AWS マルチアカウント管理の実践

はじめに

このホワイトペーパーは、Amazon Web Services(AWS)のマルチアカウント管理に関するベストプラクティスをまとめるため、AWS パートナーネットワークセキュリティコンソーシアム・ジャパン(ASC-J)により作成されました。AWS 公式の情報やベストプラクティスも加味しつつ、様々な経験を持つ ASC-J 参加企業のメンバーにより、実際の構築における複雑さや AWS 以外の選択肢も含むスコープの広い解決策などを含む、より実践的なホワイトペーパーです。

本書の構成

本書では AWS マルチアカウント管理の情報を、以下の観点で網羅的にまとめています。

- これから AWS を利用する方も、すでに多数の AWS 環境を利用している方も利用できるよう幅広いスコープに対応
- AWS 利用の初期の段階から AWS マルチアカウント管理に関する知識をつけ、読者が置かれた状況に合わせて、管理知識を活用できる
- マルチアカウント管理に取り組んでいるすべての企業・組織でも活用できる
- 様々なステージの読者が、様々な規模の企業・組織にて活用できる

本書の流れは以下のように順序立てています。

- 1 章: AWS 利用の理想形とマルチアカウント管理の必要性
  - AWS を利用する上で企業・組織が直面する課題やマルチアカウント管理の必要性を整理
- 2 章: AWS マルチアカウント管理の具体要素と基本技術
  - 基本的なマルチアカウント管理の要素と技術を押さえる
- 3 章: AWS マルチアカウント管理のアレンジと運用
  - より応用的な内容として具体的な設定例やチューニングの勘所などを紹介

AWS マルチアカウント管理もクラウドを利用した 1 つの仕組みです。他のワークロードと同じように、ビジネスに合わせて柔軟に変更し、スケールし、継続的な運用をしていく必要があります。

対象読者

本書の対象読者は企業・組織の AWS アカウントの管理者や CCoE(Cloud Center of Excellence)担当者・セキュリティ担当者を始め、間接的に関わる調達部門や監査部門も含みます。あるいは SIer や CIer など、企業・組織を支援する立場の方も含みます。ビジネスの規模や AWS 利用のステージに制限はありません、どのようなケースでも参考になります。AWS マルチアカウント管理は、一人では実現できません。ビジネスおよびシステムに関わるすべてのロールの関係者を巻き込んで取り組みましょう。

注意

本書は情報提供のみを目的としています。本書のいかなる内容も、ASC-J、AWS、その他関係者からの保証、表明、契約的責任、条件や確約を意味するものではありません。

1 章 AWS 利用の理想形とマルチアカウント管理の必要性

近年、各企業・組織でのクラウド活用が成熟し、クラウド企業がベストプラクティスを提供するようになっています。クラウドを利用するに当たり、このベストプラクティスに則らない理由はありません。しかしベストプラクティスを理解したとしても、それを企業・組織に適用するには様々な課題が出てきます。本章では AWS を利用した場合の課題とマルチアカウント管理の必要性を説明します。

1.1.クラウド利用の理想形

クラウドは非常に魅力的なツールです。インフラストラクチャを迅速に構築でき、需要に対して柔軟に対応が可能であり、マネージドサービスの活用で構築や運用の手間を削減しビジネス側にリソースを集中できます。また、高いスケーラビリティとコストパフォーマンスを両立し、開発速度を維持したまま高いセキュリティを実現できます。AWS を利用する企業・組織はクラウドのメリットを最大限享受しながらビジネスを促進していきます。その中でほとんどの企業・組織はビジネスにおける AWS の利用範囲が広がり、AWS アカウントの数も増え、管理コストが発生します。企業・組織は AWS の特性を理解し、提供される AWS の各種サービスを理解し、自社の状況や性質を理解し、今と将来を見据えた AWS アカウ

ントの管理策を検討してく必要があります。

少し歴史の話をします。AWS のアカウント管理機能が充実していない頃、AWS をよく活用していた企業は 1 つの AWS アカウントに沢山の事業・沢山のサービス・沢山のユーザーが集まっていました。結果としてアクセス制御は複雑になり、コスト分離は難航し、誤操作誤設定による障害対応に追われました。今では 1 つの AWS アカウントに様々なワークロードを構築することはアンチパターンとなりました。現在の AWS アカウント分割方法は、事業・サービスどころか 1 つのサービスで利用する開発・検証・本番といった環境レベルでの分割がベストプラクティスです。

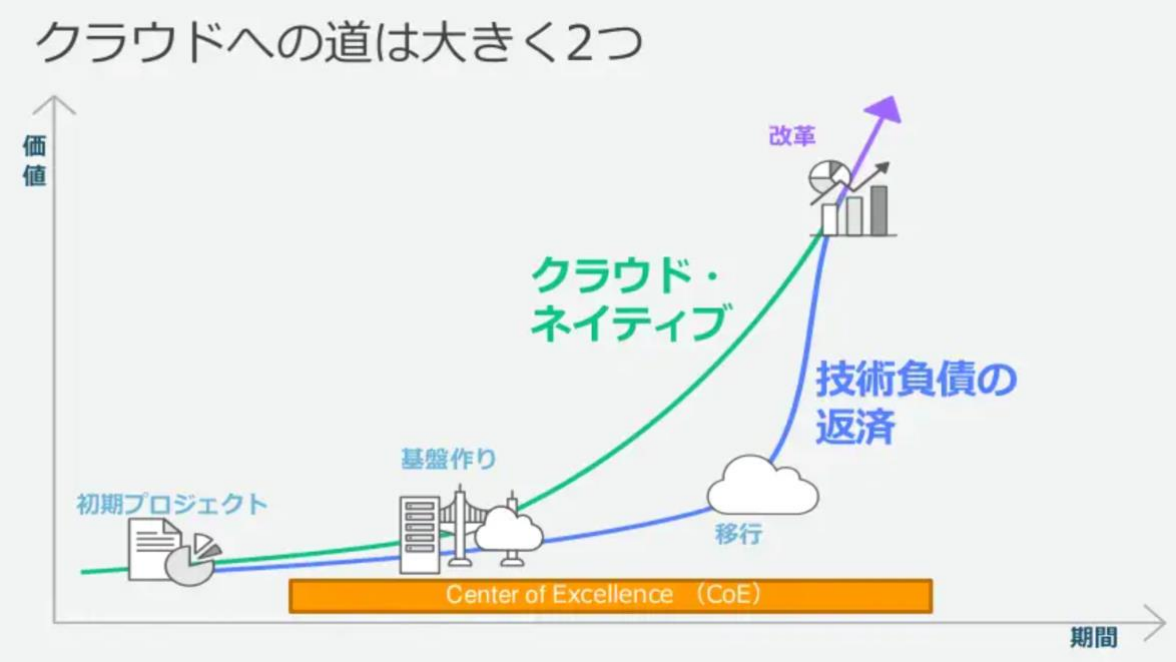
一方で AWS アカウントを細かく分割しただけでは、問題は解決しません。AWS マルチアカウント管理の機能は AWS 利用における上位の概念であり、より幅広い知識が必要になり、影響範囲を理解して計画的に設計・実装する必要があります。これが AWS マルチアカウント管理の難易度を上げる要因の 1 つでもあります。例えば、AWS アカウントの利用をこれから本格的に拡大したい企業・組織においては、AWS マルチアカウント管理の手法はこれから習熟していく対象であるにも関わらず計画的に取り組まなければいけないと考えると、難易度の高さが伺えます。

1.2.AWS 活用のステージ

クラウド活用の道のり、つまりクラウドジャーニーは千差万別です。クラウド活用の目的や利用規模、あるいはクラウドネイティブに始めているか・オンプレミスから大規模な移行を伴うかなど様々な背景やステージがあります。

AWS マルチアカウント管理に取り組むため、自身がどのような AWS 活用のステ

ージ(段階)にいるのかを理解します。企業・組織における AWS 活用の道のりは大きく 2 つに分けられます。



20180417 AWS White Belt Online Seminar クラウドジャーニより引用

1 つ目はクラウドの特性を最大限に活用するクラウドネイティブの道のりです。オンプレミスにおける既存システムを持たないスタートアップ企業や、新規システム案件で小規模で AWS を利用する場合は、クラウドネイティブの考え方で AWS を活用することとなります。

2 つ目は企業・組織の既存システムをクラウドへマイグレーションする道のりです。オンプレミスにおける既存システムを持つ多くの企業・組織が通る大規模移行などのたどる道のりです。こちらではハード老朽化対応や固定費から変動費への移行によるコスト柔軟性の確保・コスト削減などを目的としてクラウドへリフト&シフト(ある程度その状態のまま移行する)することを最初の目標とします。リ

フト&シフト完了後にクラウド環境での最適化を図ります。それぞれの道のりにおける各ステージでクラウド利用の理想形を実現するためにどのようなギャップが発生するかを見ていきましょう。

