企業の上場時に必要な監査要件とマネジメントサービスによる解決

3/16/2023 AWS Startup.fm

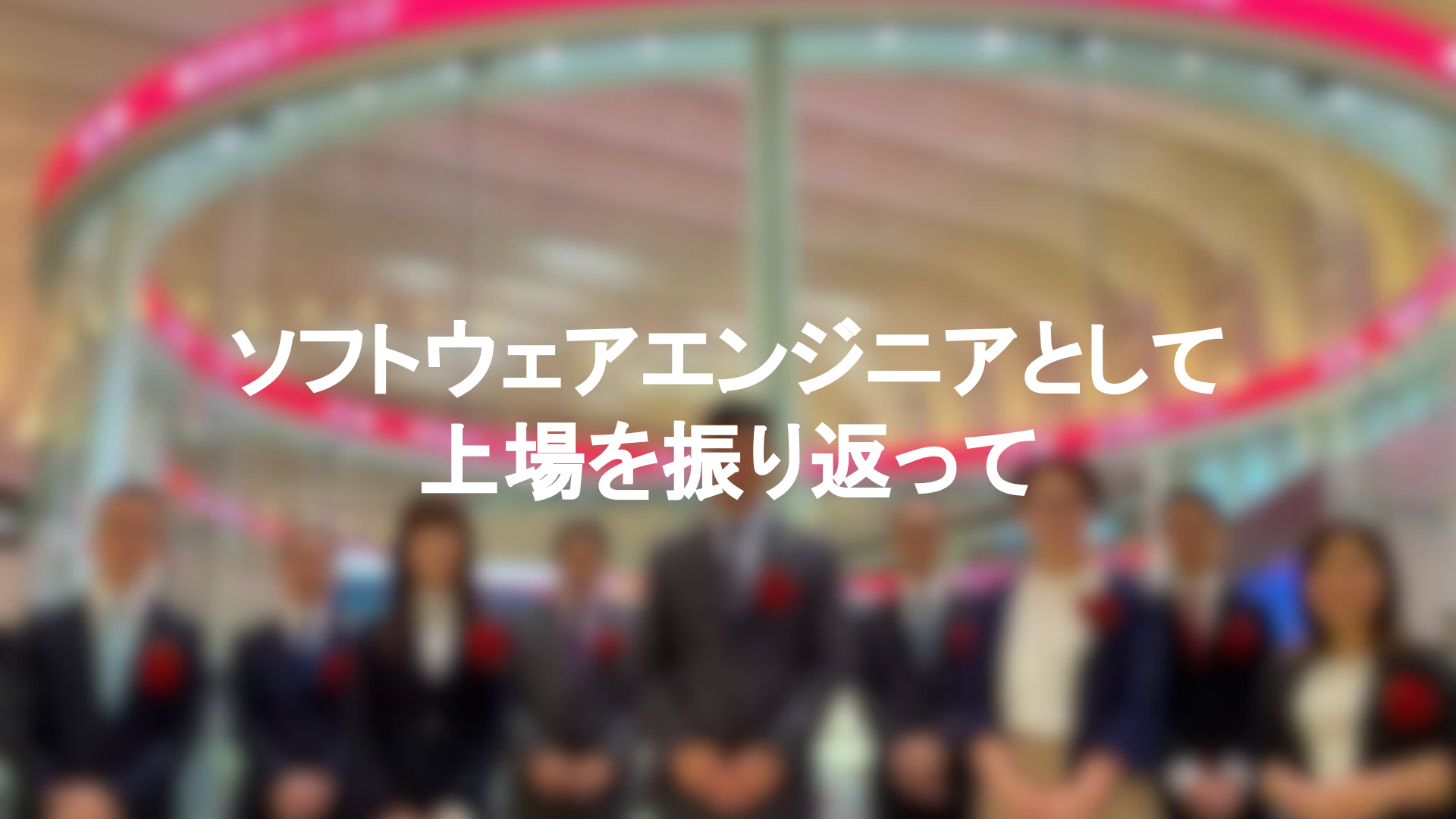
株式会社ヌーラボ
SRE 課 大野一樹

はじめに

- 自己紹介
  - 所属と名前: 株式会社ヌーラボ SRE: 大野一樹
  - 普段の業務: Backlog の Platform Engineering, EKS Admin
  - 経歴: 過去数社上場企業での勤務を経験
- 今日のおはなし
  - 現場のソフトウェアエンジニアから見た上場の振り返り
  - 上場までに必要な IT 全般統制と会計監査の監査要件
  - ヌーラボでの監査要件への具体的な対応と進め方の事例
  - 現場にとってよい監査対応の進め方

株式会社ヌーラボ

- 2004年設立、2022年6月に東証グロース市場上場
- Backlog, Cacoo を中心とした SaaS 企業


ソフトウェアエンジニアとして
上場を振り返って

上場の振り返り

- 結果として上場していた感覚(おまつり後のような感覚😂)
  - 上場の中心にはいなかった; 上場審査のため断片的にタスクを担当
  - 各種申請フローが厳格になっていく(上場企業での感覚を思い出す)
  - 審査から東証の上場まではリリースができない期間が設定された

    => システムを安定稼働させるため本番環境が触れない & バグ修正は OK😲
- 上場申請までに前年から対応が必要だったこと
  - VPoE, 管理部長(CFO), 情シス, セキュリティが中心のタスクフォースでの対応
  - 監査法人様との定期的なミーティング

