



複数への移行 AWS アカウント

# AWS 規範ガイドンス



# AWS 規範ガイド: 複数への移行 AWS アカウント

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon の商標およびトレードドレスは Amazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していないその他のすべての商標は Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

# Table of Contents

序章 .....	1
対象者 .....	2
目的 .....	3
シングルアカウントアーキテクチャの例 .....	3
基本的なフレームワーク .....	5
AWS Well-Architected フレームワーク .....	5
での Cloud Foundation AWS .....	5
ID 管理とアクセス制御 .....	6
組織を設定する .....	6
ベストプラクティス .....	7
ランディングゾーンを作成する .....	8
ベストプラクティス .....	8
組織単位を追加する .....	9
ベストプラクティス .....	10
初期ユーザーを追加する .....	10
ベストプラクティス .....	11
メンバーアカウントを管理する .....	12
既存のアカウントを招待する .....	12
で VPC 設定をカスタマイズする AWS Control Tower .....	13
スコーピングの基準を定義する .....	14
アクセス許可とアクセスの管理 .....	16
エンジニアリング文化の考慮事項 .....	16
許可セットの作成 .....	17
請求に対する許可セット .....	17
開発者許可セット .....	18
本番稼働用許可セット .....	20
アクセス許可の境界の作成 .....	21
個人へのアクセス許可の管理 .....	24
ネットワーク接続 .....	26
VPC の接続 .....	26
アプリケーションの接続 .....	26
ベストプラクティス .....	27
エグレスの一元化 .....	27
送信トラフィックを保護するためのベストプラクティス .....	29

インGRESの分散化 .....	30
セキュリティインシデント対応 .....	34
Amazon GuardDuty .....	34
ベストプラクティス .....	35
Amazon Macie .....	35
ベストプラクティス .....	35
AWS Security Hub CSPM .....	36
ベストプラクティス .....	36
バックアップ .....	38
アカウントの移行 .....	39
リソース移行 .....	40
AWS AppConfig .....	41
AWS Certificate Manager .....	41
Amazon CloudFront .....	41
AWS CodeArtifact .....	42
Amazon DynamoDB .....	42
Amazon EBS .....	42
Amazon EC2 .....	42
Amazon ECR .....	43
Amazon EFS .....	43
Amazon ElastiCache (Redis OSS) .....	43
AWS Elastic Beanstalk .....	43
Elastic IP アドレス .....	43
AWS Lambda .....	44
Amazon Lightsail .....	44
Amazon Neptune .....	44
Amazon OpenSearch Service .....	44
Amazon RDS .....	45
Amazon Redshift .....	45
Amazon Route 53 .....	45
Amazon S3 .....	46
Amazon SageMaker AI .....	46
AWS WAF .....	46
請求に関する考慮事項 .....	47
結論 .....	48
寄稿者 .....	49

リソース .....	50
AWS 規範ガイド .....	50
AWS ブログ投稿 .....	50
AWS ホワイトペーパー .....	50
AWS コードサンプル .....	50
ドキュメント履歴 .....	51
用語集 .....	53
# .....	53
A .....	54
B .....	56
C .....	58
D .....	61
E .....	65
F .....	68
G .....	69
H .....	70
I .....	72
L .....	74
M .....	75
O .....	79
P .....	82
Q .....	85
R .....	85
S .....	88
T .....	92
U .....	93
V .....	94
W .....	94
Z .....	95
.....	xcvi

# 複数への移行 AWS アカウント

アマゾン ウェブ サービス ([寄稿者](#))

2024 年 11 月 ([ドキュメント履歴](#))

多くの企業は、Amazon Web Services (AWS) アカウントを 1 つ使用してジャーニーを始めます。企業内の複数のロールがこのアカウントを使用して事業を運営しています。エンジニアはコードを開発し、開発環境やテスト環境にデプロイし、本番環境へ変更を昇格させます。プロダクトマネージャーはデータソースをクエリして、業績に関する洞察を収集します。営業チームは、新しい顧客を誘導するために、本番環境からデモを実施しています。財務チームは、AWS Billing コンソールからクラウド支出をモニタリングしています。

これらの個別のロールがすべて 1 つを使用する場合 AWS アカウント、[最小特権のアクセス許可を適用する](#)というセキュリティのベストプラクティスを適用するのは難しい場合があります。つまり、ジョブの実行に必要な最小限のアクセス許可のみを付与します。スタートアップ企業が成長していく特定の段階で、誰かが「エンジニア全員が本番環境にアクセスする必要があるのか?」という質問をすることになります。答えはほとんどの場合「いいえ」ですが、多くの企業は、ビジネスを遅らせることなく、既存のシングルアカウント環境をマルチアカウント環境にする方法のことで悩んでいます。

このガイドには、シングルアカウント環境からマルチアカウント環境への移行に役立つベストプラクティスが載っています。アカウントの移行、ユーザー管理、ネットワーク、セキュリティ、アーキテクチャに関して必要な決定について説明します。ビジネスや日常業務のダウンタイムを最小限に抑えるか、まったく発生させないように設計されています。このガイドでは、単一アカウント環境からマルチアカウント環境に移行する際の以下の機能に焦点を当て AWS アカウント ています。

- [ID 管理とアクセス制御](#)
- [アクセス許可とアクセスの管理](#)
- [ネットワーク接続](#)
- [セキュリティインシデント対応](#)
- [バックアップ](#)
- [アカウントの移行](#)
- [リソース移行](#)
- [請求に関する考慮事項](#)

機能の詳細については、「[での Cloud Foundation AWS](#)」を参照してください。

このガイドは、「[Organizing Your AWS Environment Using Multiple Accounts](#)」ホワイトペーパー、「[AWS Security Reference Architecture \(AWS SRA\)](#)」、「[Establishing Your Cloud Foundation on AWS](#) ホワイトペーパー」など、このトピックに関連する既存のリソースに対応しています。このガイドに記載のない、より具体的なガイダンスについては、引き続きこれらのリソースを使用してください。

## 対象者

このガイドは、複数の AWS アカウントへの移行を望む、または移行する必要がある企業に最適です。スタートアップ企業の場合、このニーズは通常、製品市場に適合し、資金調達を行い、インフラストラクチャ、開発運用 (DevOps)、セキュリティなど異なるエンジニアリング分野で人を雇用し始めたときに発生します。

会社にこの移行を行う準備が整っていなくても、このガイドを参考にして、移行中に必要な決定について理解し、準備を始めることができます。

## マルチアカウントアーキテクチャへの移行の目的

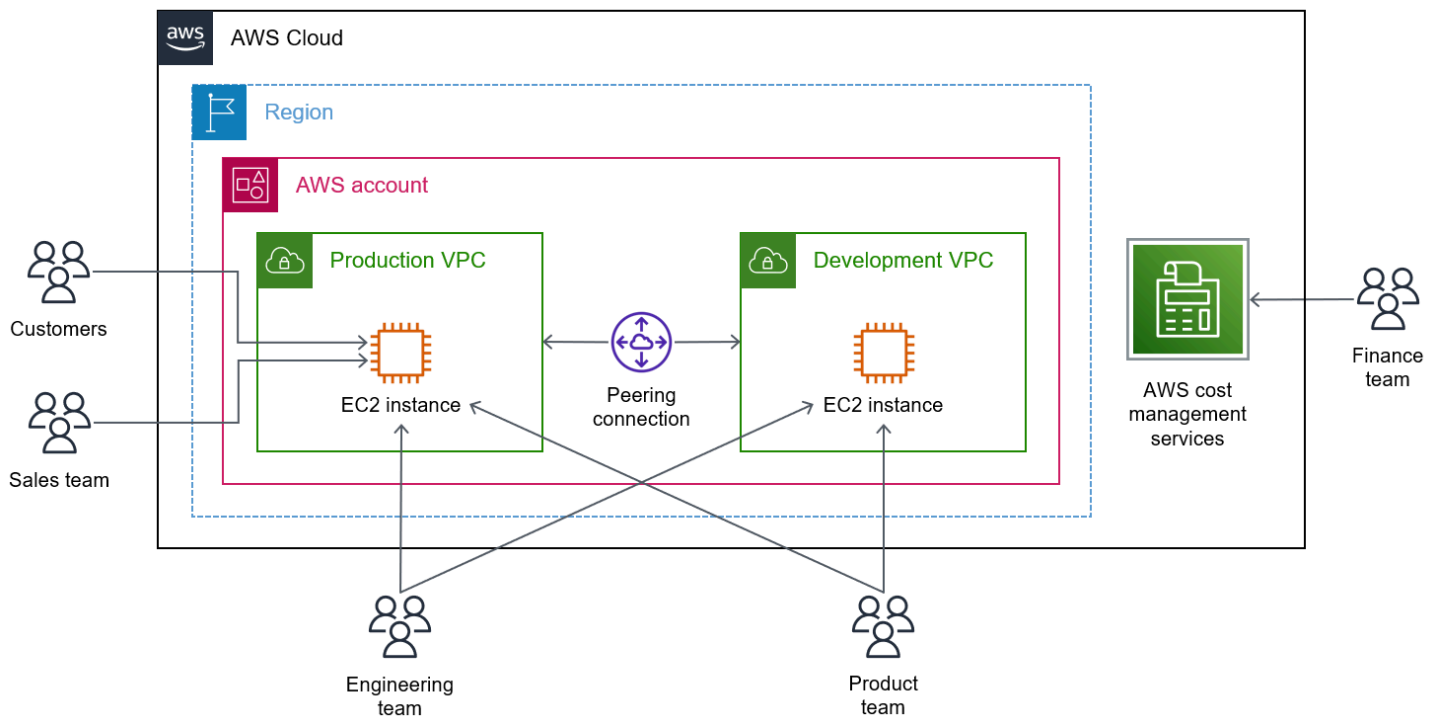
マルチアカウントアーキテクチャへの移行は、通常、以下のメリットの中から 1 つ以上のメリットを求めるビジネスニーズがあるで行なわれます。

- ビジネスの目的または所有権に基づいたワークロードのグループ化
- 環境ごとの個別セキュリティコントロールの適用
- 機密データへのアクセス制限
- イノベーションと俊敏性の促進
- 有害事象による影響範囲の制限
- 複数の IT 運用モデルのサポート
- のコスト管理
- クォータと API AWS のサービス リクエストレート制限の配布

マルチアカウントアーキテクチャを使用する多くの利点の詳細については、[「Organizing Your AWS Environment Using Multiple Accounts \(AWS ホワイトペーパー\)」](#) および [「ガイドライン」](#) を参照して、[適切に設計された環境をセットアップする](#) (AWS Control Tower ドキュメント) を参照してください。

## シングルアカウントアーキテクチャの例

スタート地点として、スタートアップ企業や小規模企業では 1 つの AWS リージョン を利用して、[VPC ピアリング](#) で接続した仮想プライベートクラウド (VPC) を 2 つ持つのが一般的です。各 VPC には、Amazon Elastic Compute Cloud (Amazon EC2) インスタンスなどのコンピューティングリソースが含まれています。エンジニアリングチームは、開発用 VPC で直接コードを開発します。製品チームが変更をレビューし、エンジニアリングチームが変更内容を本番稼働用 VPC に手動で昇格させます。財務チームは にアクセスできる AWS アカウント ため、AWS Billing and Cost Management コンソールを確認できます。