クラウドネイティブの道のりにおけるステージ

クラウドネイティブに AWS を活用し始めている場合は、マイグレーションの道のりと比べより様々なアプローチがあり十把一絡げにステージを定義することは難しいです。ここでは AWS アカウントの分割単位に焦点を当てて 3 つのステージを定義します。

- 1 Workload 1 Service
- 1 Workload Multi Service
- Multi Workload Multi Service

Workload(ワークロード)とは「ビジネス価値をもたらす一連のコンポーネント」と定義され、ビジネス単位で識別されます。1 つのサービスで構成される場合もあれば、マイクロサービスアーキテクチャのように複雑なサービスの組み合わせで構成される場合もあります。詳細は AWS Well-Architected Tool とはを参照してください。

1 Workload 1 Service のステージでは、AWS アカウントはごく少数です。したがってマルチアカウント管理の仕組みを利用することによる効果は限定的です。しかしながら、1 つのシステムでも AWS アカウントを全く分割しないということはありません。リリースの安定化のために開発・検証・本番環境と複数の環境を用意するため、AWS アカウント自体を分割します。それと同時に CI/CD を整備しリリース速度を向上します。

1 Workload Multi Service のステージでは、複数のサービスが独立してリリースサイクルを持ち、API ベースでサービス間のやり取りをします。これに合わせ AWS アカウントが増加します。サービス毎に利用する AWS サービス・設定基準・セキュリティなど様々な違いが出てくるため、これに伴う課題が増加します。組織全体で統一した基準や管理機能を提供するためのマルチアカウント管理や、ナレッジの集約などの必要性が上がります。

Multi Workload Multi Service のステージでは、ワークロード自体が増え複数の独立したビジネスや組織が存在します。AWS アカウントはもちろんのこと、AWS を直接操作する利用者や AWS に間接的に関わる関係者も増加し、役割分担が進む場合もあります。利用している AWS アカウント全体へのガバナンス・セキュリティの適用、共通で利用する仕組みや情報の集約・効率化、請求の管理・分割や最適化などマルチアカウント管理の機能が必須となります。

マイグレーションの道のりにおけるステージ

マイグレーションステージにおける AWS の利用は 4 つのステージで利用が促進されていきます。詳細についてはクラウドジャーニーの現在 | AWS Summit 2019 をご確認ください。

- PROJECT：限られた人が多くの経験を積む
  - 特定の要件のために利用
  - 効率的な選択肢か検証
  - 社内で有識者が認知
- FOUNDATION:：基礎を固めながら小さな成功を積み重ねる。
  - 推進組織(CCoE)が立ち上がる

  - ガイドラインや基盤が整備
  - 徐々に本番稼働
  - 既存データセンターと接続
- MIGRATION：全社的に利用しビジネス効果を享受
  - 長期利用の準備
  - 推進組織が確立
  - 多くのシステムが移行
  - データセンターの縮小
- REINVENTION：クラウドを最大活用し、ビジネス効果を最大化
  - デフォルトの選択肢となる
  - “なぜクラウドなのか”から”なぜクラウドではないのか”に変わる
  - アーキテクチャ/運用/組織が最適化

「PROJECT」のステージでマルチアカウント管理が必ずしも必要とはなりません。ほとんどの場合 1 つ以上の個人またはチーム管理アカウントを利用し、AWS の実験的な評価を実施します。そのため、このステージでは明確な方針を持ってアカウントを管理する必要性はあまりありません。ただし、本書の内容を理解し、今後検討すべき要素を把握すれば、構築する AWS の共通基盤の構築がよりスムーズになります。具体的には以下の要素です。

- AWS アカウントの作成・管理
- 社内での AWS 標準ルール策定
- AWS リソースの作成・管理
- AWS 利用コスト/請求/予算管理

「FOUNDATION」のステージでは本格的に AWS の利用を推進するために、全社でクラウドの利用者が利用できる AWS の共通基盤を構築します。このステージでマルチアカウントの管理を意識することにより柔軟性のある AWS の共通基盤を構築できます。特にこのステージで必要になる要素は以下の通りとなります。

- クラウド利用者の負担軽減
- ガバナンス
- コスト管理/最適化
- マイグレーションの加速
- セキュリティ

「MIGRATION」のステージでは利用規模の拡大に伴い、初期段階で担当していた役割を 1 つの組織では対応しきれなくなります。そのため、AWS を利用する組織への教育と権限の委譲、AWS アカウント増加に伴うマルチアカウント管理を検討します。初期段階に加え以下の要素が必要となります。

- AWS アカウントのアクセス管理
- マルチアカウント管理のための仕組みづくり(AWS アカウントへの共通設定展開)
- オンプレミス環境とのネットワーク接続管理
- AWS 利用者へノウハウのフィードバック、技術支援/アドバイザリ(AWS アーキテクチャのレビューなど)
- セキュリティイベントの管理(イベント発生時の対応をどうするか、発生後の対応)
- AWS ログの監査

- AWS 人材育成、トレーニング管理
- AWS ナレッジのサービス化

「REINVENTION」のステージでは AWS の利用が習熟し、安定した運用ができている状態です。クラウドのメリットを最大限活用できるように、これまでのステージを踏まえた継続的な改善サイクルが適切に回っている状態でもあります。

1.3 AWS 利用におけるアカウント管理の課題とアンチパターン

AWS の利用が進んでステージが変わっていくと、AWS アカウントの増加に伴い様々な課題が出てくることが、2 つのステージの流れからも感じられるでしょう。取り組むべき課題を整理します。ここでは以下の 4 点を上げます。

- 利便性
- セキュリティ
- コスト管理
- ガバナンス

利便性

クラウドのメリットは俊敏性・柔軟性・マネージドサービスなどの利便性にあります。しかし、AWS の利用規模が拡大するにつれこれらの利便性に制約を掛けてしまいがちです。

よくある制約の原因は過度な利用制限を伴う中央集権型のコントロールです。利用規模拡大に伴い全体のセキュリティ管理や利用基準の統一のため、AWS アカウ

ントの払い出しや設計・構成、利用できる AWS サービスの種類や IAM 権限に厳しい制約をかけます。すべて中央でコントロールし何もできないようにしてしまうことをゲートキーパー型の管理と表現します。

ゲートキーパー型による中央集権は管理組織による対応工数が膨大になり、各利用者からのリクエストに対応できず利便性が失われます。

このアンチパターンに陥らないように、本当にやってはいけないことだけ仕組みで防ぎ、その中で自由に権限を与えるガードレール型の管理が必要です。AWS マルチアカウント管理の仕組みでこのガードレールを整備することで利便性を保ちつつ利用規模の拡大に耐えうることが可能です。

セキュリティ

AWS 上のセキュリティも利用規模拡大に伴い変わっていきます。小規模の場合は適切に確保できていたセキュリティも、利用者が増えることで弱い部分が出てきたり、AWS アカウントごとにばらつきが出てきます。これによりセキュリティ事故が目立ち始めます。IAM User のクレデンシャルを漏洩する事故が発生し莫大な損失につながることが少なくありません。

また、これを防ごうとして利便性の課題と同じようにゲートキーパー型の管理になりがちです。少しの変更や新しい機能の利用に都度制限をかけることは、適切なセキュリティではありません。

AWS 上では利便性とセキュリティは両立できます。ガードレール型のセキュリティを整備して、利用者が危ない設定をすることを AWS の仕組みで防げます。セキュリティの管理者は AWS アカウント全体にまたがったセキュリティの仕組みを展開したり、ログやイベントを集約して検知・対応につなげることが可能です。

また ID も集約することで余分な IAM を発行することなく、事故を減らしたり利便性を向上できます。

コスト管理

AWS アカウントが増えてくるとコストの管理にもコストがかかります。

アンチパターンとしては、利用者個別に AWS アカウントを契約していると、契約先がバラバラで請求をまとめられないことがあります。あるいは 1 つの AWS アカウントに複数の環境を混ぜることで、タグによる分類では限界を迎えることもあります。増えた AWS アカウントや、各アカウント個別の設定などを管理する人的コストも問題になります。

請求については AWS Organizations による金銭コスト管理機能の集約が必要です。あるいは 1 つの契約先に絞ってもいいでしょう。統制された AWS アカウントの払い出しフローが整備されることで解決できます。

AWS アカウントを AWS Organizations にまとめると、まとまったディスカウントの適用が可能です。Reserved Instance / Savings Plan を複数の AWS アカウントをまたいで適用できます。

AWS アカウントや共通した機能の人的管理コストも、一元的に AWS アカウントを集約することで削減できます。

ガバナンス

利便性・セキュリティ・コスト管理など様々な課題が出てきました。これらは分断された組織での個別の管理や、定められていない全体の方針に起因します。このような環境では適切に管理されていない AWS アカウントが生まれ、リテラシー