    => 上場の要件に基づく会計監査、IT 統制への対応 📑

上場の振り返り

- 監査要件への対応
  - 上場監査要件を監査法人とタスクフォースで管理
  - タスク担当者を現場にアサインするトップダウン方式

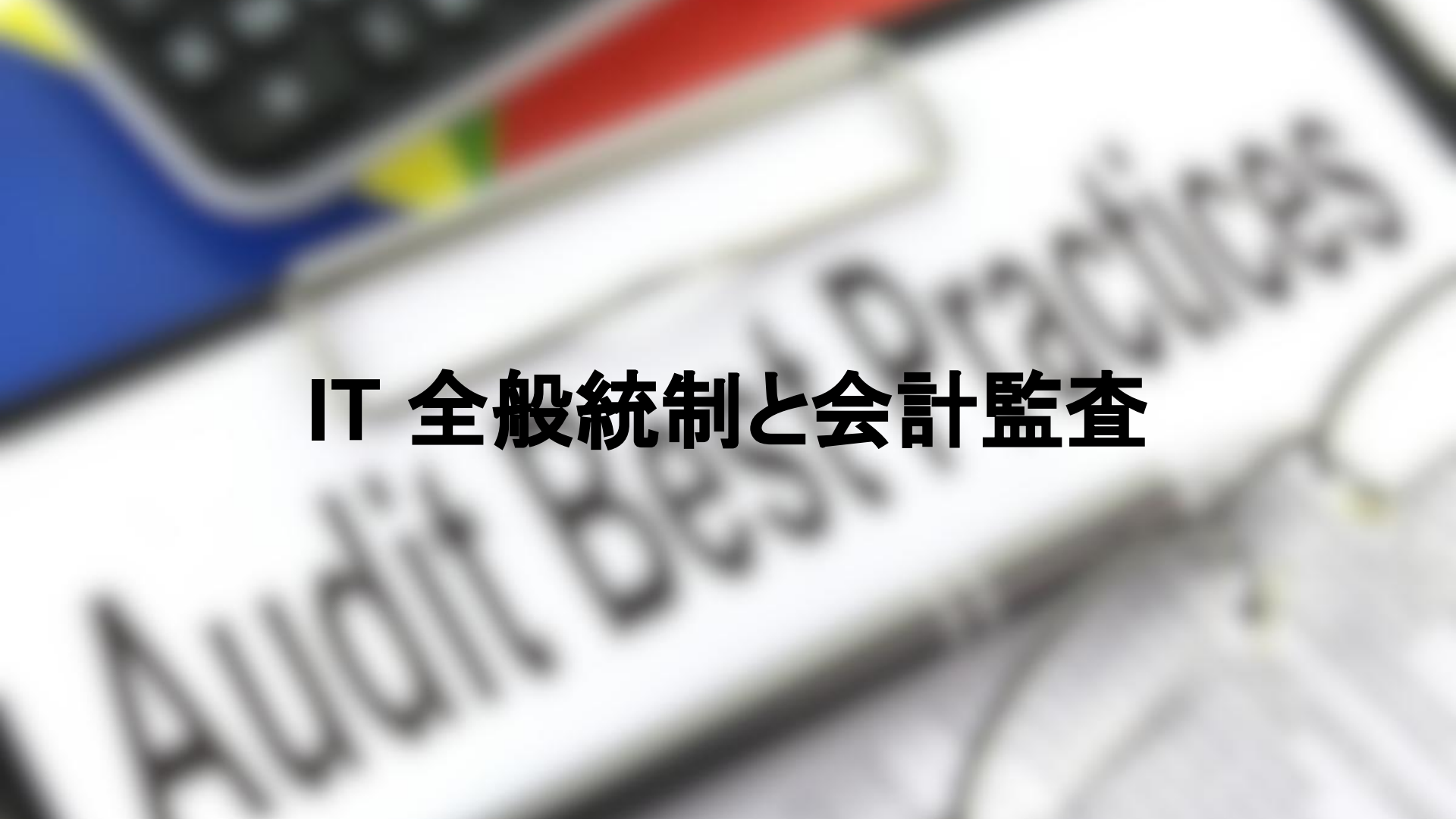
    => 技術要件や細かい仕様については監査法人と担当者で相談

    監査仕様を現場の裁量で決められて良かった😊※後ほど実例で紹介
  - ISMS, J-SOX の要件に対応するために上場してからも改善が必要

    => 最初からやっておいたほうがいい

    - 途中から対応すると、とにかく大変😩

    (期限が厳しい監査対応の業務&ソフトウェア仕様変更+普段の業務)



IT 全般統制と会計監査

なぜ監査が必要なのか？

- 上場審査、ISMS、セキュリティ、コンプライアンスへの対応(要件は異なる)
  - 必要最小限の権限にして乗っ取りやオペレーションミスによる被害防ぐ
  - アカウント管理、パスワードポリシーの策定
- 誰が、いつ、どこで、どのように、何をしたかを記録する
  - インシデントが発生したときの調査を行うため(リスク管理)

    顧客への説明、再発防止策の検討、データの復旧を行うため

ex.) -「誰が」コードを「いつ」「どのように」変更したか

-「誰が」「どこ」でコードの変更を承認したか

IT 全般統制と会計監査

- IT 全般統制
  - 社員の端末管理 => AD での MDM 管理(情シス業務)
  - IT サービスのアカウント・権限管理 ≒ AWS IAM アカウントの管理 etc.
  - インシデント発生時の対応調査準備 ≒ AWS 監査ログの取得
- 会計監査
  - 経理データの監査
  - 顧客データベースの監査 ≒ AWS マネージド DB に対する監査
  - 会計・請求処理に関するソフトウェアの監査 ≒ Git, CI/CD, ECR 管理