この環境で企業が経験する可能性のある問題には次のようなものがあります。

- エンジニアが、開発用データベースにアクセスしていると思い、誤って本番稼働用データを削除してしまっ
- 本稼働デプロイに予想以上に時間がかかったことで、販売デモが影響を受けた。
- 開発コードのロードテスト中に、本番稼働用 VPC の速度が低下し、スロットリングに関するエラーメッセージが生成された。
- 財務チームが本番環境と開発環境のコストを区別できない。
- CEO が、新たに雇用した外国の契約者の中に、本番稼働用 VPC から顧客データにアクセスする人がいることを懸念している。
- 財務チームは、高額 AWS のサービス なコストが発生する可能性のある特定の へのアクセスを許可することはできません。

マルチアカウント戦略を採用すると、ワークロードとアクセスを分割 AWS アカウント して使用することで、これらのすべての課題に対処できます。

# マルチアカウントアーキテクチャに移行するための基本的なフレームワークとセキュリティの責任

このガイドに記載されている情報とベストプラクティスは、インフラストラクチャとセキュリティに関する AWS の既存の推奨事項を補完するためのものです。単一 AWS アカウント から複数に移行するときは AWS アカウント、新しいマルチアカウントアーキテクチャが AWS Well-Architected フレームワークと Cloud Foundation の原則と一致していることを確認することが重要です。これにより、ガバナンスの要件と AWS ベストプラクティスに従いながら、セキュリティ、パフォーマンス、回復性を重視した環境を構築して運用できます。

## AWS Well-Architected フレームワーク

[AWS Well-Architected フレームワーク](#)は、アプリケーションとワークロードのための安全で高性能、耐障害性、効率的なインフラストラクチャを構築するのに役立ちます。このガイドは、このフレームワークの柱にある[運用上の優秀性](#)、[セキュリティ](#)、[信頼性](#)に沿ったものです。これにより、現在の AWS 推奨事項に従うことで、ビジネス要件と規制要件を満たすことができます。

AWS アカウントで [AWS Well-Architected Tool](#) を使用して、Well-Architected のベストプラクティスへの準拠を評価することができます。

## での Cloud Foundation AWS

[でのクラウド基盤の確立 AWS](#) (AWS ホワイトペーパー) は、ビジネスニーズに合わせて AWS 環境を調整するのに役立つガイダンスを提供します。機能ベースのアプローチを使用すると、ワークロードをデプロイ、運用、管理するための環境を作成できます。また、要件が変化し、クラウドにワークロードを追加でデプロイする場合に、機能を強化して環境を拡張することもできます。で定義される 30 の機能の詳細については AWS、[「の機能」](#)を参照してください。このガイドには、意図した順序で初期機能を実装するためのベストプラクティスが記載されています。

運用上およびガバナンスのニーズに応じて、機能を導入して実装することができます。ビジネス要件が成熟したら、機能ベースのアプローチを、クラウド環境がワークロードをサポートし、必要に応じてスケールする準備ができていることを検証するためのメカニズムとして使用することができます。このアプローチにより、ビルダーとビジネスのためのクラウド環境を自信を持って構築できます。

# マルチアカウントアーキテクチャへ移行するための ID 管理とアクセス制御

マルチアカウントアーキテクチャに移行する際に最初に行うのは、組織内で新しいアカウント構造を設定することです。その次に、ユーザーを追加し、アカウントへのアクセスを設定します。このセクションでは、複数の AWS アカウントへのアクセスを管理する方法について説明します。

このセクションでは以下のタスクを取り上げます。

- [組織を設定する](#)
- [ランディングゾーンを作成する](#)
- [組織単位を追加する](#)
- [初期ユーザーを追加する](#)
- [メンバーアカウントを管理する](#)

## 組織を設定する

複数の [AWS アカウント](#)、[組織](#)を通じてそれらのアカウントを論理的に管理できます [AWS Organizations](#)。のアカウント AWS Organizations は、AWS リソースと AWS アカウント、それらのリソースにアクセスできる ID を含む標準です。組織は、単一のユニットとして管理 AWS アカウント できるように [を統合するエンティティ](#)です。

アカウントを使用して組織を作成すると、そのアカウントが組織の管理アカウント (支払い者アカウントまたはルートアカウントとも言います) になります。組織が持つことのできる管理アカウントは 1 つだけです。AWS アカウント [組織](#)に [を](#)追加すると、メンバーアカウントになります。

### Note

各 [には](#)、ルートユーザーと呼ばれる単一の ID AWS アカウント もあります。アカウントの作成に使用したメールアドレスとパスワードを使用して、ルートユーザーとしてサインインできます。ただし、日常的なタスクには、それが管理者タスクであっても、ルートユーザーを使用しないことを強くお勧めします。詳細については、「[AWS アカウントのルートユーザー](#)」を参照してください。

また、[メンバーアカウントのルートアクセスを一元化](#)し、組織内のメンバーアカウントからルートユーザーの認証情報を削除することをお勧めします。

アカウントは、組織ルート、組織単位 (OU)、メンバーアカウントから成る階層ツリー構造に整理されます。ルートとは、組織のすべてのアカウントが設定された親コンテナのことです。組織単位 (OU) とは、[ルート](#)内にある[アカウント](#)のコンテナのことです。OU には他の OU やメンバーアカウントを含めることができます。OU は、親を 1 つだけ持つことができ、各アカウントは 1 つの OU にのみ属することができます。詳細については、「[用語と概念 \(AWS Organizations ドキュメント\)](#)」を参照してください。

[サービスコントロールポリシー \(SCP\)](#) は、ユーザーとロールが使用できるサービスとアクションを指定します。SCP は、アクセス許可を付与しない点を除いて、AWS Identity and Access Management (IAM) アクセス許可ポリシーに似ています。その代替りとして、SCP はアクセス許可の最大数を定義します。ポリシーを階層内のノードのどれかにアタッチすると、そのポリシーは、ノード内のすべての OU とアカウントに適用されます。例えば、ポリシーをルートに適用した場合、そのポリシーは組織内のすべての [OU](#) と [アカウント](#) に適用されます。またポリシーを OU に適用した場合、そのポリシーは、ターゲット OU の OU とアカウントにのみ適用されます。

[リソースコントロールポリシー \(RCP\)](#) は、組織内のリソースに対して使用可能なアクセス許可の最大数を一元的に制御します。RCP は、アカウント内のリソースが組織のアクセスコントロールガイドラインの範囲内に収まるようにするのに役立ちます。

AWS Organizations コンソールを使用して、組織内のすべてのアカウントを一元的に表示および管理できます。組織を利用する利点の 1 つは、管理アカウントとメンバーアカウントに関連するすべての料金が記載された一括請求書を受け取ることができることです。詳細については、「[一括請求 \(AWS Organizations ドキュメント\)](#)」を参照してください。

## ベストプラクティス

- 既存の を使用して組織 AWS アカウント を作成しないでください。新しいアカウントを作成します。これが組織の管理アカウントになります。特権オペレーションは組織の管理アカウント内で実行でき、SCP と RCP 管理アカウントには適用されません。そのため、管理アカウントに含まれるクラウドリソースとデータは、管理アカウントで管理する必要があるものだけに制限する必要があります。
- 管理アカウントへのアクセスを、新しい をプロビジョニング AWS アカウント し、組織を管理する必要がある個人だけに制限します。
- SCP を使用して、ルート、組織単位、メンバーアカウントにアクセス許可の最大数を定義します。SCP を管理アカウントに直接適用することはできません。
- RCP を使用して、メンバーアカウントのリソースの最大アクセス許可を定義します。RCP を管理アカウントに直接適用することはできません。
- (AWS Organizations ドキュメント) の [ベストプラクティス AWS Organizations](#) に従ってください。

## ランディングゾーンを作成する

ランディングゾーンは、ワークロードとアプリケーションをデプロイするための出発点となる、適切に設計されたマルチアカウント AWS 環境です。マルチアカウントアーキテクチャ、ID とアクセスの管理、ガバナンス、データセキュリティ、ネットワーク設計、ログ記録を開始するためのベースラインとなります。[AWS Control Tower](#) は、自動化されたガードレールを提供することで、マルチアカウント環境の保守とガバナンスを簡素化するサービスです。通常、アカウント AWS のサービス内の他のランディングゾーンをオーケストレーションすることで、all AWS リージョン、AWS Control Tower works 全体で環境を管理する単一の AWS Control Tower ランディングゾーンをプロビジョニングします。詳細については、「[ランディングゾーンをセットアップするとどうなるか](#) (AWS Control Tower ドキュメント)」を参照してください。

でランディングゾーンを設定するときは AWS Control Tower、管理アカウント、ログアーカイブアカウント、監査アカウントの 3 つの共有アカウントを識別します。詳細については、「[共有アカウントとは](#) (AWS Control Tower ドキュメント)」を参照してください。管理アカウントの場合、ワークロードをホストしていない既存のアカウントを使用してランディングゾーンをセットアップする必要があります。ログアーカイブアカウントと監査アカウントでは、既存のアカウントを再利用するか AWS アカウント、自分で作成 AWS Control Tower するかを選択できます。

AWS Control Tower ランディングゾーンを設定する方法については、「[開始](#)方法 (AWS Control Tower ドキュメント)」を参照してください。

### ベストプラクティス

- [マルチアカウント戦略の設計原則](#) (AWS ホワイトペーパー) のベストプラクティスに従ってください。
- [管理者向けの AWS Control Tower ベストプラクティス](#) (AWS Control Tower ドキュメント) に従ってください。
- ワークロードの大部分をホスト AWS リージョン する にランディングゾーンを作成します。

#### Important

ランディングゾーンをデプロイした後にこのリージョンを変更する場合は、 の支援が必要であり AWS サポート、ランディングゾーンを廃止する必要があります。この方法は推奨されません。

- 管理するリージョンを決定するときは AWS Control Tower、ワークロードをすぐにデプロイする 予定のリージョンのみを選択します。このリージョンは後で変更または追加することができます。

がリージョン AWS Control Tower を管理する場合、検出ガードレールはとしてそのリージョンにデプロイされます [AWS Config ルール](#)。