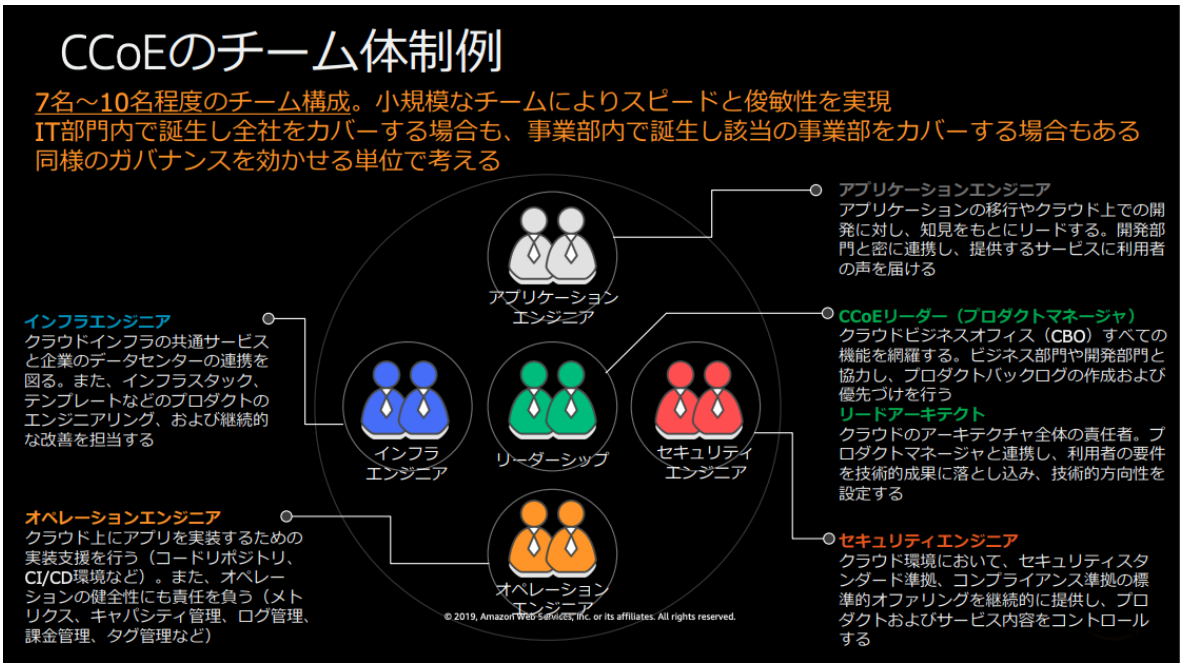
の低いまま運用され、結果多大なコストを支払うことに繋がります。

AWS マルチアカウント管理で統一した基準を簡単に全体適用していきます。組織的な管理を徹底し、人ではなく AWS の仕組みを使いガバナンスを効かせます。

[Tips] 仕組み以外の課題

前述の 4 点について、特に AWS マルチアカウント管理の仕組みで課題解決のアプローチを説明しました。しかしながら、これには当然仕組みだけではなく組織的な取り組みも必要になります。

クラウド活用において中心的に活動する組織を CCoE(Cloud Center of Excellence)といいます。CCoE の形や役割は組織により異なり、部門横断でバーチャルな組織もあれば、DX 推進チームが担う組織もあります。主な役割はクラウド全体の設計、利用方針・基準・ガイドラインの作成、各プロジェクトへの人的リソースやナレッジの提供、教育などがあります。クラウドを深く理解しているチームがあるからこそ、クラウドのメリットを活かせる全体管理の仕組みが確立できます。



クラウド推進組織を起点としたクラウド導入の進め方 | AWS Inovate 資料

CCoE が担う役割の一例は以下の記事も参考になります。

CCoE がやることについてまとめてみた | DevelopersIO

1 章まとめ

この章では AWS 利用の理想形から、AWS マルチアカウント管理の必要性を説明しました。自身のクラウドの道のりやステージを意識しつつ、アンチパターンに陥らないよう必要なステージで取り組むべき課題を意識してください。

2 章: AWS マルチアカウント管理の具体要素と基本技術

この章では AWS マルチアカウント管理に利用する手法やその採用基準や検討順序を解説します。AWS マルチアカウント管理の戦略は一言でいうと「どのような Landing Zone を作るか」です。まずは Landing Zone という言葉の解説から始め、AWS マルチアカウント管理の基本を押さえます。

2.1. Landing Zone とは

AWS マルチアカウント管理は、様々な手法やソリューションを組み合わせて実現します。セキュリティとコンプライアンスのベストプラクティスに基づいて構築される AWS マルチアカウント管理の環境を Landing Zone と呼びます。Landing Zone は 1 つの決まったパターンではなく、AWS マルチアカウント管理に必要な要素をまとめた概念です。実際に実装される Landing Zone は様々であり、千差万別です。組織は、後述する Landing Zone に必要な要素を検討し、自分たちに最適な Landing Zone を模索する必要があります。

そして、継続的に Landing Zone をアップデートしていく必要があります。AWS の利用全般もそうですし、構築する Landing Zone でも同じことが言えますが、作ったら終わりではありません。状況に合わせて柔軟に構成変更できることもクラウドのメリットです。Landing Zone も日々変わっていくことが求められますし、変わっても良いと思いながら作ることが大切です。スケーラビリティは確保しつつ、自分たちに最適な Landing Zone を模索しましょう。

[Tips] AWS Landing Zone ソリューション

AWS から提供されている Landing Zone に関する情報に AWS Landing Zone ソリューションというものが存在します。Landing Zone と AWS Landing Zone ソリューションは別の意味で利用されます。Landing Zone は概念であることに対し、

AWS Landing Zone ソリューションは 1 つの Landing Zone の実装方法です。その実装は CloudFormation で提供されています。なお、AWS Landing Zone ソリューションは現在長期サポート中であり、追加機能は提供されません。Landing Zone に関心のあるお客様は、次に紹介する AWS Control Tower をご検討ください。

2.2. Landing Zone に必要な要素

実際に Landing Zone に必要な要素は前章でも取り上げた以下があります。

- ID 管理: 各 AWS アカウントを利用するユーザーとアクセス権限の管理
- ガードレール: AWS 利用者が反する操作しないように検知・防止
- ベースライン展開: セキュアで統制された環境を自動的に払い出すための仕組み
- ログ管理・監視: 必要なログを集約・可視化し健全な状態を保つ
- 請求管理: 全体のコストを適切に管理・最適化

AWS Control Tower の実装を参考に、これらの要素を実現する手段を確認します。

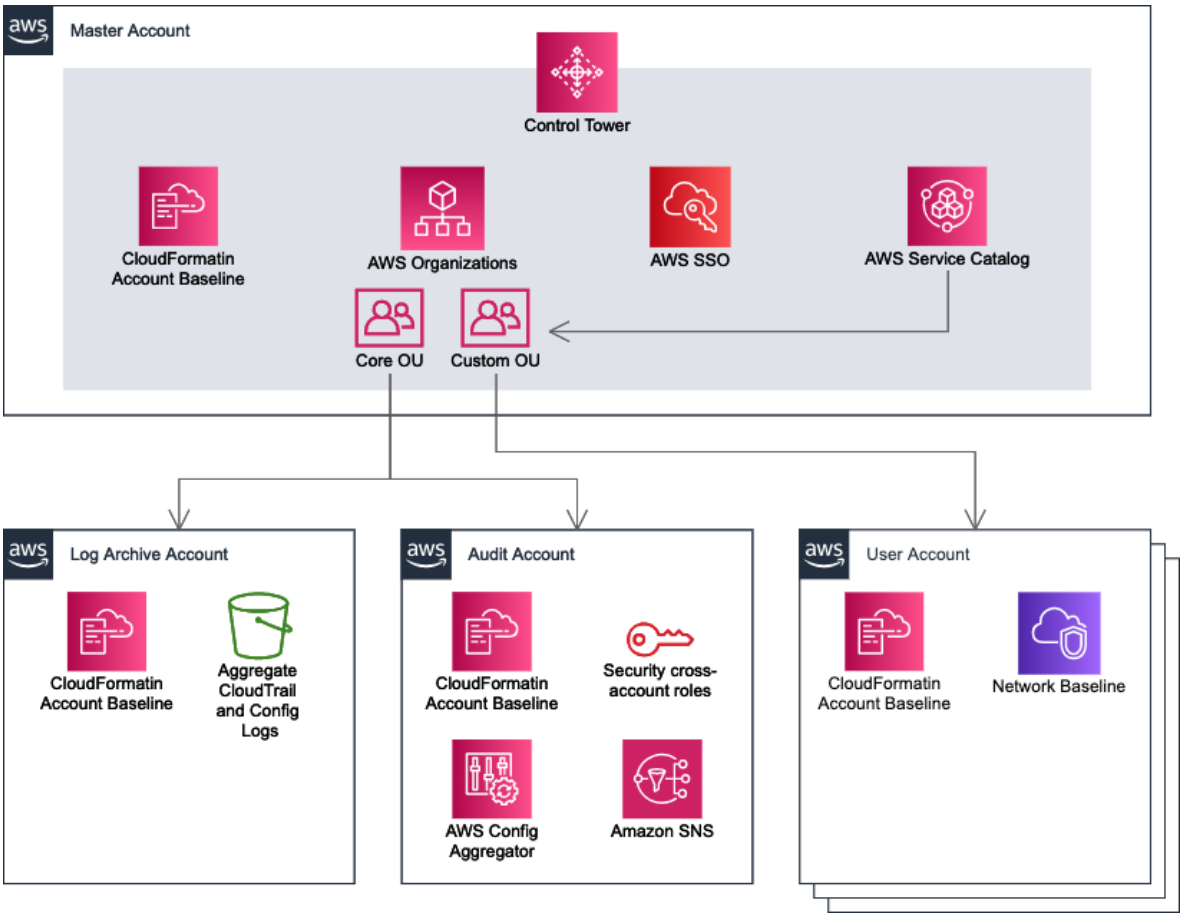
2.3. AWS Control Tower による Landing Zone の構築例