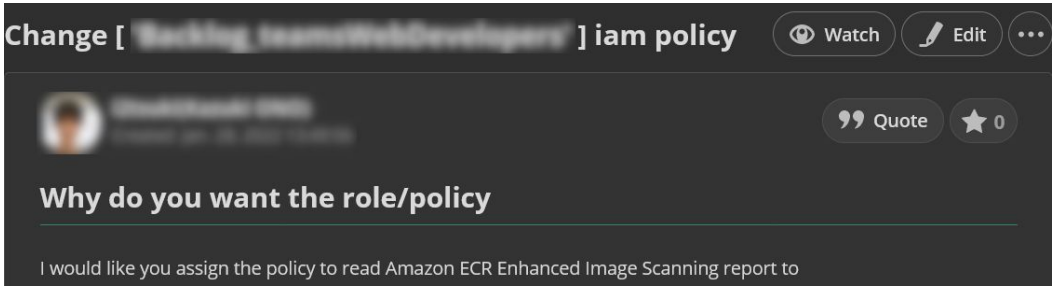
AWS サービス外の監査対応

nulab

AWS サービスの外の監査対応

IT 全般統制: IT サービスのアカウント・権限管理

- チケットシステム(Backlog)の Issue でアカウントの申請を証跡として管理
  - 全 IT サービスに基づくためフローやフォーマットは情シスが管理
  - 誰が何の権限を申請して、誰が承認したかの証跡を残す事が重要
    => 申請フォーマットは作業者/承認者が用意(ex. AWS だったら SRE)



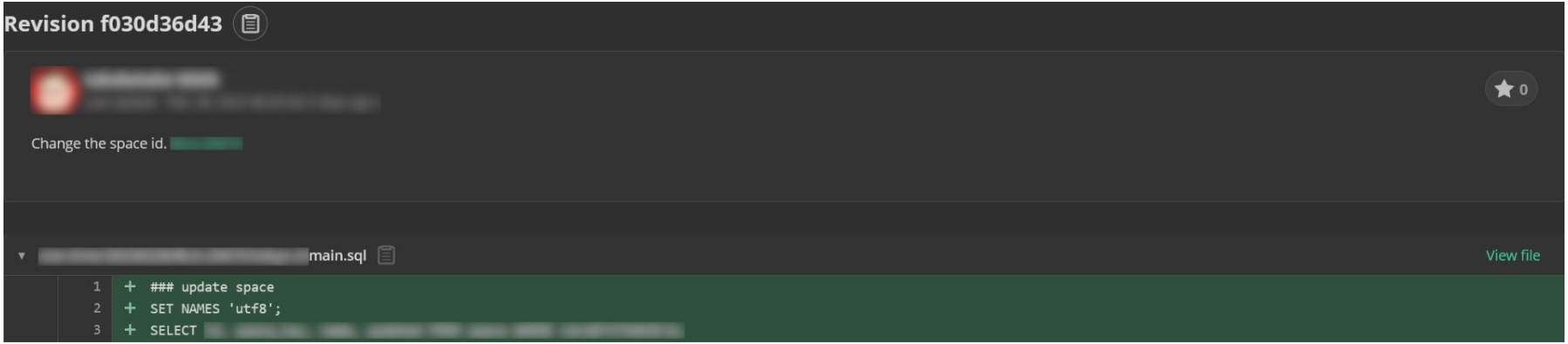
※あとからフローを整えることはできるが、既存のアカウントについて棚卸しが必要になるので、初めからやっておいたほうがいい