- 管理するリージョンを決定したら AWS Control Tower、管理されていないすべてのリージョンへのアクセスを拒否します。これにより、ワークロードと開発者は承認された AWS リージョンしか使用できないようになります。これは組織内のサービスコントロールポリシー (SCP) として実装されます。詳細については、[AWS リージョン「拒否コントロールの設定」](#) (AWS Control Tower ドキュメント) を参照してください。
- デランディングゾーンを設定するときは AWS Control Tower、次の OUs とアカウントの名前を変更することをお勧めします。
  - セキュリティ OU を Security\_Prod に変更することをお勧めします。この OU が本番稼働用セキュリティ関連 AWS アカウントに使用されることを示すためです。
  - AWS Control Tower 追加の OU を作成してから、サンドボックスからワークロードに名前を変更することを許可することをお勧めします。次のセクションでは、ワークロード OU 内部に追加の OU を作成し、これを AWS アカウントの整理に使用します。
  - 集中ロギングの名前を Log Archive AWS アカウント から log-archive-prod に変更することをお勧めします。
  - 監査アカウントの名前は、監査から security-tooling-prod に変更することをお勧めします。
- 不正を防ぐために、ラン AWS Control Tower デランディングゾーンに追加する前に使用履歴 AWS アカウントがある AWS が必要です。使用履歴 AWS アカウントのない新しいを使用している場合は、新しいアカウントで、AWS 無料利用枠にない Amazon Elastic Compute Cloud (Amazon EC2) インスタンスを起動できます。インスタンスを数分間実行してから終了します。

## 組織単位を追加する

マルチアカウント環境を設定するには、適切な組織構造を確立することが重要です。サービスコントロールポリシー (SCP) を使用して OU と OU 内のアカウントに対してアクセス許可の最大数を定義するため、管理、権限、財務報告の観点から組織構造は論理的でなければなりません。組織単位 (OUs [「用語と概念」](#) (AWS Organizations ドキュメント) ) を参照してください。

このセクションでは、本番環境と非本番環境などの環境のセグメント化と構造化に役立つネストされた OU を作成して、デランディングゾーンをカスタマイズします。これらの推奨ベストプラクティスは、デランディングゾーンを分割して本番環境と非本番環境のリソースを分離し、インフラストラクチャをワークロードから分離することを目的としています。

OUs [「組織単位の管理」](#) (AWS Organizations ドキュメント) を参照してください。

## ベストプラクティス

- [ランディングゾーンを作成する](#) に作成したワークロード OU 内に、次のネストされた OU を作成します。
  - Prod — この OU は、顧客データを含む本番稼働用データを保存し、そこにアクセスする AWS アカウント に使用します。
  - NonProd — この OU は、開発環境、ステージング環境、テスト環境などの本番稼働用以外のデータを保存する AWS アカウント に使用します。

組織ルートの下に Infrastructure\_Prod OU を作成します。この OU を使用して、一元化されたネットワークアカウントをホストします。

## 初期ユーザーを追加する

ユーザーに AWS アカウントアクセスを許可するには、2 つの方法があります。

- IAM ID (ユーザー、グループ、ロール)
- の使用などによる ID フェデレーション AWS IAM アイデンティティセンター

小規模な企業やシングルアカウント環境では、新しい人が入社したときに管理者が IAM ユーザーを作成するのが一般的です。IAM ユーザーに関連付けられたアクセスキーとシークレットキーの認証情報は、有効期限がないため、長期認証情報と呼ばれます。ただし、攻撃者がこれらの認証情報を侵害した場合、そのユーザー用に新しい認証情報を生成する必要があるため、セキュリティのベストプラクティスとして推奨されていません。にアクセスするもう 1 つの方法は、[IAM ロール](#)を使用すること AWS アカウント です。[AWS Security Token Service](#) (AWS STS) を使用して、設定した時間が経過すると有効期限が切れる短期認証情報を一時的にリクエストすることもできます。

[IAM Identity Center](#) AWS アカウント を使用して、へのユーザーアクセスを管理できます。従業員や契約社員ごとに個別のユーザーアカウントを作成したり、各自のパスワードや多要素認証 (MFA) ソリューションを管理したり、グループ化してアクセスを管理したりできます。MFA を設定するときは、認証アプリケーションなどのソフトウェアトークンを使用するか、YubiKey デバイスなどのハードウェアトークンを使用できます。

IAM アイデンティティセンターは、Okta、JumpCloud、Ping ID などの外部 ID プロバイダー (IdP) からのフェデレーションもサポートしています。詳細については、「[Supported identity providers](#)」(IAM アイデンティティセンターのドキュメント) を参照してください。外部 IdP とフェデレーショ

ンすることで、アプリケーション全体のユーザー認証を管理し、IAM アイデンティティセンターを使用して特定の AWS アカウントへのアクセスを認可できます。

## ベストプラクティス

- ユーザーアクセスの設定は、「[セキュリティのベストプラクティス](#)」(IAM ドキュメント) を順守します。
- アカウントアクセスを個々のユーザーではなくグループごとに管理します。IAM アイデンティティセンターに、各ビジネス機能を代表する新しいグループを作成します。例えば、エンジニアリング、財務、営業、製品管理のグループなどです。
- 多くの場合、すべての AWS アカウント にアクセス (多くは読み取り専用アクセス) する必要があるユーザーと、1つの AWS アカウントへのアクセス権を必要とするユーザーを分けることで、グループを定義します。グループに関連付けられた AWS アカウント および アクセス許可を簡単に識別できるように、グループには次の命名規則を使用することをお勧めします。

<prefix>-<account name>-<permission set>

- 例えば、AWS-A-dev-nonprod-DeveloperAccess グループの場合、AWS-A が 1 つのアカウントへのアクセスを示すプレフィックスです。dev-nonprod がアカウント名で、DeveloperAccess がグループに割り当てられた許可セットです。AWS-0-BillingAccess グループの場合、AWS-0 が組織全体へのアクセスを示すプレフィックスで、BillingAccess がグループの許可セットを示しています。この例では、グループは組織全体にアクセスできるため、グループ名にはアカウント名が含まれていません。
- 外部の SAML ベースの IdP で IAM アイデンティティセンターを使用していて、MFA が必要な場合は、属性ベースのアクセス制御 (ABAC) を使用して、認証方法を IdP から IAM アイデンティティセンターに渡すことができます。属性は SAML アサーションを通じて送信されます。詳細については、「[Enable and configure attributes for access control](#)」(IAM アイデンティティセンターのドキュメント) を参照してください。

Microsoft Azure Active Directory や Okta などの多くの IdP は、SAML アサーション内で認証方法リファレンス (amr) クレームを使用して、ユーザーの MFA ステータスを IAM アイデンティティセンターに渡すことができます。MFA ステータスのアサーションに使用されるクレームとその形式は IdP によって異なります。詳細については、IdP のドキュメントを参照してください。

IAM Identity Center では、AWS リソースにアクセスできるユーザーを決定するアクセス許可セットポリシーを作成できます。ABAC を有効にして属性を指定すると、IAM アイデンティティセンターは認証されたユーザーの属性値を IAM に渡し、ポリシー評価で使用できるようにします。詳細については、「[Create permission policies for ABAC](#)」(IAM アイデンティティセンターのドキュ

メント) を参照してください。次の例に示すように、aws:PrincipalTag 条件キーを使用して MFA のアクセス制御ルールを作成します。

```
"Condition": {
  "StringLike": { "aws:PrincipalTag/amr": "mfa" }
}
```

## メンバーアカウントを管理する

このセクションでは、既存のアカウントを組織に招待し、組織内に新しいアカウントを作成します。このプロセスで重要な点は、新しいアカウントをプロビジョニングする必要があるかどうかの判断基準を定義することです。

このセクションでは以下のタスクを取り上げます。

- [既存のアカウントを招待する](#)
- [で VPC 設定をカスタマイズする AWS Control Tower](#)
- [スコーピングの基準を定義する](#)

### 既存のアカウントを招待する

では AWS Organizations、会社の既存のアカウントを新しい組織に招待できます。他のアカウントを招待できるのは、組織内の管理アカウントだけです。招待されたアカウントを管理者が承認すると、そのアカウントはすぐに組織に加わり、組織の管理アカウントが、新しいメンバーアカウントで発生するすべての料金を負担することになります。詳細については、「[組織への AWS アカウントの招待](#)」および「[組織からの招待の承諾または拒否](#)」(AWS Organizations ドキュメント) を参照してください。

#### Note

アカウントを組織に招待できるのは、そのアカウントが現在別の組織に所属していない場合のみです。アカウントが既存の組織のメンバーである場合は、アカウントをその組織から削除する必要があります。アカウントが誤って作成された別の組織の管理アカウントである場合は、その組織を削除する必要があります。

**⚠ Important**

既存のアカウントからコストまたは使用状況の履歴情報にアクセスする必要がある場合は、AWS Cost and Usage Report を使用してその情報を Amazon Simple Storage Service (Amazon S3) バケットにエクスポートできます。組織への加入を承認する前に、これを行ってください。アカウントが組織に追加されると、このアカウントの履歴データにはアクセスできなくなります。詳細については、「[Setting up an Amazon S3 bucket for Cost and Usage Reports](#)」(AWS Cost and Usage Report ドキュメント) を参照してください。

## ベストプラクティス

- 本番ワークロードを含んでいる可能性が高い既存のアカウントを、[組織単位を追加する](#) で作成したワークロード > Prod の組織単位に追加することをお勧めします。
- デフォルトでは、組織の管理アカウントには、組織に招待されたメンバーアカウントに対する管理アクセス権がありません。管理アカウントに管理制御をさせる場合は、メンバーアカウント内に OrganizationAccountAccessRole の IAM ロールを作成して、そのロールを引き受けるアクセス許可を管理アカウントに付与する必要があります。詳細については、「[招待されたメンバーアカウントでの OrganizationAccountAccessRole の作成](#) (AWS Organizations ドキュメント)」を参照してください。
- 組織に招待した既存のアカウントについては、[メンバーアカウントのベストプラクティス](#) (AWS Organizations ドキュメント) を確認し、アカウントがこれらの推奨事項に従っていることを確認します。

## で VPC 設定をカスタマイズする AWS Control Tower

Account [Factory](#) AWS アカウント を使用して新しい をプロビジョニングすることをお勧めします AWS Control Tower。Account Factory を使用すると、Amazon EventBridge との統合を使用して AWS Control Tower、アカウントが作成され AWS アカウント するとすぐに新しい にリソースをプロビジョニングできます。

新しい を設定すると AWS アカウント、[デフォルトの Virtual Private Cloud \(VPC\)](#) が自動的にプロビジョニングされます。ただし、Account Factory から新しいアカウントをセットアップすると、AWS Control Tower で追加の VPC が自動的にプロビジョニングされます。詳細については、「[AWS Control Tower と VPC の概要 VPCs](#) (AWS Control Tower ドキュメント)」を参照してください。つまり、デフォルトでは AWS Control Tower が新しいアカウントごとに 2 つのデフォルト VPC をプロビジョニングします。

企業がアカウント内の VPC をより細かく制御しようとするのはよくあることです。多くの場合は、AWS CloudFormation Hashicorp Terraform や Pulumi などの他のサービスを使用して VPCs。AWS Control Towerによりプロビジョニングされる追加の VPC が作成されないように、Account Factory の設定をカスタマイズする必要があります。手順については、「[Amazon VPC 設定の構成 \(AWS Control Tower ドキュメント\)](#)」を参照して、次の設定を適用します。

1. インターネットアクセス可能なサブネットのオプションを無効にします。
2. プライベートサブネットの最大数は、0 を選択します。
3. VPC 作成のリージョンでは、すべてのリージョンをクリアします。
4. アベイラビリティゾーンは、3 を選択します。