Landing Zone の基本は AWS Control Tower で学ぶ

AWS Control Tower は AWS が提供する Landing Zone 実装のマネージドサービスです。1 つの実装なので、これが最適となる組織もあれば、不十分あるいは冗長となる組織もあります。しかしながら、Landing Zone がどのように実装されるかを AWS Control Tower から学ぶことは価値があります。まずは AWS Control Tower による具体的な実装を紹介しますので、Landing Zone のイメージを深めてください。

AWS Control Tower による Landing Zone の構成内容

Landing Zone に必要な要素に対して、AWS Control Tower がどのような実装をしているか解説します。アーキテクチャとしては以下の図のようになります。



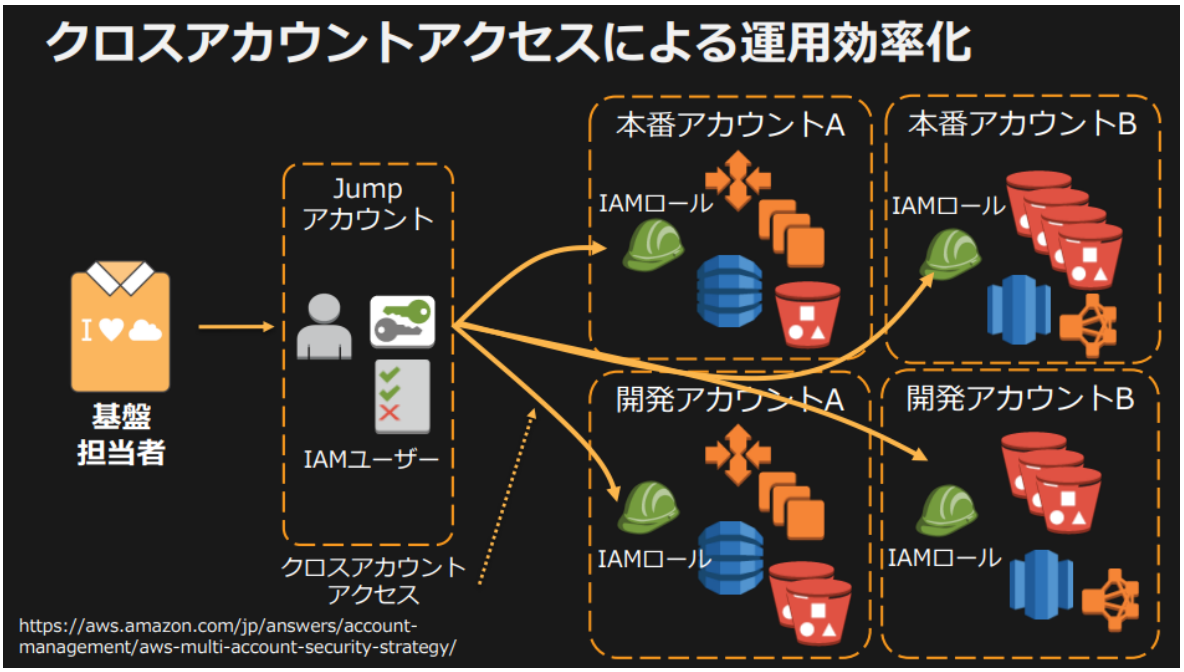
「AWS Control Tower を利用したマルチアカウント管理とセキュリティ統制」JAWS DAYS 2021 登壇資料 #jawsug #jawsdays #jawsdays2021 #jawsdays2021_C | DevelopersIO より引用

ID 管理

AWS Control Tower では各 AWS アカウントに対するユーザーとアクセス権限の管理に AWS Single Sign-On (AWS SSO)を利用します。AWS SSO は AWS Organizations と連携して、配下のアカウントに対するアクセスを管理します。どのアカウントにどのような権限でアクセスできるかという組み合わせを設定できます。全体のアクセス管理を AWS SSO のコンソール 1 箇所で集中管理できる

ため、管理が非常に楽になります。ユーザー情報は直接 AWS SSO に登録できる他、Active Directory や外部 IdP とも連携が可能です。

従来のマルチアカウントに対するアクセス管理の手法として、IAM User を 1 つのアカウントに集約して Switch する手法があります。いわゆる Jump アカウントという手法です。この方式は、ユーザーは 1 箇所に集まりますが、そのユーザーにどの権限を与えるかは Switch 先の IAM Role で管理されます。そのため、ユーザーの管理と権限の管理、つまり認証と認可が分散するため、管理が煩雑になる傾向がありました。AWS SSO はこの煩雑さを克服できる手法です。



AWS Summit 2017 AWS におけるマルチアカウント管理の手法とベストプラクティスより引用

ガードレール

AWS Control Tower では予防的ガードレールと発見的ガードレールという 2 つ

のガードレールが実装されています。予防的ガードレールは AWS Organizations の機能である SCP(Service Control Policy)で実装されます。発見的ガードレールは AWS Config の機能である Config Rules で実装されます。それぞれ以下のような特徴と役割があります。

- SCP(予防的ガードレール)
  - 特徴
    - AWS Organizations を使って OU/アカウント全体に強制的に適用するポリシー
  - 役割
    - 展開したガードレールやベースラインを削除できないようにする
- Config Rules(発見的ガードレール)
  - 特徴
    - ポリシーに違反する設定を検知して通知・追加で自動修復の作り込み可能
    - AWS Organizations が無くても利用可能
    - Aggregator によりマルチアカウントの集約可能
  - 役割
    - AWS Organizations で一括禁止できない細かい問題の検知
    - 設定値ベースのセキュリティ強化

通常 SCP や Config Rules の利用では手動での設定が必要ですが、AWS Control Tower では用意されたガードレールを利用できます。用意されたガードレールの種類は必須と任意があります。任意のガードレールはコンソール上のボタン操作

で有効化/無効化が可能です。組織は自分たちのポリシーに合わせて簡単にカスタマイズ可能です。一方で、用意されていないガードレールを設定する場合には、通常と同じく手動での設定が必要となり、2 重管理となります。

