GKEの新機能 Workload Identityで機密情報を安全管理

クラウドエース株式会社
富永 裕貴(Yuki Tominaga)

about me

クラウドエース株式会社 (2017/11 ~) リードエンジニア

光/闇の戦士 (とみなが ゆうき)

Google Cloud 認定トレーナー

好きなGCPサービス
Google Kubernetes Engine, Cloud Build, Dataflow

今日話すこと

GKEに登場した新機能 *Workload Identity* の紹介

- Workload Identity 登場前の課題
- Workload Identity 登場後の利便性
- Workload Identity 登場後の課題

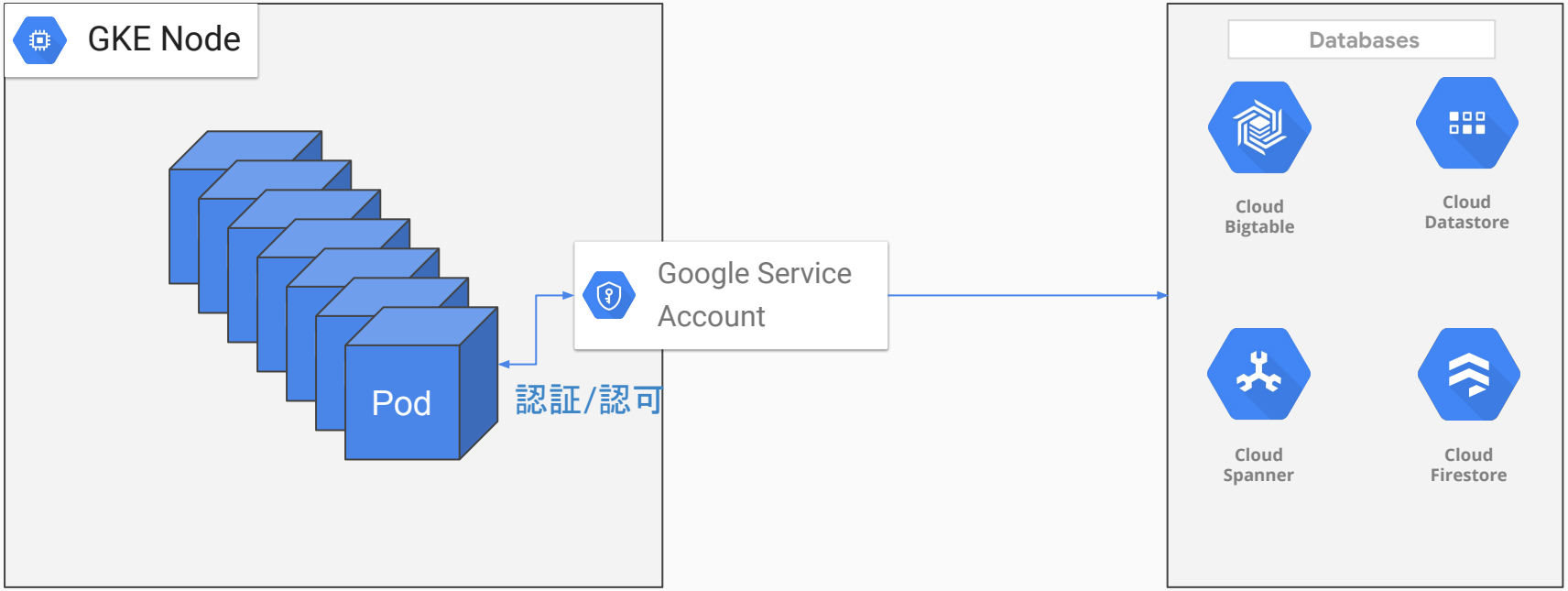
Workload Identityのための用語

- GSA (Google Service Account)
  GCPのサービスアカウント
- KSA (Kubernetes Service Account)
  Kubernetesのサービスアカウント
- RBAC (Role Based Access Control)
  Kubernetesにおける権限管理

GKEにおけるコンテナの認証

1. Nodeのサービスアカウントを使用する

Podが適切に特定のNodeにスケジュールされることを制御しないといけなくて辛い



GKEにおけるコンテナの認証

2. コンテナにサービスアカウントをマウントする

```
# サービスアカウント鍵ファイルを作って
$ gcloud iam service-accounts keys create credentials.json --iam-
account=sa@projects.iam.gserviceaccount.com --key-file-type=json

# 作った鍵ファイルをSecretに登録して
$ kubectl create secret generic super-secret-info --from-file=credentials.json=credentials.json
```

```
# Podにマウントする
spec:
  volumes:
    - name: volume-name
      secret:
        secretName: super-secret-info
  containers:
  - name: container-name
    image: image
    volumeMounts:
      - mountPath: /etc/credentials.json
        name: volume-name
```

- サービスアカウント鍵ファイルを作ること自体がリスク
- Secretを参照できれば機密情報にアクセス可能
- RBACでSecretの読み取りを制御するのもめんどい

Workload Identity

GSAとKSAをバインドしてPodにKSAを割り当てると、PodがGSAの役割を持てる

- サービスアカウント鍵ファイル(機密情報)を作る必要がない
- Kubernetesで実行されるコンテナの権限管理がGCPのサービスアカウント
- とにかくやってみよう! (ライブデモ

まとめとTIPS

- Workload Identity を使うとGCPへ安全に認証/認可ができるので、今後はこの方法を主流にしていきたい
- 設定が結構めんどくさい!!!
  Googleさんなんとかして!!!
- TIPS: TerraformやHelmを使って構成を自動化/テンプレート化しておくとこの辺の運用とても便利だったのでおすすめ

お知らせ

クラウドエース発「コンテナブースター」サービス提供開始

VMで実行されるモノリシックなアプリケーションをコンテナ化します

- Anthos によるGCPへの移行
- プロセス/サービス単位で分割しコンテナ化
- スケーラビリティを考慮した GCPでの運用提案/支援
- Stackdriver Monitoring / 弊社パートナー製品を利用した監視とアラートの設定
- セキュリティを考慮したネットワーク /アカウント設計

などなど、弊社コンテナ運用 1軍メンバーによるご支援をいたします。