## ベストプラクティス

- 新しいアカウントごとに自動的にプロビジョニングされるデフォルト VPC を削除します。これにより、ユーザーは専用の VPC を明示的に作成しなければ、アカウントでパブリック EC2 インスタンスを起動できなくなります。詳細については、「[デフォルトサブネットとデフォルト VPC の削除](#)」(Amazon Virtual Private Cloud ドキュメント)を参照してください。[AWS Control Tower Account Factory for Terraform](#) (AFT) を設定して、新しく作成されたアカウントのデフォルト VPC を自動的に削除することもできます。
- dev-nonprod AWS アカウント という新しい をワークロード > NonProd 組織単位にプロビジョニングします。このアカウントは開発環境に使用してください。手順については、「(AWS Control Tower ドキュメント) [を使用して Account Factory アカウントをプロビジョニング AWS Service Catalog](#)する」を参照してください。

## スコーピングの基準を定義する

新しい をプロビジョニングするかどうかを決定する際に、会社が使用する基準を選択する必要があります AWS アカウント。ビジネスユニットごとにアカウントをプロビジョニングすることも、本番、テスト、QA などの環境に基づいてアカウントをプロビジョニングすることもできます。各企業には、それぞれの規模について独自の要件 AWS アカウント があります。通常、アカウントの規模を決定する際には、次の 3 つの要素を評価します。

- サービスクォータのバランス — サービスクォータは、AWS のサービス 内の各 のリソース、アクション、および項目の最大値です AWS アカウント。多数のワークロードが同じアカウントを共有していて、1 つのワークロードがサービスクォータのほとんどまたはすべてを消費している場合、同じアカウントの別のワークロードに悪影響を及ぼす可能性があります。その場合は、そのような

ワークロードを他のアカウントに分ける必要があるかもしれません。詳細については、「[AWS のサービスクォータ](#)」(AWS 全般のリファレンス) を参照してください。

- コストレポート - ワークロードを個別のアカウントに分離することで、コストと使用状況のレポートでコストをアカウントレベルで確認できます。同じアカウントを複数のワークロードに使用する場合、タグをリソースの管理と識別に使用することができます。タグ付けの詳細については、「[AWS リソースのタグ付け \(\)](#)」を参照してくださいAWS 全般のリファレンス。
- アクセス制御 - ワークロードがアカウントを共有する場合、ユーザーが必要のないワークロードにアクセスできないように、アカウントリソースへのアクセスを制限する IAM ポリシーをどのように設定するかを検討する必要があります。別の方法としては、複数のアカウントや IAM アイデンティティセンターで[許可セット](#)を使用して、個々のアカウントへのアクセスを管理することができます。

## ベストプラクティス

- [AWS AWS Control Tower ランディングゾーンのマルチアカウント戦略](#)のベストプラクティスに従ってください (AWS Control Tower ドキュメント)。
- AWS リソースの識別と管理に役立つ効果的なタグ付け戦略を確立します。タグを使用し、リソースを目的、事業ユニット、環境などの基準別に分類できます。詳細については、「[タグ付けのベストプラクティス](#) (AWS 全般のリファレンス ドキュメント)」を参照してください。
- ワークロードが多すぎるアカウントに負荷をかけすぎないようにします。ワークロードの需要がサービスクォータを超えると、パフォーマンスの問題が発生する可能性があります。競合するワークロードを別のワークロードに分割 AWS アカウント することも、サービスクォータの引き上げをリクエストすることもできます。クォータ引き上げのリクエストの詳細については、Service Quotas ドキュメントの「[Requesting a quota increase](#)」を参照してください。

# マルチアカウントアーキテクチャのアクセス許可とアクセスの管理

このセクションでは以下のトピックを取り上げます。

- [エンジニアリング文化の考慮事項](#)
- [許可セットの作成](#)
- [アクセス許可の境界の作成](#)
- [個人へのアクセス許可の管理](#)

## エンジニアリング文化の考慮事項

AWS Well-Architected フレームワークの柱の 1 つは運用上の優秀性です。チームは[運用モデル](#)と、ビジネス成果を達成する上での各自の役割を理解している必要があります。各自の責任を理解して、所有権を持つことができ、意思決定の方法を知っていれば、チームは共通の目標の達成に集中できます。

成長が早い初期段階の企業では、チームの全員が複数の役割を果たします。ユーザーが AWS アカウント全体に対して非常に特権的なアクセス権を持っていることは珍しくありません。企業が成長するにつれて、最小特権の原則に従い、ユーザーが仕事を遂行するのに必要な権限のみを付与することがよくあります。範囲の制限には、[AWS Identity and Access Management Access Analyzer](#) を使用してユーザーまたは IAM ロールが実際に使用しているアクセス権限を確認し、過度な権限を削除できます。

社内の誰に IAM ロールを作成する権限を持たせるかを判断するのは難しい場合があります。IAM ロールの作成は通常、権限を昇格させる手段です。権限昇格とは、ユーザーが自分の権限やアクセス範囲を拡大するときのことです。例えば、権限が制限されたユーザーが新しい IAM ロールを作成できる場合、そのユーザーは、AdministratorAccess 管理ポリシーが適用された新しい IAM ロールを作成して引き受けることで、権限を昇格させることができます。

企業によっては、IAM ロールのプロビジョニングを、信頼できる個人を集めたチームに限定しています。このアプローチの欠点は、ほとんどすべてが運用に IAM ロール AWS のサービス を必要とするため、このチームがすぐにボトルネックになる可能性があることです。別の方法として、[アクセス許可の境界](#)を使用して、クラウドインフラストラクチャの開発、テスト、起動、管理を行うユーザーのみに IAM アクセスを委任する方法があります。ポリシーの例については、「[Example Permission Boundaries](#)」(GitHub) を参照してください。

プラットフォームチームとも呼ばれる開発運用 (DevOps) チームでは、多くの場合、複数の社内開発チームのセルフサービス機能とアプリケーション運用の安定性とのバランスを取る必要があります。職場における自主性、習得、目的を重視するエンジニアリング文化を育むことで、チームのモチベーションを向上させることができます。エンジニアは、他の人に頼ることなく、自主独立して仕事をすることを望んでいます。DevOps チームがセルフサービスソリューションを実装できれば、物事を済ませる際に他のチームが DevOps チームに頼る時間も短縮されます。

## 許可セットの作成

のアクセス[許可セット](#)を使用して AWS アカウント アクセスを管理できます AWS IAM アイデンティティセンター。許可セットとは、IAM ポリシーを 1 つ以上、複数の AWS アカウントにデプロイすることができるテンプレートのことです。許可セットを AWS アカウントに割り当てると、IAM アイデンティティセンターが IAM ロールを作成し、IAM ポリシーをそのロールにアタッチします。詳細については、「[Create and manage permission sets](#)」(IAM アイデンティティセンターのドキュメント)を参照してください。

AWS では、ビジネス内のさまざまなペルソナにマッピングするアクセス許可セットを作成することをお勧めします。

例えば、次の許可セットを作成できます。

- [請求に対する許可セット](#)
- [開発者許可セット](#)
- [本番稼働用許可セット](#)

次のアクセス許可セットは、AWS CloudFormation テンプレートからのスニペットです。このコードを開始点として使用し、ビジネスに合わせてカスタマイズしてください。CloudFormation テンプレートの詳細については、「[Learn template basics](#)」(CloudFormation ドキュメント)を参照してください。

### 請求に対する許可セット

財務チームは BillingAccessPermissionSet を使用して、各アカウントの AWS Billing コンソールダッシュボードと AWS Cost Explorer を表示します。

```
BillingAccessPermissionSet:  
  Type: "AWS::SSO::PermissionSet"  
  Properties:  
    Description: Access to Billing and Cost Explorer
```

```
InstanceArn: !Sub "arn:${AWS::Partition}:sso::instance/ssoins-instanceId"
ManagedPolicies:
  - !Sub "arn:${AWS::Partition}:iam::aws:policy/job-function/Billing"
Name: BillingAccess
SessionDuration: PT8H
RelayStateType: https://console.aws.amazon.com/billing/home
```

## 開発者許可セット

エンジニアリングチームは、DeveloperAccessPermissionSet を使用して非本番稼働用アカウントにアクセスできます。

```
DeveloperAccessPermissionSet:
  Type: "AWS::SSO::PermissionSet"
  Properties:
    Description: Access to provision resources through CloudFormation
    InlinePolicy: !Sub |-
      {
        "Version": "2012-10-17",
        "Statement": [
          {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:${AWS::Partition}:iam::*:role/CloudFormationRole",
            "Condition": {
              "StringEquals": {
                "aws:ResourceAccount": "${!aws:PrincipalAccount}",
                "iam:PassedToService": "cloudformation.${AWS::URLSuffix}"
              }
            }
          },
          {
            "Effect": "Allow",
            "Action": [
              "cloudformation:ContinueUpdateRollback",
              "cloudformation:CreateChangeSet",
              "cloudformation:CreateStack",
              "cloudformation>DeleteStack",
              "cloudformation:RollbackStack",
              "cloudformation:UpdateStack"
            ],
            "Resource": "arn:${AWS::Partition}:cloudformation::*:stack/app-*",
            "Condition": {
```

```
    "ArnLike": {
      "cloudformation:RoleArn": "arn:${AWS::Partition}:iam:${!
aws:PrincipalAccount}:role/CloudFormationRole"
    },
    "Null": {
      "cloudformation:ImportResourceTypes": true
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "cloudformation:CancelUpdateStack",
      "cloudformation>DeleteChangeSet",
      "cloudformation:DetectStackDrift",
      "cloudformation:DetectStackResourceDrift",
      "cloudformation:ExecuteChangeSet",
      "cloudformation:TagResource",
      "cloudformation:UntagResource",
      "cloudformation:UpdateTerminationProtection"
    ],
    "Resource": "arn:${AWS::Partition}:cloudformation:*:*:stack/app-*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "cloudformation>CreateUploadBucket",
      "cloudformation:ValidateTemplate",
      "cloudformation:EstimateTemplateCost"
    ],
    "Resource": "*"
  }
]
}
InstanceArn: !Sub "arn:${AWS::Partition}:sso:::instance/ssoins-instanceId"
ManagedPolicies:
  - !Sub "arn:${AWS::Partition}:iam::aws:policy/AWSServiceCatalogEndUserFullAccess"
  - !Sub "arn:${AWS::Partition}:iam::aws:policy/AWSBillingReadOnlyAccess"
  - !Sub "arn:${AWS::Partition}:iam::aws:policy/AWSSupportAccess"
  - !Sub "arn:${AWS::Partition}:iam::aws:policy/ReadOnlyAccess"
Name: DeveloperAccess
SessionDuration: PT8H
```