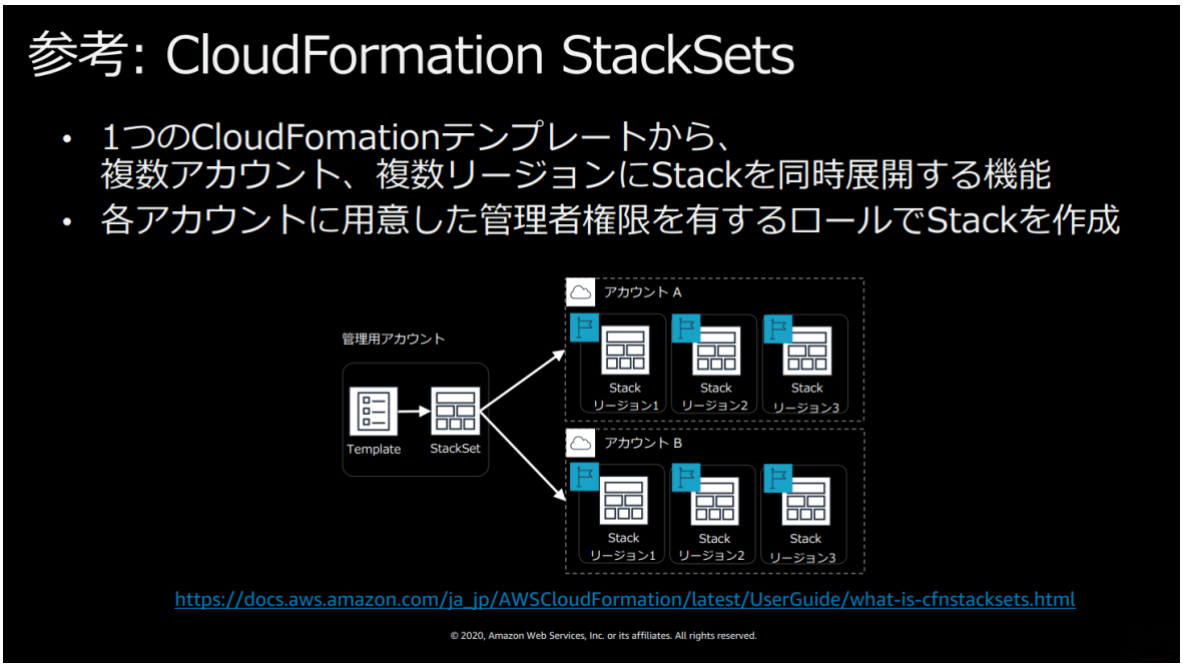
ベースライン展開

AWS Control Tower では、以下の設定がベースラインとしてすべてのアカウントに展開されます。

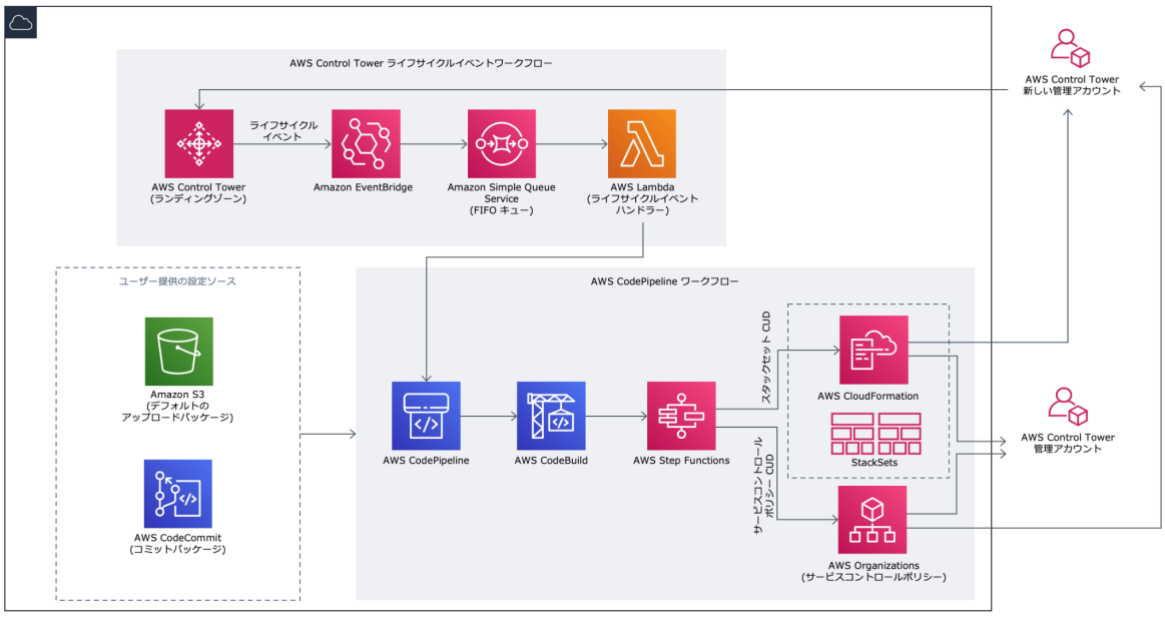
- AWS CloudTrail
- AWS Config

また、任意で VPC 及びそのサブネット構成を指定して展開が可能です。VPC のベースライン展開機能は、すべてのアカウントで共通の設定を展開する場合には有効ですが、違う設定を展開したい場合には別の仕組みが必要です。

AWS Control Tower による現状のベースライン展開は機能が少ないため、CloudFormation StackSets やカスタマイズソリューションなどの活用を検討しましょう。StackSets は AWS Organizations と連携することにより、自動的に Stack の展開が可能です。



AWS INNOVATE 2020 増加するシステムをマルチアカウントで効率よく管理するより引用



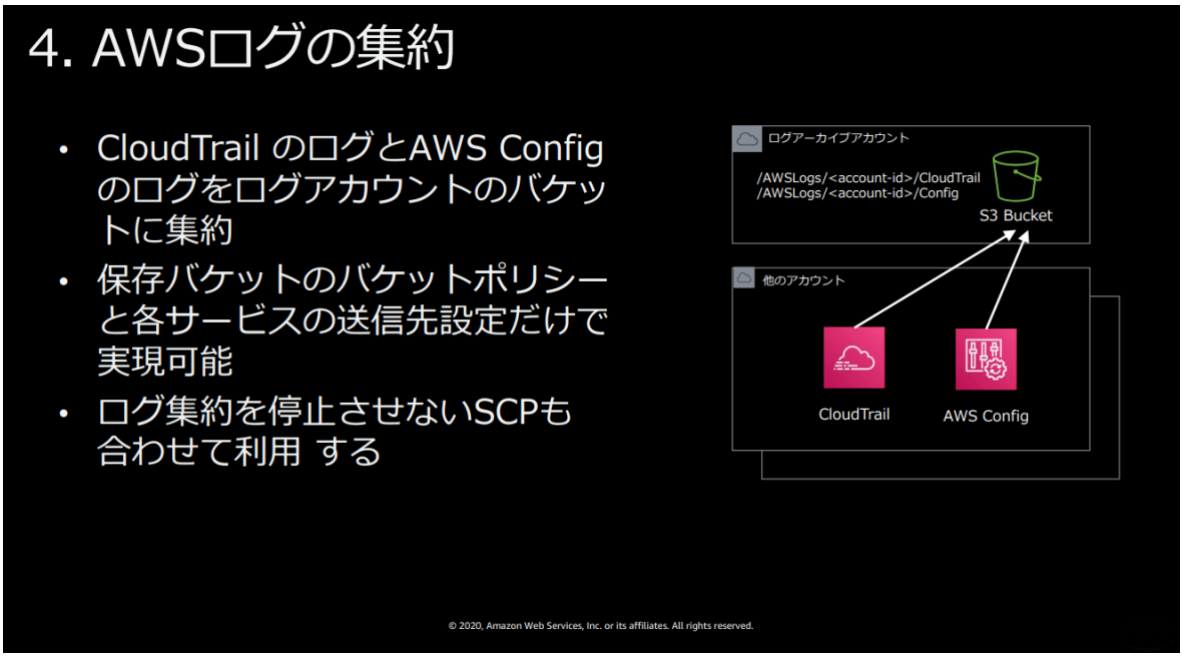
Control Tower カスタマイズソリューション(CfCT)を使ってガードレールと CloudFormation を自動展開してみた | DevelopersIO

ログ管理

AWS Control Tower では、管理用の AWS アカウントとして Log Archive アカウントが作成されます。このアカウントに AWS CloudTrail と AWS Config のログが集約されます。このアカウントで SIEM ソリューションを利用して、ログを分析・可視化できます。AWS ネイティブで活用できる OSS の SIEM のソリューションでは SIEM on Amazon OpenSearch Service があります。これにより AWS CloudTrail のログを可視化し、より詳細なセキュリティリスクの可視化や対応が可能です。

AWS Control Tower のデフォルトで集約されるログ以外にも、実装したアプリケーションや AWS サービスのログも集約を検討しましょう。例えば以下のような AWS サービスのログです。

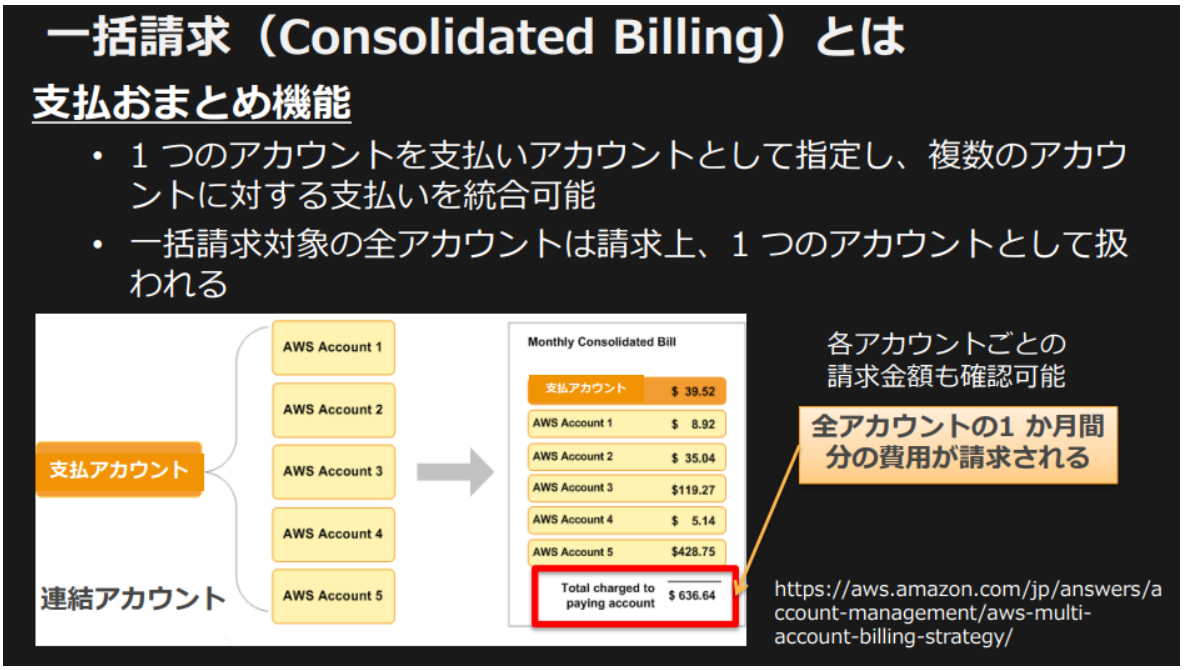
- Amazon CloudFront
- Elastic Load Balancer
- AWS WAF
- Amazon VPC Flow Logs



