SRA の技術情報誌
GS LetterNeo
Vol.159　2021 年 10 月

Jira x AWS 連携による AWS Account Deploy 自動化

杉本 匡光

アドバンストクラウドエンジニアリング事業部

はじめに

SRA では検証用途やシステム構築向けの AWS 環境の社内への提供、運用を行っています。この活動では、運用メンバーがこれまでに現場で培ったクラウド構築・運用のノウハウを活かし、社内に還元しています。

AWS アカウントを払い出し、管理するチームは専任ではないため、メンバーが少しでも楽できるために用意した Jira Service Management と AWS Service Catalog の連携事例を紹介いたします。

社内 AWS 環境について

社内に提供している AWS 環境は以下の通りです。

- 事業部・プロジェクト 単位で AWS アカウントを払い出す
- AWS アカウントは Organizations のメンバーとして組織に属する
- 用途に応じた Organizations の OU に属する
- 利用者はログイン後、対象アカウントに スイッチロール して環境を利用する
- SCP / AWS Config を利用し、適切なガードレールが設定される

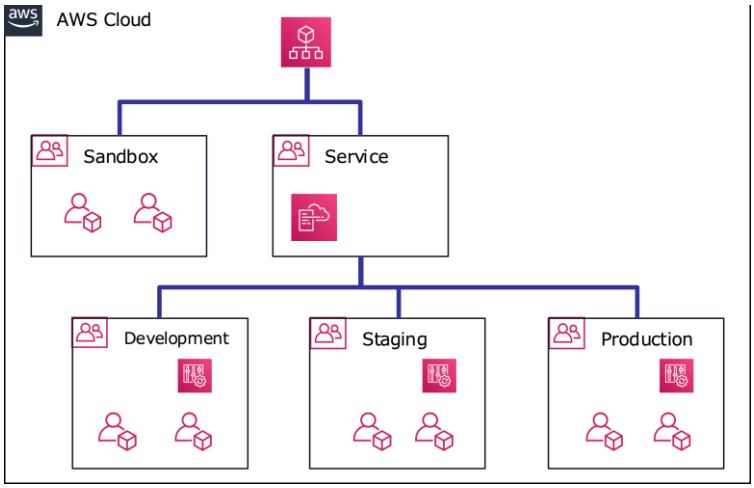

図 *1 AWS Organizations* 構成イメージ

運用開始当初はマネジメントコンソールから AWS アカウントの払い出しを行っていましたが、以下のような変遷でスクリプト化していきました。

1. AWS マネジメントコンソールから手動で払い出し
2. AWS CLI で払い出し
3. Python + Boto3 を使ったスクリプトで払い出し

払い出し作業はスクリプト化されましたが、メール受付、スクリプトの実行、メールでの完了通知等のコミュニケーションは手動のまま残り、さらなる効率化のために Jira Service Management と AWS の連携を試すことにしました。

Jira Service Management x AWS 連携

Jira Service Management と AWS Service Catalog を連携し、AWS アカウントの払い出し作業の自動化を行いました。払い出しに必要な AWS のサービスと流れは以下の通りです。