## 本番稼働用許可セット

エンジニアリングチームは、ProductionPermissionSet を使用して、本番稼働用アカウントにアクセスできます。この許可セットには制限があり、閲覧のみのアクセス権しかありません。

```
ProductionPermissionSet:
  Type: "AWS::SSO::PermissionSet"
  Properties:
    Description: Access to production accounts
    InlinePolicy: !Sub |-
      {
        "Version": "2012-10-17",
        "Statement": [
          {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:${AWS::Partition}:iam::*:role/CloudFormationRole",
            "Condition": {
              "StringEquals": {
                "aws:ResourceAccount": "${!aws:PrincipalAccount}",
                "iam:PassedToService": "cloudformation.${AWS::URLSuffix}"
              }
            }
          },
          {
            "Effect": "Allow",
            "Action": "cloudformation:ContinueUpdateRollback",
            "Resource": "arn:${AWS::Partition}:cloudformation::*:stack/app-*",
            "Condition": {
              "ArnLike": {
                "cloudformation:RoleArn": "arn:${AWS::Partition}:iam:${!aws:PrincipalAccount}:role/CloudFormationRole"
              }
            }
          },
          {
            "Effect": "Allow",
            "Action": "cloudformation:CancelUpdateStack",
            "Resource": "arn:${AWS::Partition}:cloudformation::*:stack/app-*"
          }
        ]
      }
    InstanceArn: !Sub "arn:${AWS::Partition}:sso::instance/ssoins-instanceId"
```

```
ManagedPolicies:
  - !Sub "arn:${AWS::Partition}:iam::aws:policy/AWSBillingReadOnlyAccess"
  - !Sub "arn:${AWS::Partition}:iam::aws:policy/AWSSupportAccess"
  - !Sub "arn:${AWS::Partition}:iam::aws:policy/job-function/ViewOnlyAccess"
Name: ProductionAccess
SessionDuration: PT2H
```

## アクセス許可の境界の作成

許可セットをデプロイしたら、アクセス許可の境界を設定します。このアクセス許可の境界とは、クラウドインフラストラクチャを開発、テスト、起動、管理しているユーザーのみに IAM アクセスを委任するメカニズムのことです。これらのユーザーは、ポリシーとアクセス許可の境界で許可されているアクションのみを実行できます。

AWS CloudFormation テンプレートでアクセス許可の境界を定義し、CloudFormation StackSets を使用してテンプレートを複数のアカウントにデプロイできます。これにより、1 回の操作で組織全体に標準化されたポリシーを確立して維持できます。詳細については、「[Working with AWS CloudFormation StackSets](#)」(CloudFormation ドキュメント) を参照してください。

次の CloudFormation テンプレートは、IAM ロールをプロビジョニングし、アクセス許可の境界として機能する IAM ポリシーを作成します。スタックセットを使用すると、このテンプレートを組織内のすべてのメンバーアカウントにデプロイできます。

```
CloudFormationRole:
  Type: "AWS::IAM::Role"
  Properties:
    AssumeRolePolicyDocument:
      Version: "2012-10-17"
      Statement:
        Effect: Allow
        Principal:
          Service: !Sub "cloudformation.${AWS::URLSuffix}"
        Action: "sts:AssumeRole"
      Condition:
        StringEquals:
          "aws:SourceAccount": !Ref "AWS::AccountId"
    Description: !Sub "DO NOT DELETE - Used by CloudFormation. Created by CloudFormation ${AWS::StackId}"
    ManagedPolicyArns:
      - !Sub "arn:${AWS::Partition}:iam::aws:policy/AdministratorAccess"
    PermissionsBoundary: !Ref DeveloperBoundary
```

```
RoleName: CloudFormationRole
```

```
DeveloperBoundary:
```

```
Type: "AWS::IAM::ManagedPolicy"
```

```
Properties:
```

```
Description: Permission boundary for developers
```

```
ManagedPolicyName: PermissionsBoundary
```

```
PolicyDocument:
```

```
Version: "2012-10-17"
```

```
Statement:
```

```
- Sid: AllowModifyIamRolesWithBoundary
```

```
Effect: Allow
```

```
Action:
```

- "iam:AttachRolePolicy"
- "iam:CreateRole"
- "iam>DeleteRolePolicy"
- "iam:DetachRolePolicy"
- "iam:PutRolePermissionsBoundary"
- "iam:PutRolePolicy"

```
Resource: !Sub "arn:${AWS::Partition}:iam::${AWS::AccountId}:role/app/*"
```

```
Condition:
```

```
ArnEquals:
```

```
"iam:PermissionsBoundary": !Sub "arn:${AWS::Partition}:iam::  
${AWS::AccountId}:policy/PermissionsBoundary"
```

```
- Sid: AllowModifyIamRoles
```

```
Effect: Allow
```

```
Action:
```

- "iam>DeleteRole"
- "iam:TagRole"
- "iam:UntagRole"
- "iam:UpdateAssumeRolePolicy"
- "iam:UpdateRole"
- "iam:UpdateRoleDescription"

```
Resource: !Sub "arn:${AWS::Partition}:iam::${AWS::AccountId}:role/app/*"
```

```
- Sid: OverlyPermissiveAllowedServices
```

```
Effect: Allow
```

```
Action:
```

- "lambda:\*"
- "apigateway:\*"
- "events:\*"
- "s3:\*"
- "logs:\*"

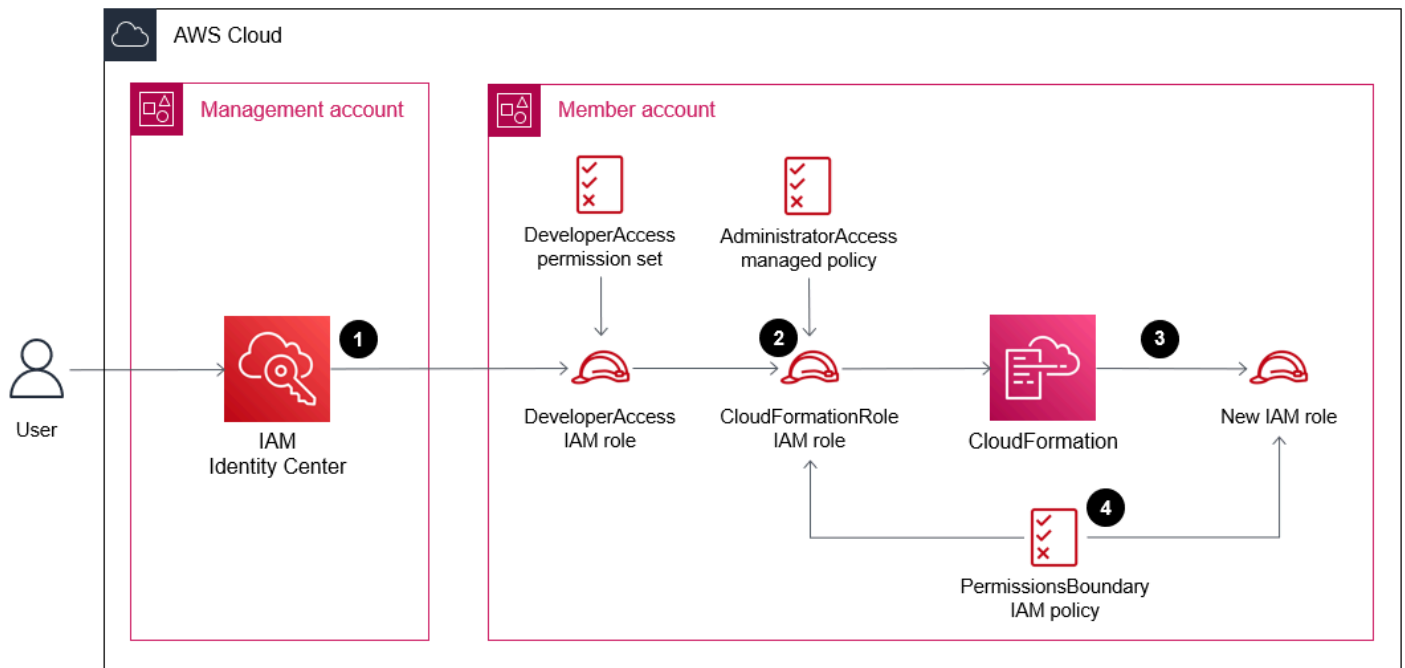
```
Resource: "*"
```

CloudFormationRole ロール、PermissionsBoundary ポリシー、DeveloperAccess 許可セットが連携して次の権限を付与します。

- ユーザーは、ReadOnlyAccess AWS 管理ポリシーを通じて AWS のサービス、ほとんどの への読み取り専用アクセス権を持ちます。
- ユーザーは、AWSSupportAccess AWS 管理ポリシーを通じて、オープンサポートケースにアクセスできます。
- ユーザーは、AWSBillingReadOnlyAccess AWS 管理ポリシーを通じて、AWS Billing コンソールダッシュボードに読み取り専用でアクセスできます。
- ユーザーは、AWSServiceCatalogEndUserFullAccess AWS 管理ポリシーを通じて、Service Catalog から製品をプロビジョニングできます。
- インラインポリシーによって、CloudFormation テンプレートのコストをどれでも検証して見積もることができます。
- CloudFormationRole IAM ロールを使用することで、ユーザーは、app/ で始まる CloudFormation スタックを作成、更新、削除できます。
- CloudFormation を使用して、ユーザーは app/ で始まる IAM ロールを作成、更新、削除できます。PermissionsBoundary IAM ポリシーは、ユーザーが権限を昇格することを防ぎます。
- ユーザーは AWS Lambda、CloudFormation を使用してのみ、Amazon EventBridge、Amazon CloudWatch、Amazon Simple Storage Service (Amazon S3)、および Amazon API Gateway リソースをプロビジョニングできます。

次の画像は、開発者などの権限を持つユーザーが、このガイドで説明されている許可セット、IAM ロール、アクセス許可の境界を使用してメンバーアカウントに新しい IAM ロールを作成する方法を示しています。

1. ユーザーは IAM アイデンティティセンターで認証を行い、DeveloperAccess IAM ロールを引き受けます。
2. ユーザーが cloudformation:CreateStack アクションを実行し、CloudFormationRole のIAM ロールを引き受けます。
3. ユーザーが iam:CreateRole アクションを実行し、CloudFormation を使用して新しい IAM ロールを作成します。
4. PermissionsBoundary のIAM ポリシーが新しい IAM ロールに適用されます。



CloudFormationRole のロールには [AdministratorAccess](#) 管理ポリシーがアタッチされていますが、PermissionsBoundary IAM ポリシーにより CloudFormationRole のロールの有効な許可は PermissionsBoundary ポリシーと同じになります。PermissionsBoundary ポリシーは、iam:CreateRole アクションの許可時に自身のポリシーを参照します。これにより、アクセス許可の境界が適用されている場合にのみロールを作成できます。

## 個人へのアクセス許可の管理

許可セット、アクセス許可の境界、CloudFormationRole の IAM ロールを使用することで、個々のプリンシパルに直接割り当てる必要があるアクセス許可の量を制限できます。これにより、会社の成長に合わせてアクセスを管理し、最小特権を付与するというセキュリティのベストプラクティスを適用できます。