AWS INNOVATE 2020 増加するシステムをマルチアカウントで効率よく管理するより引用

請求管理

AWS Control Tower は AWS Organizations の機能により一括請求となります。Cost Explorer を利用した組織全体での可視性が提供されます。なお、AWS アカウントが AWS との直接的な契約でない場合は、各ベンダーの提供方法に従う必要があります。



AWS INNOVATE 2020 増加するシステムをマルチアカウントで効率よく管理するより引用

2 章まとめ

この章では Landing Zone の考え方を AWS Control Tower をベースに紹介しました。組織に最適な Landing Zone は千差万別であり、常に進化していくものです。既存の情報を参考に、トライアンドエラーを繰り返しながら実装していきましょう。各運用の仕組みや体制づくりは一朝一夕では実現できません。既存のナレッジを活用しながら進めましょう。

3 章: AWS マルチアカウント管理のアレンジと運用

Landing Zone は千差万別ですので、基本以外にも様々な考慮点や Tips、実装方法などがあります。これらを紹介します。

3.1. Landing Zone をどのようにアレンジするか

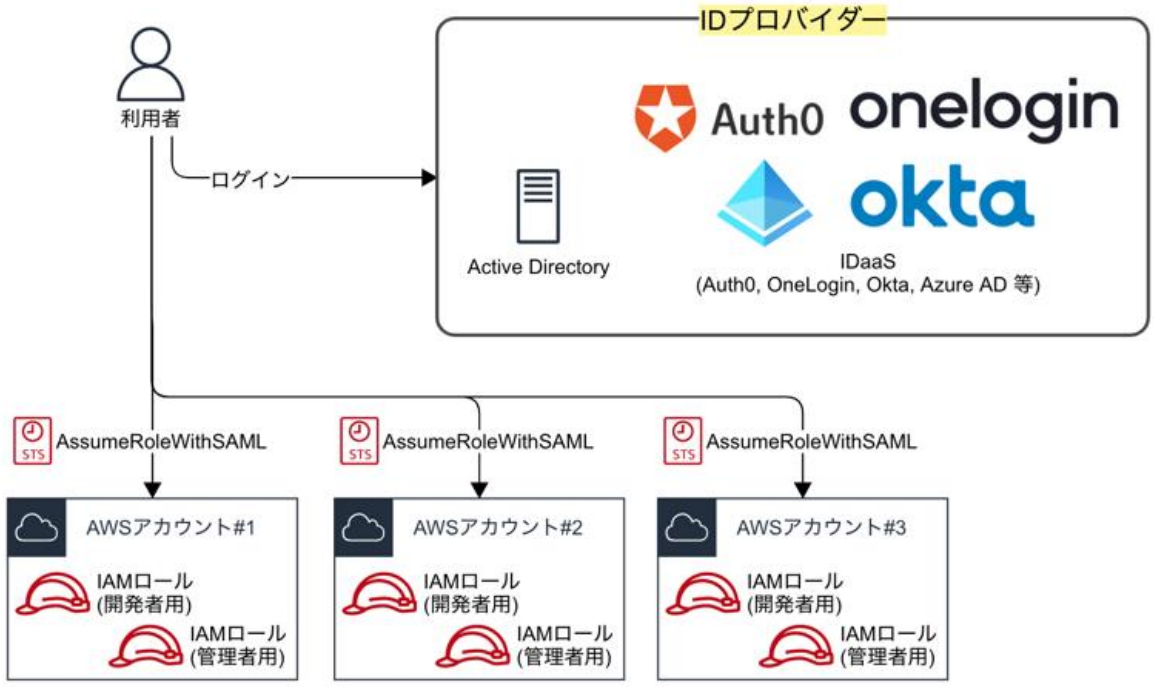
AWS Control Tower による Landing Zone の実装をベースに、Landing Zone で他にどのような実装ができるか、以下の内容を例示します。

- AWS Organizations を利用しない ID 管理
- AWS 以外も視野に入れた ID 管理
- 独自 Account Vending Machine の実装
- セキュリティベースライン
- 検証用の AWS Organizations・OU

AWS Organizations を利用しない ID 管理

AWS Organizations を利用できない場合でも ID 管理は分散させず、集約すべきです。ベーシックな実装方法には、前述の Jump アカウントを利用する手法があります。Jump アカウントにはすべての AWS アカウント利用者の IAM User を作成し、通常の IAM User は Switch Role の権限のみ保持します。

Jump アカウントの派生的な実装に、Jump アカウントのユーザーを IAM User ではなく SAML で連携する手法があります。例えば Azure AD を利用した「非 Organizations 環境の AWS アカウントに Azure AD のユーザーから SSO してスイッチロールしてみた」のブログが参考になります。もちろん各 AWS アカウントと直接 SAML 連携する方法もあります。



以下のリンクに様々な手法がまとまっています。

[マルチアカウントな AWS 環境のマネジメントコンソールへのアクセス方法をまとめてみた | DevelopersIO]

AWS 以外も視野に入れた ID 管理

AWS SSO の場合には、基本的に AWS 上での SSO を目的として利用されます。しかしながら、企業・組織で利用するサービスは AWS だけではありません。その他のサービスに対しても SSO の利用は積極的に実施すべきです。従って、AWS に依存しない IdP サービスを利用した ID 管理と SSO も検討すべきです。例えば Okta や OneLogin などのサービスがあります。

独自 Account Vending Machine の実装

Account Vending Machine とは AWS アカウントに対して様々な初期設定を実施する仕組みです。AWS Control Tower では AWS CloudTrail と AWS Config が展開されました。一般的には各アカウントに対して他にも様々な設定を追加したくなります。そのため、独自の Account Vending Machine の実装を検討します。

CloudFormation StackSets はその実装方法の 1 つです。StackSets は AWS Organizations との連携により、OU に対して自動で設定の展開が可能です。ただし、実行順序が決められないため依存関係には気をつける必要があります。

AWS Organizations が無くても StackSets の利用は可能です。この場合、最初に StackSets が利用する IAM Role の作成が必要になります。

一方で、CloudFormation に頼らないスクリプトを作成して都度実行することも検討します。CloudFormation はなにもない環境へ共通設定を展開するには向いていますが、既存の設定がある場合コンフリクトしやすいです。コンフリクトを回避するためには、冪等性を意識した Terraform や柔軟に組める独自のスクリプトのほうがよい場合もあります。

セキュリティベースライン

AWS では便利なセキュリティサービスがたくさんあります。以下もベースラインとして展開すると良いでしょう。

- Amazon GuardDuty
- AWS Security Hub
- IAM Access Analyzer

- Amazon Detective

検証用の AWS Organizations･OU

AWS Organizations を利用する場合には、検証用 OU を用意します。これは主に SCP の検証のためです。SCP は非常に強制力の強い機能であるため、本番環境に即時適用すべきではありません。検証用 OU を作成し、しっかり検証してから本番環境に適用しましょう。

特に AWS Control Tower を利用する場合には、OU だけでなく検証用の AWS Organizations も検討します。AWS Control Tower にはバージョンの概念があります。そして、新しいバージョンが出たらバージョンアップを実施します。この作業は現状不可逆であるため、取り返しがつきません。このためバージョンアップ検証用の AWS Organizations を検討しましょう。

AWS アカウント間や VPC 間のネットワーク設計

各システム共通で利用する基幹システムやシステム間の通信を確立するために AWS アカウントや VPC をネットワーク接続する場合があります。複数の接続方式があるため、その方法や用途を紹介します。

VPC Peering による *VPC* 間接続

VPC Peering は VPC 同士を 1 対 1 で接続します。VPC Peering を採用する場合、基本的には個別の、少数の VPC を接続する目的です。

VPC 間の接続は、接続されている各 VPC で重複しない IP 範囲(CIDR)を使用する場合に実現できます。一度作成した VPC の CIDR 変更は難易度が高いため、VPC

間の接続する可能性があれば、各 VPC で重複しない CIDR ブロックを割り当てる運用が必要です。

VPC Peering の数に応じてルートテーブルや、Security Group の管理も増えます。そのため VPC Peering でのフルメッシュ接続(すべての VPC 間を接続するつなぎ方)は構成が複雑になり運用が困難です。以下の式で VPC Peering 数を算出できます。

n(n-1)/2
n=VPC の数
VPC5 個：5(5-1)/2 = 10
VPC10 個：10(10-1)/2 = 45

5 個の VPC をフルメッシュ接続する場合は 10 個の VPC Peering、10 個の VPC をフルメッシュ接続する場合は 45 個の VPC Peering が必要となります。