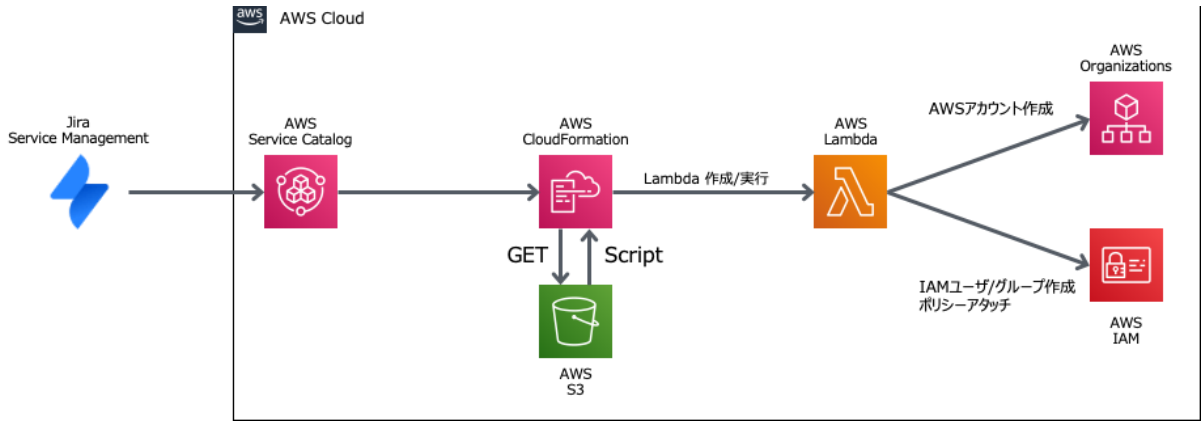

図 *2 Jira x AWS* 連携概要

AWS アカウント作成は Python + Boto3 のスクリプトを Lambda で実行させています。Lambda で実行する Python スクリプトは S3 に格納しておき、Service Catalog でデプロイされる CloudFormation の中で S3 からダウンロードし、Lambda にデプロイし実行しています。

AWS 側、Jira Service Management 側での設定内容は以下の通りです。

AWS 側

1. Jira Service Management Connector 用の権限設定のため、CloudFormation 実行
2. Service Catalog ポートフォリオを作成
3. Service Catalog ポートフォリオにプロダクトを追加
4. アカウント作成スクリプトを S3 にアップロード

Jira Service Management 側

1. AWS 連携用アプリ Jira Service Management Connector をインストール
2. Jira Service Management Connector の AWS 接続設定
3. Service Catalog のポートフォリオに対して、申請・承認権限を付与
4. Jira のプロジェクトに対し、Request Type 追加 / ポータルに追加

アカウント払い出しの流れ

連携の準備が出来たところで、実際の Jira のサービスポータルでの申請からアカウントの払い出しまでの流れを確認してみます。

1. 利用者が Jira のサービスポータルから AWS の申請を選択

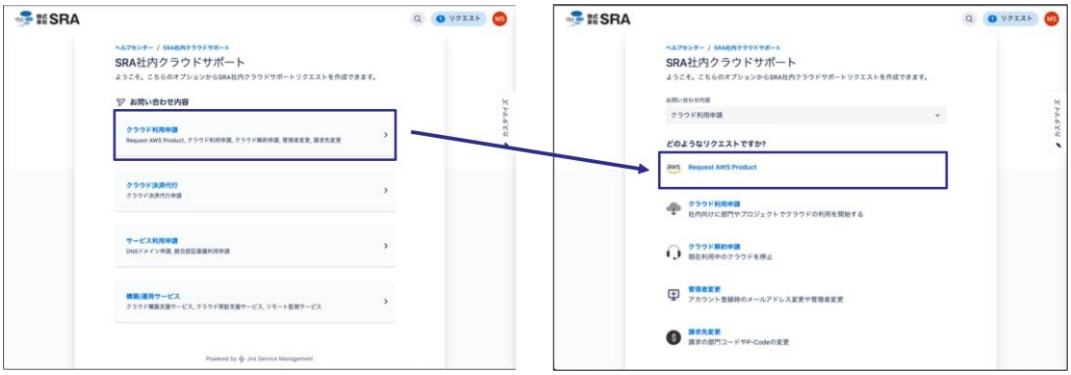

図 *3* ポータルでの申請

2. AWS アカウント名、メールアドレスなどを入力

ここで入力するアカウント名、メールアドレス などの変数は AWS Service Catalog Portfolio に登録した製品 の CloudFormation テンプレートで定義します。