AWS サービスの外の監査対応

会計監査: 顧客、会計データベースの監査

- なぜやるのか？ => データが正しいこと、変更されていないことを保証する
- DB に対するマイグレーション、オペレーションの履歴、承認フローを残す

  リポジトリ管理して PR 形式で作成者、承認者を明確にする



- DB のデータを変更するバッチ作業は別途部課長の承認が必要

AWS サービス内の監査対応

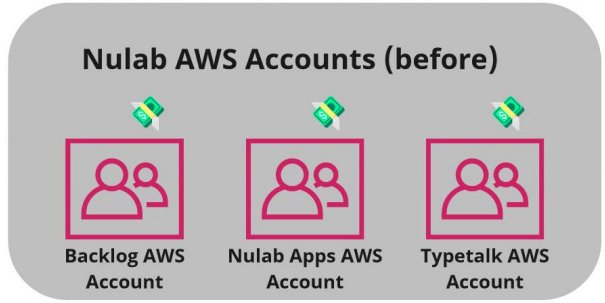
IT 全般統制: AWS アカウントの管理

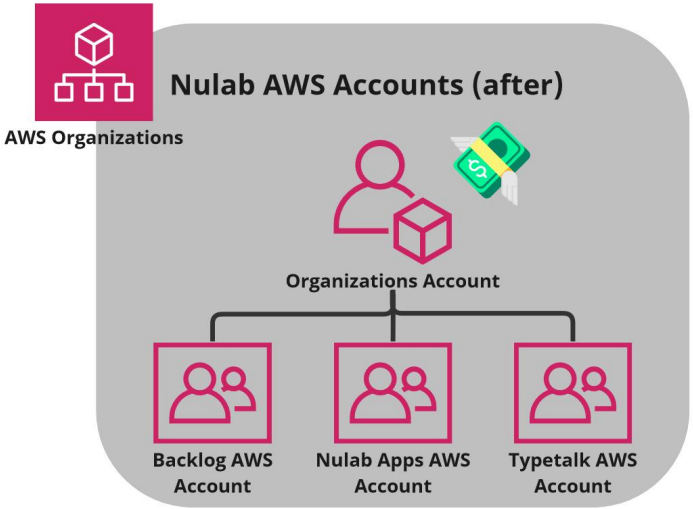
- 社内の AWS アカウントを統制する

  => AWS Organizations で各 AWS アカウントを統制する(認証は SAML)

  管理アカウントにサービスを持たせない

- AWS CloudTrail での監査ログの収集設定を AWS Control Tower で設定する
  - 各アカウントの権限で監査ログの収集を無効化できないようにする
  - ログ保存用のアカウントに集約し Organization アカウントの権限で削除できないようにする