目安として、5 個以上の VPC 間をフルメッシュ接続したい場合は Transit Gateway を用いること検討しましょう。

Transit Gateway によるネットワーク接続

Transit Gateway は VPC だけでなく VPN や Direct Connect とのハブになる機能です。複雑な VPC Peering のフルメッシュ接続と比較して簡単に複数 AWS アカウント(複数 VPC)間の接続が可能です。

Transit Gateway は基本的に接続した VPC 全体と通信できるため、部分的に VPC 間の通信させたくない場合は以下の手法が必要です。

- Transit Gateway の分割

- ルートテーブルの分割
- Security Group や NACL で制限

VPC のどこに Transit Gateway を接続するかも重要です。EC2 インスタンス等と同じサブネットに接続すると、ルートテーブルが複雑になりやすいです。Transit Gateway アタッチメント専用サブネットを作成するとルートテーブルの管理がわかりやすくなるためベストプラクティスです。

Transit Gateway のコストは、通信料の他アベイラビリティーゾーン(AZ)毎に 1 時間当たりのエンドポイントコストが発生するため、利用前に適切に試算しましょう。

ヘアピンパターンによる折り返し接続

ルートの制御は AWS 内に閉じない方法もあります。オンプレミスとの Direct Connect を複数利用している場合は、オンプレミス側のルータで VPC 間をルーティングできます。ヘアピンパターンと呼びます。

現在では基本的にこのパターンではなく Transit Gateway を採用すべきです。しかしオンプレミス側ルーティングの延長線で管理できることがメリットになる場合もあります。採用する場合には Direct Connect の帯域に十分ゆとりが必要です。

AWS Solutions Architect ブログ: ヘアピン DX パターン～AWS クラウド編

PrivateLink によるサービス提供

VPC 間で双方向の通信ではなく、サービス提供のように片方向からのリクエストを行う場合には PrivateLink での接続も可能です。Private Link は SaaS の API

提供（Web アプリの公開）のようにアクセス元の VPC に Private Link のエンドポイントを提供し、サービスで利用する TCP ポート通信のみ提供できます。

条件が合えば VPC Peering の代わりに利用することもあります。VPC Peering 利用時は接続されている各 VPC で重複しない IP 範囲設定が必要なため、重複している場合の対処法として Private Link が利用できます。

Private Link のエンドポイントは提供側 VPC と同一の AZ 内にサービスプロバイダ(NLB)がある場合のみ作成可能なため、すべての AZ でエンドポイント ENI を作成しておく必要があります。

ネットワーク設計参考リンク

AWS ホワイトペーパー Amazon VPC-to-Amazon VPC connectivity options

AWS Summit Tokyo 2019 ネットワークデザインパターン Deep Dive

AWS Summit Tokyo 2019 ネットワークデザインパターン Deep Dive 動画

AWS Summit Online 2020 AWS-40 Transit Gateway,PrivateLink VPC アーキテクチャー deep dive

AWS Summit Online 2020 AWS-40 Transit Gateway,PrivateLink VPC アーキテクチャー deep dive 動画

3.2. 各ガバナンス･セキュリティ機能の運用例

AWS マルチアカウント管理に伴う作業や各機能の運用例を、以下の要素ごと紹介します。

- AWS 基本教育と移譲する権限

- アカウント払い出し
- セキュリティチェック
- IAM チェック
- コスト最適化
- インシデントレスポンス

AWS 基本教育と移譲する権限

AWS マルチアカウント管理を実施する大前提として、AWS アカウントを各組織・プロジェクト・利用者へ移譲して利用を促進します。AWS アカウントは扱いを誤ると様々な問題に繋がることから、すべての利用者へ AWS 基本教育を徹底しなければいけません。

IAM の扱い、S3 の性質と設定、Security Group など、基本的なネットワークセキュリティの基本教育プロセスを策定・運用します。この役割は通常 CCoE のような AWS に関する全体管理・最適化を行う部門が担当します。

AWS の利用を極端に制限すると、クラウドのメリットを損ないます。一方で何も管理せず、制限もかけず、無作為に AWS アカウントを渡すことも適切ではありません。移譲する権限は各組織・プロジェクト・利用者の AWS 習熟度により判断します。

AWS の習熟度が低い組織に対しては、EC2/RDS/VPC など基本的なインフラストラクチャー構成を払い出し、AWS レイヤーの操作権限をほとんど渡さないところから始める選択肢もあります。ただし、これはほとんど AWS のメリットを活用できないため、インフラストラクチャーの管理をせず、継続的な改善もあまりない前提の環境です。

ある程度習熟した組織では、組織単位で IAM や AWS アカウント全体の設計を適切に判断できる責任者を設け、その権限を移譲します。責任者の管理の上で AWS を利用し管理部門がゲートを設けないことで、ビジネスの速度を落とすことなく、逆に加速できます。管理部門はゲート型対策の替わりに、後続に紹介する各種ガードレール型の対策で移譲した範囲も監視・保護します。

AWS アカウントを預かる各組織・プロジェクト・利用者はその責任を明確にする必要があります。移譲する権限や AWS アカウント・あるいはその上のアプリケーションや運用など、責任の範囲を明確にします。このような組織づくりは Well-Architected フレームワーク/運用上の優秀性/組織が参考になります。

アカウント払い出し

AWS アカウントを払い出す仕組みは AWS マルチアカウント管理において初期の段階から設計され、継続的に改善する必要があります。

アカウント払い出し時に適用されるべき設定は、前述の Account Vending Machine 及びセキュリティベースラインで挙げた以下の項目です。

- AWS CloudTrail
- AWS Config
- Amazon GuardDuty
- AWS Security Hub
- IAM Access Analyzer
- Amazon Detective

それぞれ CloudFormation StackSets を利用した設定展開が手法の 1 つです。あ

るいは Terraform などのサードパーティツールを利用する方法もあります。AWS Control Tower を利用している場合には CloudTrail 及び Config はデフォルトで展開されます。

以下の設定を決めておく必要があります。

- どのリージョンに展開するか
- ログをどこに集めるか
- 詳細な設定をどうするか

AWS の利用が進むにつれこれらの設定は変わるため、展開手法は継続的に運用できるものを選定する必要があります。どの手法にも長短があるためそれを確認しつつ、実際に利用する現場で好ましいものを選定します。

GuardDuty や Security Hub は Organizations 連携することで一括有効化したりイベントを一元管理できますが、詳細な設定を直接行うことが出来ません。これは StackSets でも解消できないため、スクリプトなどを用意し払い出し時に設定を実施するといいでしょう。継続的な変更も考えると、スクリプトを実行するための IAM Role もアカウント払い出し時に作成を検討します。これは利用が拡大するほどあとから追加することが手間になります。ただし権限の行使は影響範囲が大きいため、検知の仕組みも合わせて検討します。

セキュリティチェック

AWS 環境のセキュリティチェックは Security Hub と Config を活用します。初期は Security Hub の AWS 基礎セキュリティのベストプラクティスの活用を推奨します。幅広い AWS サービスのセキュリティをチェックでき、継続的に AWS が

アップデートし、設定を簡単に展開できるため、楽に運用できます。

Security Hub は Organizations の有無に関わらずマルチアカウントで管理可能です。Organizations 全体での Security Hub の管理は、Organizations の管理アカウントで行う状態がデフォルトですが、別の AWS アカウントに委任可能です。セキュリティ周りの管理は Organizations の管理とは分けて管理することがベストプラクティスなので、適切なセキュリティ管理アカウントを用意して委任しましょう。AWS Control Tower 環境であれば、委任先は Audit アカウントとすることが多いです。チェック項目は必要に応じ無効化できます。無効化設定はアカウントレベルで管理されるため、全体を管理している Security Hub と個別アカウントの Security Hub で粒度を変えられます。全体管理では最低限の項目を、個別アカウントでは詳細な項目までチェックするといいでしょう。

Security Hub では用意された項目のチェックのみ可能で、個別のセキュリティ要件に合わせたチェックはできません。対応できないものには Config で独自のルールを作成して展開する必要があります。ルールの実装は AWS Lambda で行います。作成難易度は比較的高くなるため、サードパーティのツール利用も選択肢の 1 つです。例えば Dome9 はこのルールの作成が比較的簡単にできます。

IAM チェック

IAM の管理は様々なレベルで行います。1 人の利用者は IAM ベストプラクティスに則り最小権限を意識します。IAM User やアクセスキーの発行は最小限にします。

AWS アカウントの責任者は作成された IAM User・Role・ポリシーのレビューを徹底します。レビューに役立つ機能として IAM Access Analyzer があります。こ

れは特に該当 AWS アカウント外とやりとりする権限を洗い出しチェックできます。

全体管理でも IAM Access Analyzer は有効に利用できます。Organizations 連携等で情報を集約し、各 AWS アカウントの責任者とやり取りしながら運用します。Security Hub/Config の情報も併用します。特に危険な設定は自動的に修復される仕組みも検討します。

