass before merging`
- ✓ `Require branches to be up to date before merging`

CircleCIを使う

- .circleci/config.ymlを書く
- GitHubのWebhookで連携
- ググるとサンプルも多くて比較的簡単に設定できる
- Workflowや対象のBranch指定が柔軟にできる
- クレデンシャル情報の登録
 - AWSの認証情報を設定する画面が用意されている
 - 設定後はキーの表示がマスクされるがCircleCIの権限（GitHubの権限）の管理をちゃんとしておかないと不安
- コンテナ数に応じた月額課金（1個無料。2個～は\$50/コンテナ）

AWS CodeBuildを使う

- buildspec.ymlを書く (`version: 0.2` で書こう)
- GitHubのWebhookで連携
- ビルドプロジェクトにIAMロールが使えるのが良い
- 実行環境で使える環境変数があるが、ドキュメントにすべてが書かれていないので `env` コマンドなどで確認すると良い
 - 例: CODEBUILD_AGENT_ENV_CODEBUILD_BUILD_URL (ビルドプロジェクトのURL)
- 分課金 (Linuxでのビルド、1分あたり\$0.005から) 。無料枠も。
- 3つのインスタンスタイプ (small、medium、large)

GitHubにTerraform実行結果の通知

tfnotify

<https://github.com/mercari/tfnotify/>

- GitHubのPull Requestのコメントに `terraform plan` や `terraform apply` の結果を書いてくれるメルカリのOSS（超便利!）
- CI/CD環境で実行
- いちいちログ画面を見に行かなくても良い

**tfnotifyが対応する環境はCircleCI
or TravisCIのみでしたが...**

昨日、AWS CodeBuildにも対応しました！

1週間前にCodeBuild対応のPull Requestを出したところ、昨日マージされました！（レビューありがとうございます！）

<https://github.com/mercari/tfnotify/pull/23>

Add support AWS CodeBuild #23

Merged b4b4r07 merged 2 commits into `mercari:master` from `isaoshimizu:add_support_codebuild` 18 hours ago

Conversation 5 Commits 2 Checks 0 Files changed 6



isaoshimizu commented 7 days ago

Contributor + 😊 ...

WHAT

Support AWS CodeBuild <https://aws.amazon.com/codebuild/>

WHY

Terraform execution with AWS CodeBuild is convenient because it can be executed under the IAM role. Environment variables used in CodeBuild's execution environment are written in this URL. <https://docs.aws.amazon.com/codebuild/latest/userguide/build-env-ref-env-vars.html>

👍 1

【補足】 TerraformをAWS CodeBuildで使う際のポイント

- git cloneの深さをFullにして過去のcommitを参照する
- ブランチフィルタにmasterを設定して、masterに向けたPull Requestのみ実行する
- Pull Requestが作られてマージされるまでは `CODEBUILD_SOURCE_VERSION` に `pr/番号` という形式の値が入ってくる。これを見て `terraform plan` を実行する
- Pull Requestのマージで `CODEBUILD_SOURCE_VERSION` はマージコミットのハッシュ値が入ってくる。これを見て `terraform apply` を実行する
- tfnotifyにPRの番号を渡せるように、マージ時に `git log` コマンドで `Merge pull request #100 from ブランチ名` といったコミット文言からPR番号を抜き出して、 `CODEBUILD_SOURCE_VERSION` にセットする

まとめ

- インフラの管理はできる限りコード化&自動化
- Terraformの実行環境はCI/CD環境がおすすめ
- GitHub Flowで運用する
- tfnotifyを使ってplan/applyの結果をGitHubへ通知
- tfnotifyはAWS CodeBuildにも対応しました
- AWS CodeBuildはIAMが使えて、実行時間課金なのがメリット

Join our team!!

<https://mitene.us/recruit>