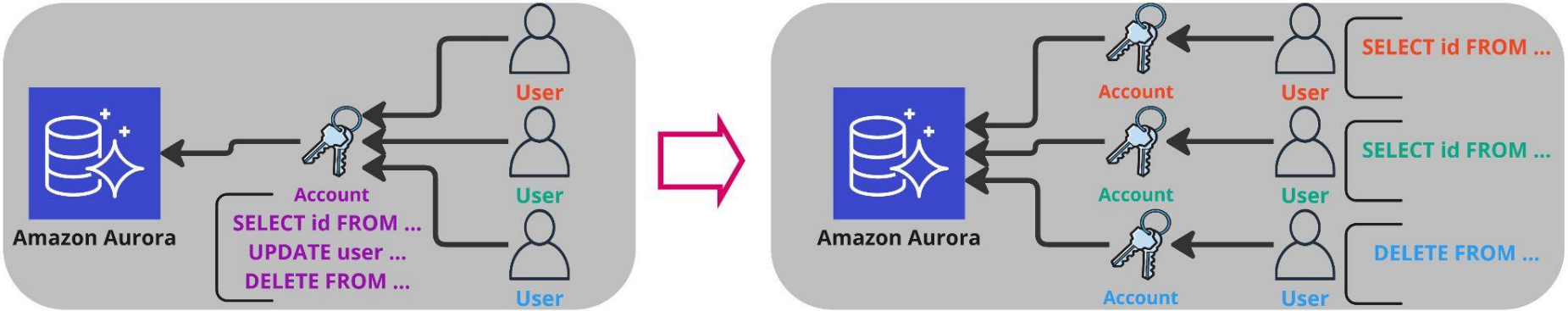




IT 全般統制: 顧客データベースの監査

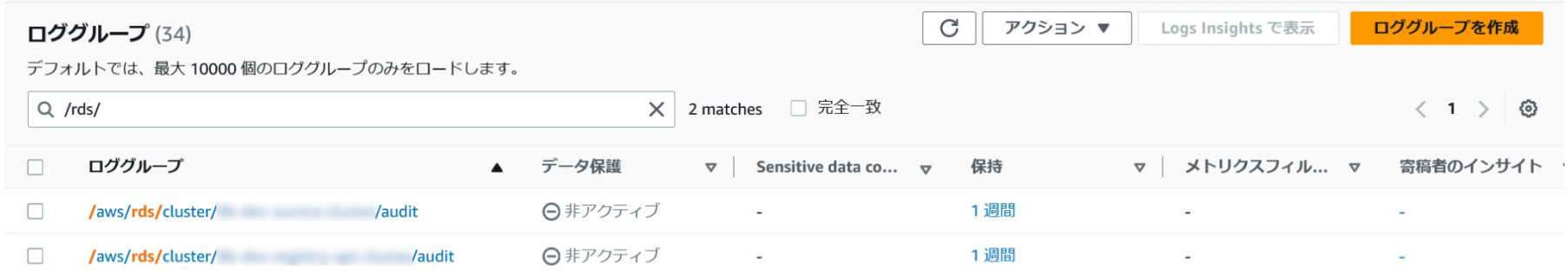
- DB アカウントの管理
  - アプリケーションごとにアカウントが分かれていない
    = 必要最低限の権限が割り当てられていない
  - 開発者個人に紐づいていない
    = 調査クエリやマイグレーションクエリを誰が発行したかわからない

=> 最低限のアカウント権限で管理する、監査ログを取得する



会計監査: 顧客データベースの監査

- DB に対するクエリのログを残す
- Amazon Aurora を使っている場合、AWS CloudWatch Logs に出力できる
  - 会計差分の追跡、会計監査としての保証
  - クエリの監査ログのモニタリング <- ログの取得の保証



リポジトリで管理しているクエリと照合

不審なクエリの実行がないかを月次で監査レポートを監査法人に提出

=> 面倒なので早い段階で仕組みを作っておく方がいい

会計監査: 会計・請求処理に関するソフトウェアの監査

- アプリケーションの改竄検知、防止
  - アプリケーションが変更されたときの会計データの
    意図しない変更を検知、防止するため
    - アプリケーションの変更追跡するため
  - Amazon EC2 の運用ではコマンド実行ログ等を
    AWS Systems Manager Session Manager, Auditbeat で取得

    => 誰が、いつ、アプリケーションをデプロイしたり、設定ファイルを
    変更したかを記録する

AWS サービス内の監査対応実例

会計監査: 会計・請求処理に関するソフトウェアの監査