サービスにリンクされたロールを使用することもできます。これは、ユーザーに代わってリソースをプロビジョニングする権限を AWS サービスに付与するものです。IAM プリンシパル (ユーザー、ユーザーグループ、ロール) にアクセス権限を付与する代わりに、サービスにアクセス権限を付与できます。たとえば、のサービスにリンクされたロール [AWS Service Catalog](#) を使用すると、IAM プリンシパルにアクセス許可を割り当てることなく、独自のテンプレート、リソース、環境をプロビジョニングできます。詳細については、「[IAM と連携する AWS のサービス](#)」および「[サービスリンクロールの使用](#)」(IAM ドキュメント) を参照してください。

別のベストプラクティスは、個人の AWS マネジメントコンソールへアクセスできる量を制限することです。コンソールへのアクセスを制限することで、[AWS CloudFormation](#)、[HashiCorp Terraform](#)、[Pulumi](#) などの Infrastructure as Code (IaC) 技術を使用してリソースをプロビジョニングするよう個人に要求することができます。IaC によるインフラストラクチャの管理では、時間の経過に伴うリソースの変化を追跡し、GitHub のプルリクエストなどの変更を承認するメカニズムを導入できます。

# マルチアカウントアーキテクチャのネットワーク接続

## VPC の接続

多くの企業で、Amazon Virtual Private Cloud (Amazon VPC) の VPC ピアリングを使用して、開発用 VPC と本番稼働用 VPC に接続しています。VPC ピアリング接続を使用すると、プライベート IP アドレスを使用して 2 つの VPC 間でトラフィックをルーティングできます。接続 VPCs は、異なる AWS アカウントと異なるにすることができます AWS リージョン。詳細については、「[VPC ピア機能とは](#)」(Amazon VPC ドキュメント) を参照してください。企業が成長し、VPC の数が増えるにつれて、すべての VPC 間のピアリング接続を維持することがメンテナンスの負担になることがあります。VPC あたりの VPC ピアリング接続の最大数によって制限がある場合もあります。詳細については、「[VPC ピアリング接続クォータ](#)」(Amazon VPC ドキュメント) を参照してください。

複数のにまたがって非本番稼働データをホストする複数の開発、テスト、ステージング環境がある場合は AWS アカウント、それらのすべての VPCs 間でネットワーク接続を提供するが、本番稼働環境へのアクセスは許可しないことをお勧めします。[AWS Transit Gateway](#) を使用して複数のアカウントにまたがる VPC を複数接続できます。ルートテーブルを分離することで、集中型ルーターとして機能するトランジットゲートウェイを介して開発用 VPC が本番稼働用 VPC と通信するのを防ぐことができます。詳細については、「[集中型ルーター](#)」(Transit Gateway ドキュメント) を参照してください。

Transit Gateway は、AWS アカウントや AWS リージョンが異なるものも含め、他のトランジットゲートウェイとのピアリングもサポートしています Transit Gateway はフルマネージド型の可用性の高いサービスであるため、リージョンごとにプロビジョニングする必要があるトランジットゲートウェイは 1 つだけです。

詳細および詳細なネットワークアーキテクチャについては、「[スケーラブルで安全なマルチ VPC AWS ネットワークインフラストラクチャの構築](#)」(AWS ホワイトペーパー) を参照してください。

## アプリケーションの接続

同じ環境 (本番環境など) 内の異なる AWS アカウントのアプリケーション間で通信を確立する必要がある場合は、次のいずれかのオプションを使用できます。

- [VPC ピアリング](#) や [AWS Transit Gateway](#) は、複数の IP アドレスやポートに幅広くアクセスする場合に、ネットワークレベルで接続を提供できます。

- [AWS PrivateLink](#) は、VPC のプライベートサブネットにエンドポイントを作成し、これらのエンドポイントは [Amazon Route 53 Resolver](#) に DNS エントリとして登録されます。DNS を使用することにより、アプリケーションは、VPC に NAT ゲートウェイやインターネットゲートウェイを必要とせずに、エンドポイントを解決して登録済みのサービスに接続できます。
- [Amazon VPC Lattice](#) は、複数のアカウントと VPC にまたがるアプリケーションなどのサービスを関連付け、サービスネットワークにまとめます。サービスネットワークに関連付けられた VPC 内のクライアントは、同じアカウントに属しているかどうかに関係なく、サービスネットワークに関連する他のすべてのサービスにリクエストを送信できます。VPC Lattice は AWS Resource Access Manager (AWS RAM) と統合されているため、他のアカウントや を通じてリソースを共有できます AWS Organizations。VPC は、1 つのサービスネットワークにのみ関連付けることができます。このソリューションでは VPC ピアリングや AWS Transit Gateway を使用してアカウント間の通信を行う必要がありません。

## ネットワーク接続のベストプラクティス

- 一元化 AWS アカウント されたネットワークに使用する を作成します。このアカウントに network-prod という名前を付け、AWS Transit Gateway と [Amazon VPC IP Address Manager](#) (IPAM) に使用します。このアカウントを Infrastructure\_Prod の組織単位の追加します。
- [AWS Resource Access Manager](#) (AWS RAM) を使用して、トランジットゲートウェイ、VPC Lattice サービスネットワーク、IPAM プールを組織の他のメンバーと共有します。これにより、組織 AWS アカウント 内のすべての がこれらのサービスとやり取りできるようになります。
- IPAM プールを使用して IPv4 と IPv6 のアドレス割り当てを一元管理することで、エンドユーザーが [AWS Service Catalog](#) を使用して VPC を自分でプロビジョニングできるようになります。これにより、VPC のサイズを適切に設定し、IP アドレス空間の重複を防ぐことができます。
- インターネットへのトラフィックにはエグレスの一元化アプローチを使用し、インターネットから環境に入るトラフィックにはイングレスの分散化アプローチを使用します。詳細については、「[エグレスの一元化](#)」および「[イングレスの分散化](#)」を参照してください。

## エグレスの一元化

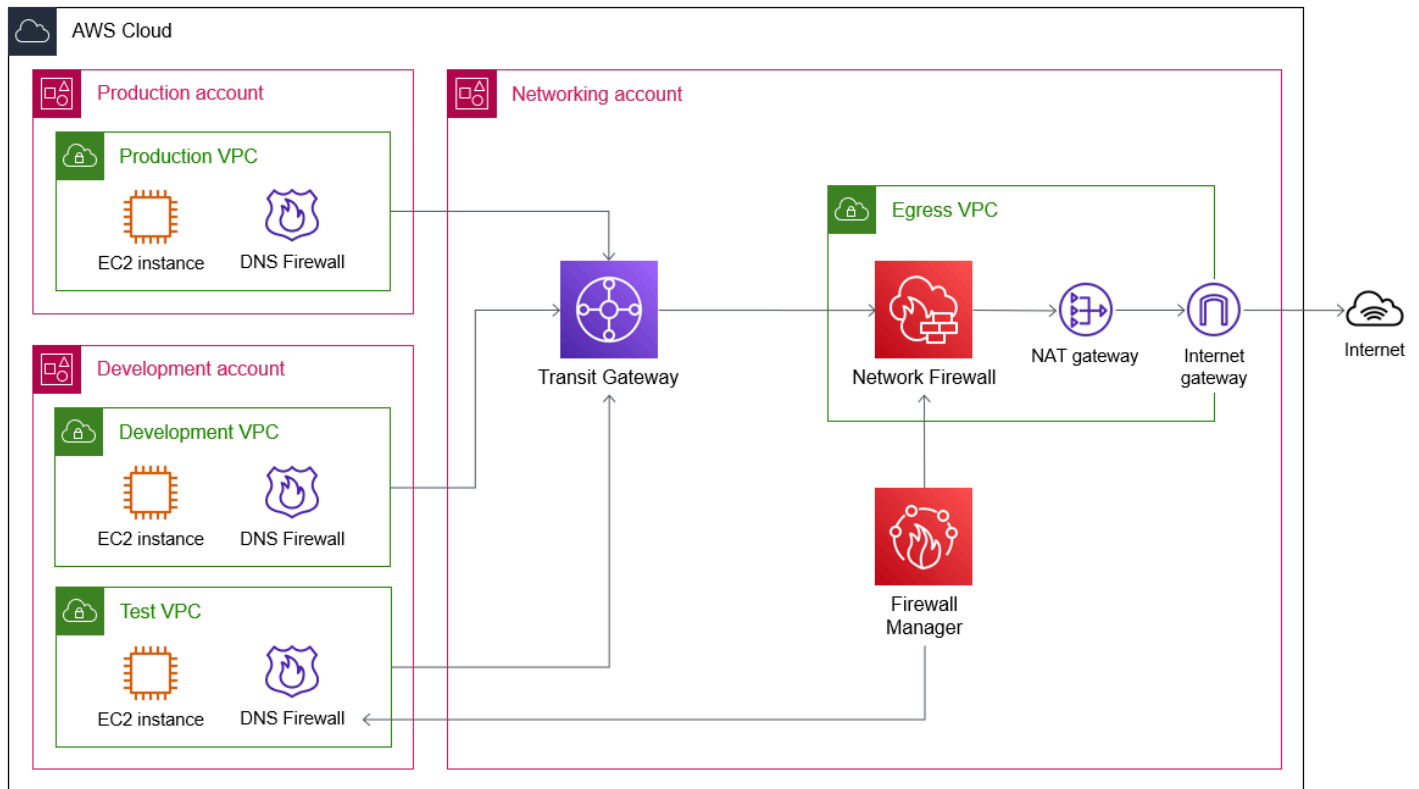
一元的な出力は、インターネット宛てのすべてのネットワークトラフィックに 1 つの共通のエントリポイントを使用する原則です。このエントリポイントで検査を設定し、指定したドメインへのトラフィックのみを許可するか、指定したポートまたはプロトコルを介してのみトラフィックを許可できます。また、エグレスを一元化することで、インターネットに到達するために各 VPC に NAT ゲートウェイをデプロイする必要がなくなるため、コスト削減にも役立ちます。これにより、マルチアカウント

のコマンドアンドコントロール (C&C) インフラストラクチャなど、外部からアクセス可能な悪意のあるリソースへの露出が制限されるため、セキュリティの観点からはメリットがあります。一元的な出力の詳細とアーキテクチャオプションについては、[「インターネットへの一元的な出力」](#) (AWS ホワイトペーパー) を参照してください。

ステートフルでマネージド型のネットワークファイアウォールであり、侵入検知と防止のサービスである [AWS Network Firewall](#) を、送信トラフィックの一元化された検査ポイントとして使用できます。このファイアウォールは、送信トラフィック専用の VPC で設定します。Network Firewall は、インターネットアクセスを特定のドメインに制限するのに使用できるステートフルルールをサポートしています。詳細については、Network Firewall ドキュメントの「[Domain filtering](#)」を参照してください。

[Amazon Route 53 Resolver DNS ファイアウォール](#) を使用して、特定のドメイン名への送信トラフィックを制限することもできます。主な目的は、データの不正流出を防ぐことです。DNS ファイアウォールルールでは [ドメインリスト](#) (Route 53 ドキュメント) を適用して指定したドメインへのアクセスを許可または拒否することができます。悪意のあるアクティビティやその他の潜在的な脅威に関連付けられているドメイン名を含む AWS マネージドドメインリストを使用することも、カスタムドメインリストを作成することもできます。DNS ファイアウォールルールグループを作成して VPC に適用します。アウトバウンド DNS リクエストは VPC のリゾルバーを経由してドメイン名を解決し、DNS ファイアウォールは VPC に適用されたルールグループに基づいてリクエストをフィルタリングします。リゾルバーに送られる再帰的な DNS リクエストは、トランジットゲートウェイと Network Firewall パスを経由しません。Route 53 Resolver と DNS ファイアウォールは VPC からの独立したエグレスパスと見なす必要があります。