[Tips] リソースベースのポリシーチェック

AWS における権限制御は IAM 以外に一部のリソースに対してリソースベースのポリシーを付与できます。詳細は「IAM ロールとリソースベースのポリシーとの相違点」が参考になります。IAM Access Analyzer では IAM だけではなくこのリソースベースのポリシーもチェック対象としています。ただし、すべてのリソースがサポートされているわけではないので、サポート外のリソースに対するチェックを検討する場合には独自スクリプトの実装検討が必要となります。IAM Access Analyzer がサポートしているリソースはサポートされているリソースタイプとして公開されています。

コスト最適化

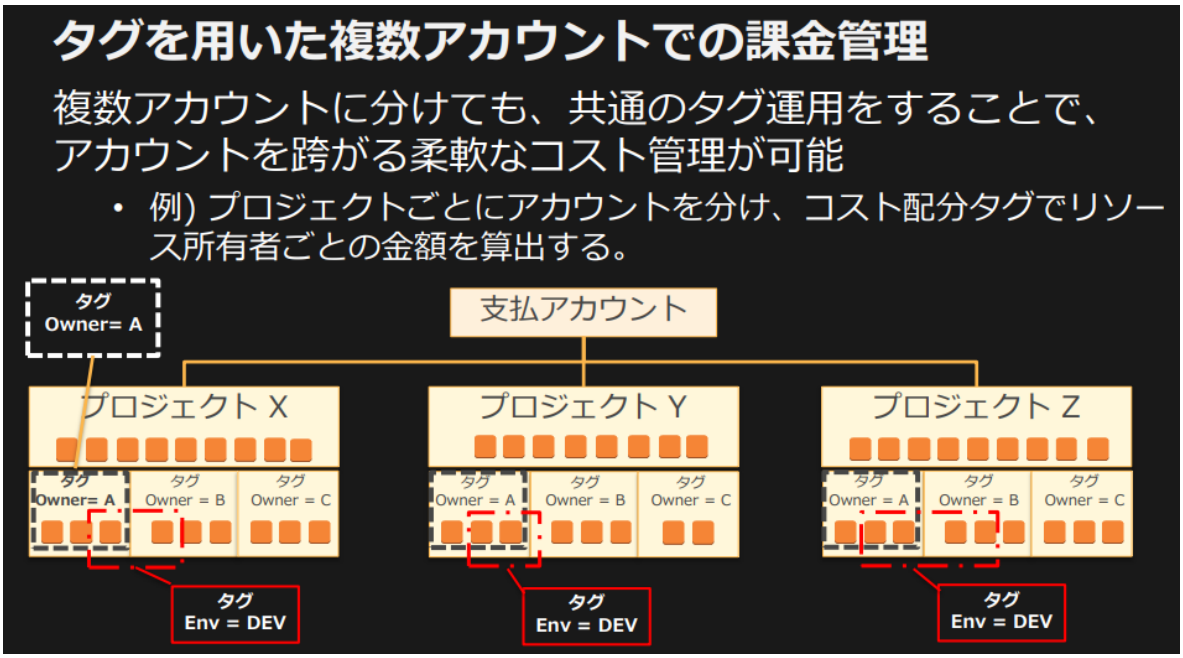
コスト最適化は AWS Well-Architected Framework の柱の 1 つであり、コストはセキュリティの守るべき対象でもあります。コスト最適化のため組織全体にまたがったコストを可視化するダッシュボードを作成します。AWS Cost Explorer や Amazon Athena/Amazon QuickSight を利用することで可視化が可能です。

Amazon EC2、Amazon EBS、AWS Lambda については AWS Compute

Optimizer でコストやパフォーマンス効率を確認できます。無償では過去 14 日間の Amazon CloudWatch メトリクスを分析してレコメンデーションをダッシュボードで可視化できます。[アップデート] AWS Compute Optimizer のダッシュボードにリソース効率メトリックが追加されました #reinvent | DevelopersIO を参考にしてください。

サードパーティのコスト管理ツールはより楽に利用できます。例えば nOps などがあります。AWS の未使用リソースを発見し削除、マルチアカウント時代のコスト削減 | DevelopersIO を参考にしてください。

コストの可視化にはコスト分配タグが有用です。しかしタグ管理は策定難易度も運用難易度が高く煩雑になるため、極力 AWS アカウントの分割でコストを分類します。過度なタグ管理は逆にコストとなります。



AWS Summit 2017 AWS におけるマルチアカウント管理の手法とベストプラクティスより引

用

[Tips] コスト割引オプションの活用

コスト割引オプションとして Reserved Instance、Savings Plans の 2 種類が用意されています。いずれのオプションも複数のアカウントでオプションの権利を共有する設定ができます。何をどの程度購入すべきか判断するためには、過去の利用実績を基に購入推奨を Cost Explorer で確認できるため活用しましょう。Reserved Instance、Savings Plans それぞれの特徴があるため、それぞれの特性を理解し用途に応じて購入すべきオプションを使い分けます。

組織における Savings Plans の活用に関しては「AWS Organizations と Savings Plans を活用したコスト削減のベストプラクティス」のブログか参考になります。

また、Savings Plans の基本的なことに関しては「Savings Plan に関するよくある質問」のブログが参考になります。

[Tips] タグ管理手段としてタグポリシーや Config ルールを活用

コスト配分のタグや社内の AWS 利用ルールに準拠することを目的に各リソースへ設定しているタグの管理が必要となることもあります。その場合、タグポリシーおよび Config のルールを利用することでタグの設定漏れや想定外のタグ値を設定する、といったことを予防できタグ管理の手段として有効です。「新機能 – タグポリシーを使用して、複数の AWS アカウントのタグを管理する」や「Config ルールを使って AWS リソースの特定タグ有無をチェックする」などが参考になります。

[Tips] コスト可視化に必要なレポート

AWS 標準のコスト確認サービスとして AWS Cost Explorer が存在します。AWS 標準以外に細かい独自の解析などを行いたい場合には、Cost Usage Report（CUR）を利用します。CUR を Amazon Athena/Amazon QuickSight やサードパーティー製のツールなどを用いて解析することでより細かい解析が可能です。なお、AWS アカウントが AWS との直接的な契約でない場合、CUR の提供については各ベンダーの提供方法に従う必要があります。

インシデントレスポンス

セキュリティ対策として未然にインシデントを防止することは重要です。しかしながらいざインシデントが起きてしまった場合の対策も同様に重要です。

GuardDuty は AWS 上の脅威検知に必須の機能です。この検知を素早く受け取り対応する必要があります。検知後は GuardDuty の各 Finding Types の内容と対応方法を解説した User Guide を参照し、インシデントの影響を調査し対応します。

GuardDuty も Security Hub と同じように Organizations 連携し、セキュリティ管理アカウントに委任しましょう。

大まかなインシデント対応の内容は Finding Types の種類により決まります。

EC2 Type は対象のインスタンスが不正な攻撃を受けている、あるいは不正プログラムに感染していることが疑われます。必要に応じ Security Group を活用した隔離や AMI スナップショットの取得による保全を行い調査します。

IAM Type は対象の IAM が漏洩し不正に利用されている可能性があります。正規

の利用か確認し、不正なものであれば速やかにアクセスキーの無効化と CloudTrail による実行履歴を元に影響範囲を確認します。

S3 Type は対象の S3 バケットやオブジェクトのデータが漏洩している可能性があります。取得していれば CloudTrail のデータイベントから詳細を確認します。

インシデントが発生したあとの検知・判断・対応・エスカレーション手順の確立と、日頃の運用が肝になります。普段からプレイブックの活用や一部フローの自動化も行います。

[Tips] Personal Health Dashboard の組織ビューによる組織規模のイベント収集

Organizations の機能である Personal Health Dashboard の組織ビューの活用により AWS 管理者が障害発生時に組織の AWS アカウント全体に対する影響を把握できます。詳細は「AWS Personal Health Dashboard で組織規模のイベント収集が可能に」および「[アップデート] AWS Personal Health Dashboard が Organizations と連携、組織ビューが見られるようになりました」のブログが参考になります。

3 章まとめ

この章では実際に Landing Zone を構築し運用していくために必要な考慮点や Tips、実装方法について紹介しました。実際に AWS マルチアカウント管理を始めていくと、外部の参考になる情報もあれば、独自に頑張っていかないと行けない課題も見えてきます。Landing Zone は千差万別ですので、継続的な改善をしながらより自分たちに最適な環境を目指しましょう。

おわりに

クラウドの管理方法に正解はありません。しかしながら、ベストプラクティスや様々な知見を組み合わせ、状況に合わせて柔軟に変えていくことで適切な維持管理が可能です。

現在の自身の置かれている状況、これから発生する課題を認識し、本書を活用してこれを解決してください。そして、新しい知見や経験を得たらどこかで発信してください。アウトプットの 1 つの成果が本書です。ぜひ同じようにアウトプットして、このエコシステムに参加してください。

リリース情報

- 2022/05/13 初版

執筆者

- クラスメソッド株式会社 臼田 佳祐
- クラスメソッド株式会社 梶 浩幸
- 株式会社 QES 矢口 元基
- ウルシステムズ株式会社　横山 芳成