- アプリケーションコンテナでの監査要件の実装
    - Amazon EC2 ではサーバの監査ログを取得しているため
      それに準拠する形で進める方針になりかけた

      => コンテナで監査ログを取得するのは大変 😓
- 現場で運用の仕様についてとりまとめて監査法人に相談
    - コンテナが動作しているホストマシンへのログイン禁止
    - コンテナへのログイン、ファイルシステムへの書き込み禁止
    - デプロイはコントロールプレーンで監査ログを取得

=> 高いセキュリティを確保することで監査ログの取得要件が発生しなくなる!

AWS サービス内の監査対応実例

会計監査: 会計・請求処理に関するソフトウェアの監査

- 監査法人と相談してわかったこと、成果
  - 監査法人は技術の専門家ではない
  - 他社事例を参考に楽な方法も提示してもらえる
  - 監査ログを取得する必要がないことを認めてもらえれば
    監査ログの取得を実装しなくていい(※重要)
    合祀的コントロールができていれば問題ない
  - 今後運用を変更するのであれば随時相談が必須

=> エンジニアとして監査仕様を現場で決められることは良かった!

まとめ

- なぜ IT 全般統制、監査が必要か？(上場 ≠ ゴール)
  - 誰が、いつ、どこで、どのように、何をしたかを記録する
- ヌーラボでの対応は基本的にトップダウンでの対応
  - 監査対応の事例をチームで集約して展開する
- 監査対応では現場に裁量を与える
  - Backlog でタスク管理する(対応事例、他チームの働きが見える)
  - 既存の運用への影響を最小限にして監査対応を行う
  - 組織内で判断できない場合は監査法人と話す機会をつくる
  - 監査対応、監査ログの保存アーキテクチャをAWS エンタープライズサポートに相談する