次の図は、エグレスの一元化を示すサンプルアーキテクチャです。ネットワーク通信が開始される前に、DNS リクエストが Route 53 Resolver に送信され、DNS ファイアウォールが通信に使用される IP アドレスの解決を許可または拒否します。インターネットに向かうトラフィックは、一元化されたネットワークアカウントのトランジットゲートウェイにルーティングされます。トランジットゲートウェイは、検査のためにトラフィックを Network Firewall に転送します。ファイアウォールポリシーで送信トラフィックが許可されている場合、トラフィックは NAT ゲートウェイ、インターネットゲートウェイを経由して、インターネットに送信されます。を使用して AWS Firewall Manager、マルチアカウントインフラストラクチャ全体で DNS Firewall ルールグループと Network Firewall ポリシーを一元管理できます。



## 送信トラフィックを保護するためのベストプラクティス

- [ログ記録専用モード](#) (Route 53 ドキュメント) で開始します。正当なトラフィックが影響を受けないことを確認したら、ブロックモードに変更します。
- [AWS Firewall Manager ネットワークアクセスコントロールリストのポリシー](#) を使用するか、を使用して、インターネットへの DNS トラフィックをブロックします AWS Network Firewall。すべての DNS クエリは Route 53 Resolver を経由する必要があります。ここでは、Amazon GuardDuty でモニタリングし (有効になっている場合)、[Route 53 Resolver DNS Firewall](#) でフィルタリングできます (有効になっている場合)。詳細については、「[VPC とネットワークの間における DNS クエリの解決](#)」 (Route 53 ドキュメント) を参照してください。
- DNS ファイアウォールと Network Firewall の [AWS マネージドドメインリスト](#) (Route 53 ドキュメント) を使用します。
- .info、.top、.xyz など、リスクが高く、あまり使われないトップレベルドメインや、一部の国コードドメインをブロックすることを検討してください。
- ポート 1389、4444、3333、445、135、139、53 など、リスクが高く、あまり使われないポートをブロックすることを検討してください。

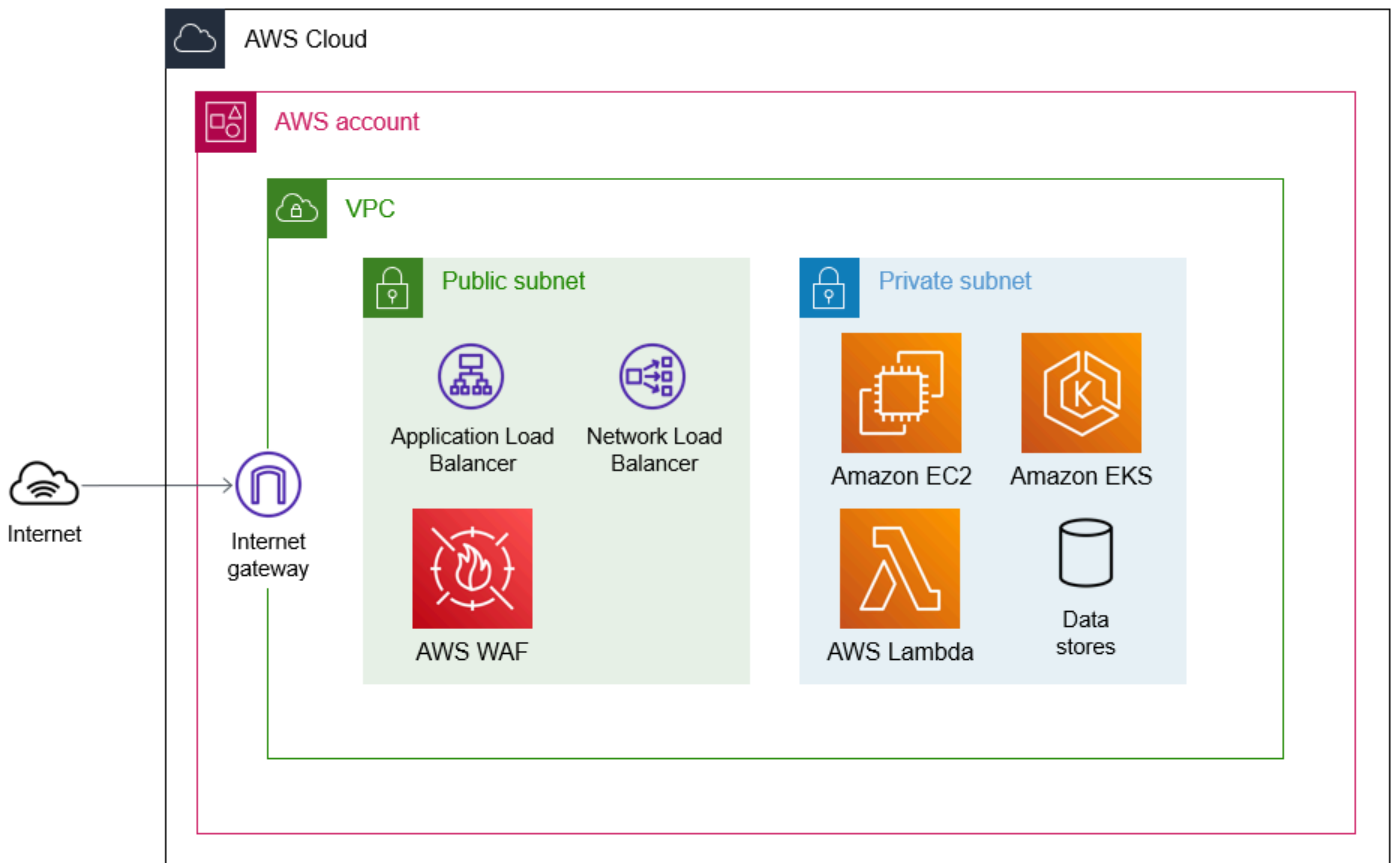
- 開始点として、AWS マネージドルールを含む拒否リストを使用できます。その後、時間の経過とともに許可リストモデルの実装に取り組むことができます。たとえば、許可リストに完全修飾ドメイン名の厳密なリストのみを含める代わりに、\*.example.com などのいくつかのワイルドカードを使用して開始します。予想される最上位ドメインのみを許可し、他のすべてのドメインをブロックすることもできます。次に、時間の経過とともに、これらも絞り込みます。
- [Route 53 プロファイル](#) (Route 53 ドキュメント) を使用して、DNS 関連の Route 53 設定を多くの VPCs と異なるに適用します AWS アカウント。
- これらのベストプラクティスの例外を処理するプロセスを定義します。

## イングレスの分散化

イングレスの分散化とは、インターネットからのトラフィックがアカウントのワークロードに到達する方法を、個々のアカウントレベルで定義する原則のことです。マルチアカウントアーキテクチャにおいて、イングレス分散化の利点の 1 つは、各アカウントがワークロードに最適なイングレスサービスまたはリソース (Application Load Balancer、Amazon API Gateway、Network Load Balancer など) を使用できることです。

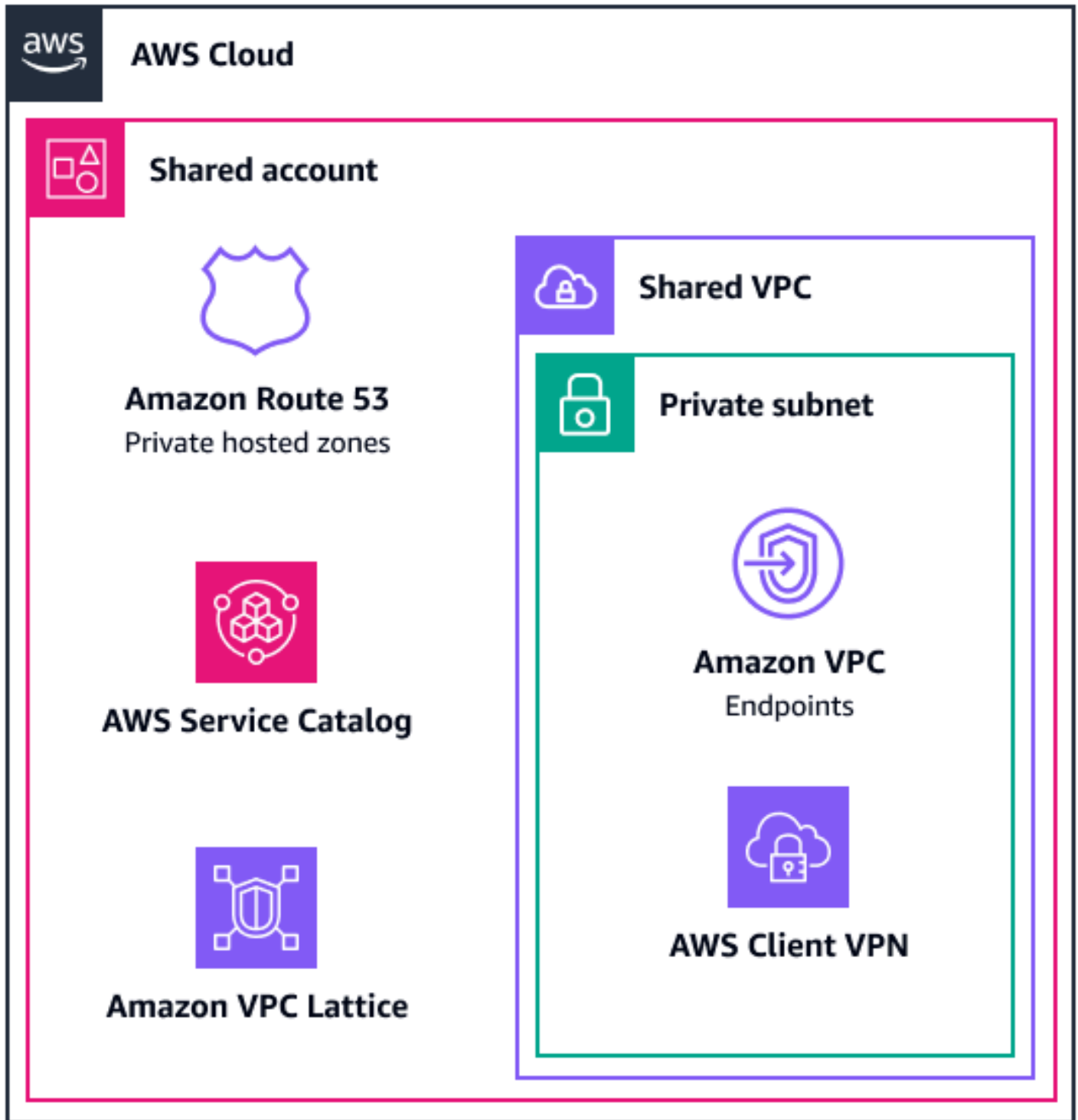
イングレスが分散されていると、各アカウントを個別に管理する必要がありますが、[AWS Firewall Manager](#) から構成を一元的に管理、維持することができます。Firewall Manager は、[AWS WAF](#) や [Amazon VPC セキュリティグループ](#) などの保護をサポートしています。Application Load Balancer、Amazon CloudFront、API Gateway、または AWS WAF に関連付けることができます AWS AppSync。エグレス VPC とトランジットゲートウェイを使用している場合、[エグレスの一元化](#) で説明されているように、各スポーク VPC にはパブリックサブネットとプライベートサブネットが含まれます。ただし、トラフィックはネットワークアカウントのエグレス VPC を経由するため、NAT ゲートウェイをデプロイする必要はありません。