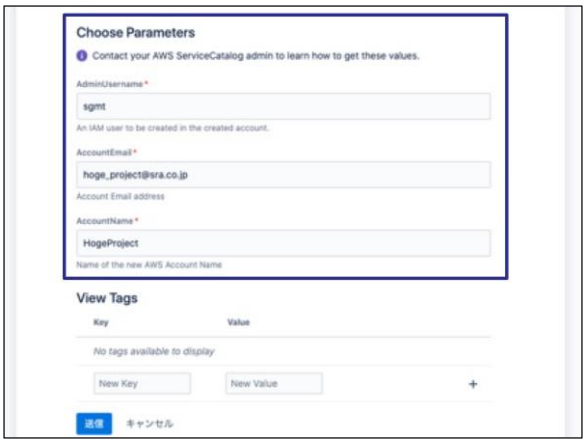

図 *4* ポータルでの申請内容の入力

3. 運用チームが申請内容を確認し、リクエストの承認

利用者が申請時に入力した内容が画面右側に表示されますので内容を確認し、「Make approval decision」をクリックします。

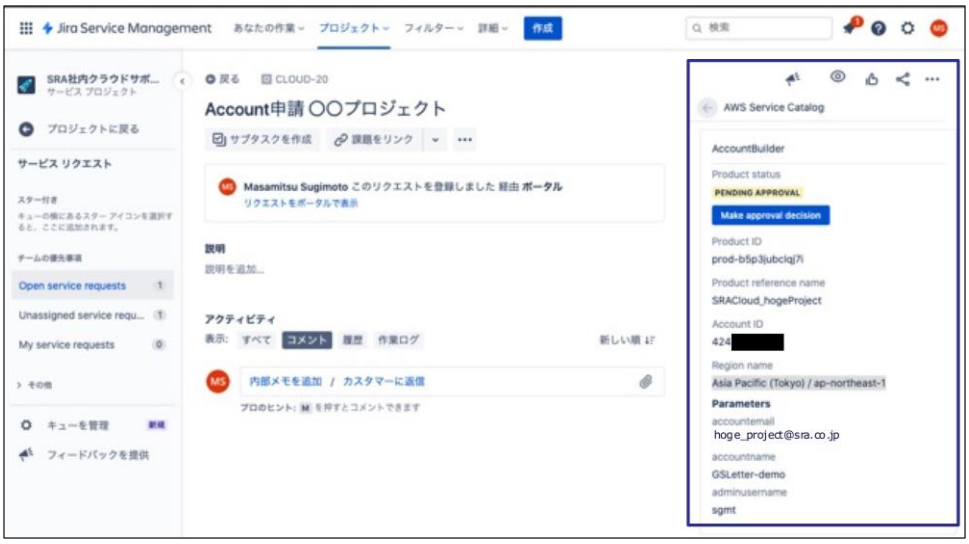

図 5 運用チームのリクエスト承認

4. 自動でアカウント払い出し

運用チームでリクエストが承認されると、AWS Service Catalog の 「製品」のデプロイが開始され、自動的にアカウントが払い出されます。
画像は Organizations で、対象の OU に属していることを確認した画面です。

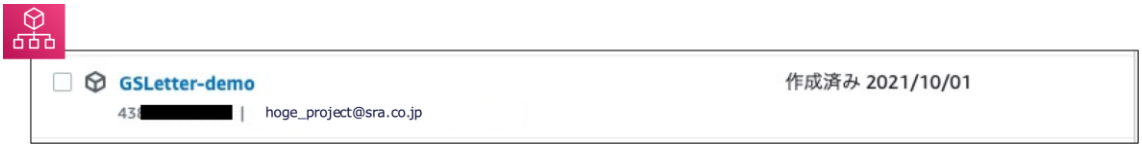

図 6 AWS アカウント Organizations での 表示

5. 利用者への通知

申請が承認されると Jira Service Management Connector から **PROVISIONING**, **PROVISIONED** それぞれのステータスでメッセージが通知されます。ステータスの更新時刻を見ると、承認から 2 分程度で AWS アカウントが払い出されていることが分かります。
最後に運用チームからログイン情報と利用上の注意が送信され、払い出し完了です。

図 7 利用者側 リクエストの表示

おわりに

以上、Jira Service Management と AWS Service Catalog を連携させた AWS アカウントの払い出しを行いました。この仕組みを利用することにより、マネジメントコンソール、AWS CLI で操作することが無くなり、運用チームメンバーの初期環境設定や、払い出し作業自体の煩わしさが無くなります。これに加え、CloudFormation StackSets で AWS Config, CloudTrail, VPC Flow Logs などのベースラインを定義しておけば、さらに幸せな世界が見えてきます。

参考 URL

Jira Service Management Connector for Jira

https://docs.aws.amazon.com/ja_jp/servicecatalog/latest/adminguide/integrations-jiraservicedesk.html

Automate account creation, and resource provisioning using AWS Service Catalog, AWS Organizations, and AWS Lambda

https://aws.amazon.com/jp/blogs/mt/automate-account-creation-and-resource-provisioning-using-aws-service-catalog-aws-organizations-and-aws-lambda/

GSLetterNeo Vol.159
2021 年 10 月 20 日発行
発行者　株式会社 SRA 先端技術研究所

編集者　熊澤努　方 学芬
バックナンバー https://www.sra.co.jp/public/sra/gsletter/
お問い合わせ　gsneo@sra.co.jp

株式会社SRA
〒171-8513 東京都豊島区南池袋 2-32-8

夢を。Yawaraka Innovation
やわらかいのべーしょん