次の図は、インターネットにアクセス可能なワークロードを含む単一の VPC AWS アカウントを持つ個人の例を示しています。インターネットからのトラフィックは、インターネットゲートウェイを経由して VPC にアクセスし、パブリックサブネットでホストされている負荷分散サービスとセキュリティサービスに到達します。(パブリックサブネットには、インターネットゲートウェイへのデフォルトルートがあります)。ロードバランサーをパブリックサブネットにデプロイし、AWS WAF アクセスコントロールリスト (ACLs) をアタッチして、クロスサイトスクリプティングなどの悪意のあるトラフィックから保護します。アプリケーションをホストするワークロードをプライベートサブネットにデプロイします。プライベートサブネットは、インターネットと直接アクセスすることはできません。



組織に VPC が多数ある場合は、専用の AWS アカウントと共有のアカウントにインターフェイス VPC エンドポイント、またはプライベートホストゾーンを作成することで、一般的な AWS のサービスを共有できます。詳細については、[「インターフェイス VPC エンドポイント AWS のサービスを使用した へのアクセス \(AWS PrivateLink ドキュメント\)」](#) および [「プライベートホストゾーンの使用 \(Route 53 ドキュメント\)」](#) を参照してください。

次の図は、組織全体で共有できるリソース AWS アカウント をホストする の例を示しています。VPC エンドポイントは、専用 VPC で作成すると複数のアカウントで共有できます。VPC エンドポイントを作成する場合、オプションで、エンドポイントの DNS エントリを AWS に管理させることができます。エンドポイントを共有するには、このオプションをオフにし、別の Route 53 プライベートホストゾーン (PHZ) に DNS エントリを作成します。その後、PHZ を組織内のすべての VPC に関連付けると、VPC エンドポイントの一元的な DNS 解決を行うことができます。また、トランジットゲートウェイのルートテーブルに、共有 VPC から他の VPC へのルートが含まれていることを確認する必要があります。詳細については、[「インターフェイス VPC エンドポイントへの集中アクセス」](#) (AWS ホワイトペーパー) を参照してください。



共有 AWS アカウントは、AWS Service Catalog ポートフォリオをホストするのにも最適な場所です。ポートフォリオは、デプロイに使用する IT サービスのコレクションであり AWS、ポートフォリオにはそれらのサービスの設定情報が含まれています。共有アカウントでポートフォリオを作成し、組織と共有できます。その後、各メンバーアカウントはポートフォリオを独自のリージョン

Service Catalog インスタンスにインポートします。詳細については、「[AWS Organizationsとの共有](#)」(Service Catalog ドキュメント) を参照してください。

同様に、Amazon VPC Lattice では、共有アカウントを使用して環境テンプレートとサービステンプレートを実体として一元管理し、組織メンバーアカウントとのアカウント接続を設定できます。詳細については、「[VPC Lattice 実体を共有する](#)」(VPC Lattice ドキュメント) を参照してください。

# マルチアカウントアーキテクチャのセキュリティインシデント対応

複数のアカウントに移行するときは AWS アカウント、組織内で発生する可能性のあるセキュリティイベントを可視化することが重要です。[ID 管理とアクセス制御](#) で、AWS Control Tower を使用してランディングゾーンをセットアップしました。そのセットアッププロセス中に、はセキュリティ AWS アカウントのために AWS Control Tower を指定しました。セキュリティサービスの管理は、security-tooling-prod アカウントに委任する必要があります。このアカウントを使用してセキュリティサービスを一元管理します。

このガイドでは、AWS アカウントと組織を保護するために以下を使用する AWS のサービス方法について説明します。

- [Amazon GuardDuty](#)
- [Amazon Macie](#)
- [AWS Security Hub CSPM](#)

## Amazon GuardDuty

[Amazon GuardDuty](#) は、AWS CloudTrail イベントログなどのデータソースを分析する継続的なセキュリティモニタリングサービスです。サポートされているデータソースのリストについては、「[基礎データソース](#)」(GuardDuty ドキュメント)を参照してください。悪意のある IP アドレスやドメインのリストなどの脅威インテリジェンスフィールドおよび機械学習を使用して、AWS 環境内での予期しない、および潜在的に未許可で悪意のあるアクティビティを識別します。

で GuardDuty を使用する場合 AWS Organizations、組織の管理アカウントは、組織内の任意のアカウントを GuardDuty の委任管理者に指定できます。委任された管理者がそのリージョンの GuardDuty 管理者アカウントになります。GuardDuty は、そのリージョンで自動的に有効になりAWS リージョン、委任管理者アカウントには、そのリージョン内の組織内のすべてのアカウントに対して GuardDuty を有効化および管理するためのアクセス許可があります。詳細については、「[AWS Organizationsを使用した GuardDuty アカウントの管理](#)」(GuardDuty ドキュメント)を参照してください。

GuardDuty はリージョンレベルのサービスです。つまり、監視する各リージョンで、GuardDuty を有効にする必要があります。

## ベストプラクティス

- サポートされているすべての AWS リージョンで GuardDuty を有効にします。GuardDuty は積極的に使用されていないリージョンでも、許可されていないアクティビティや異常なアクティビティに関する検出結果を生成できます。GuardDuty の料金は、分析したイベントの数に基づいています。ワークロードを運用していない地域でも、GuardDuty を有効にしておくことで、悪意のあるアクティビティの可能性のあるものを警告する効果的でコスト効率の高い検出ツールになります。GuardDuty が利用できるリージョンについては、「[Amazon GuardDuty サービスエンドポイント](#)」(AWS 全般のリファレンス)を参照してください。
- すべてのリージョンで、security-tooling-prod アカウントに組織の GuardDuty 管理を委任します。詳細については、「[GuardDuty の委任された管理者の指定](#)」(GuardDuty ドキュメント)を参照してください。
- 組織に追加された新しい AWS アカウントを自動的に登録するように AWS アカウント GuardDuty を設定します。詳細については、GuardDuty ドキュメントの「[AWS Organizationsを使用したアカウントの管理](#)」にある「ステップ 3 - アカウントをメンバーとして組織に追加する」を参照してください。

## Amazon Macie

[Amazon Macie](#) は、フルマネージド型のデータセキュリティおよびデータプライバシーサービスです。機械学習とパターンマッチングを使用して、Amazon Simple Storage Service (Amazon S3) 内の機密データを検出、モニタリング、保護するのに役立ちます。Amazon Relational Database Service (Amazon RDS) と Amazon DynamoDB (S3 バケット) からデータをエクスポートし、Macie を使用してデータをスキャンできます。

Macie を使用する場合は、AWS Organizations、組織の管理アカウントは、組織内の任意のアカウントを Macie 管理者アカウントとして指定できます。管理者アカウントは、組織のメンバーアカウントの Macie を有効にして管理したり、Amazon S3 インベントリデータにアクセスしたりすることができます。アカウントの機密データ検出ジョブを実行することもできます。詳細については、「[Managing accounts with AWS Organizations](#)」(Macie ドキュメント)を参照してください。

Macie はリージョン別サービスです。つまり、監視する各リージョンで Macie を有効にする必要があります。Macie 管理者アカウントは同じリージョン内のメンバーアカウントのみを管理します。

## ベストプラクティス

- 「[Considerations and recommendations for using Macie with AWS Organizations](#)」(Macie ドキュメント)を順守してください。

- すべてのリージョンで、security-tooling-prod アカウントに組織の Macie 管理を委任します。複数の Macie アカウントを一元管理するには AWS リージョン、管理アカウントは、組織が現在 Macie を使用している、または使用する各リージョンにログインし、それらの各リージョンで Macie 管理者アカウントを指定する必要があります。次に Macie 管理者アカウントは、それらの各リージョンで組織を設定できます。詳細については、「[Integrating and configuring an organization](#)」(Macie ドキュメント)を参照してください。
- Macie には、機密データ検出ジョブの [月間無料利用枠](#)があります。Amazon S3 に機密データが保存されている可能性がある場合は、毎月の無料利用枠の一部として Macie を使用して S3 バケットを分析してください。無料利用枠を超えると、アカウントの機密データ検出料金が発生します。

## AWS Security Hub CSPM

[AWS Security Hub CSPM](#) では、のセキュリティ状態を包括的に把握できます AWS。セキュリティ業界の標準とベストプラクティスに照らしてお使いの環境をチェックできます。Security Hub CSPM は、すべての、サービス (GuardDuty と Macie を含む) AWS アカウント、およびサポートされているサードパーティーパートナー製品からセキュリティデータを収集します。Security Hub CSPM は、セキュリティの傾向を分析し、最も優先度の高いセキュリティ問題を特定するのに役立ちます。Security Hub CSPM には、各でコンプライアンスチェックを実行できるように、さまざまなセキュリティ標準が用意されています AWS アカウント。

で Security Hub CSPM を使用する場合 AWS Organizations、組織内の管理アカウントは、組織内の任意のアカウントを Security Hub CSPM 管理者アカウントとして指定できます。Security Hub CSPM 管理者アカウントは、組織内の他のメンバーアカウントを有効化および管理できます。詳細については、「[AWS Organizations を使用してアカウントを管理する](#)」(Security Hub CSPM ドキュメント)を参照してください。

Security Hub CSPM はリージョン別サービスです。つまり、分析する各リージョンで Security Hub CSPM を有効にし、各リージョンの委任管理者を定義 AWS Organizations する必要があります。

### ベストプラクティス

- [前提条件と推奨事項](#) (Security Hub CSPM ドキュメント) に従ってください。
- すべてのリージョンで、security-tooling-prod アカウントを委任して、組織の Security Hub CSPM を管理します。詳細については、「[Security Hub CSPM 管理者アカウントの指定](#)」(Security Hub CSPM ドキュメント)を参照してください。
- 新しいが組織に追加 AWS アカウントされると自動的に登録されるように Security Hub CSPM を設定します。

- [AWS Foundational Security Best Practices 標準](#) (Security Hub CSPM ドキュメント) を有効にして、リソースがセキュリティのベストプラクティスから逸脱した場合を検出します。
- [クロスリージョン集約](#) (Security Hub CSPM ドキュメント) を有効にして、1つのリージョンからすべての Security Hub CSPM の検出結果を表示および管理できるようにします。

## マルチアカウントアーキテクチャのバックアップ設定

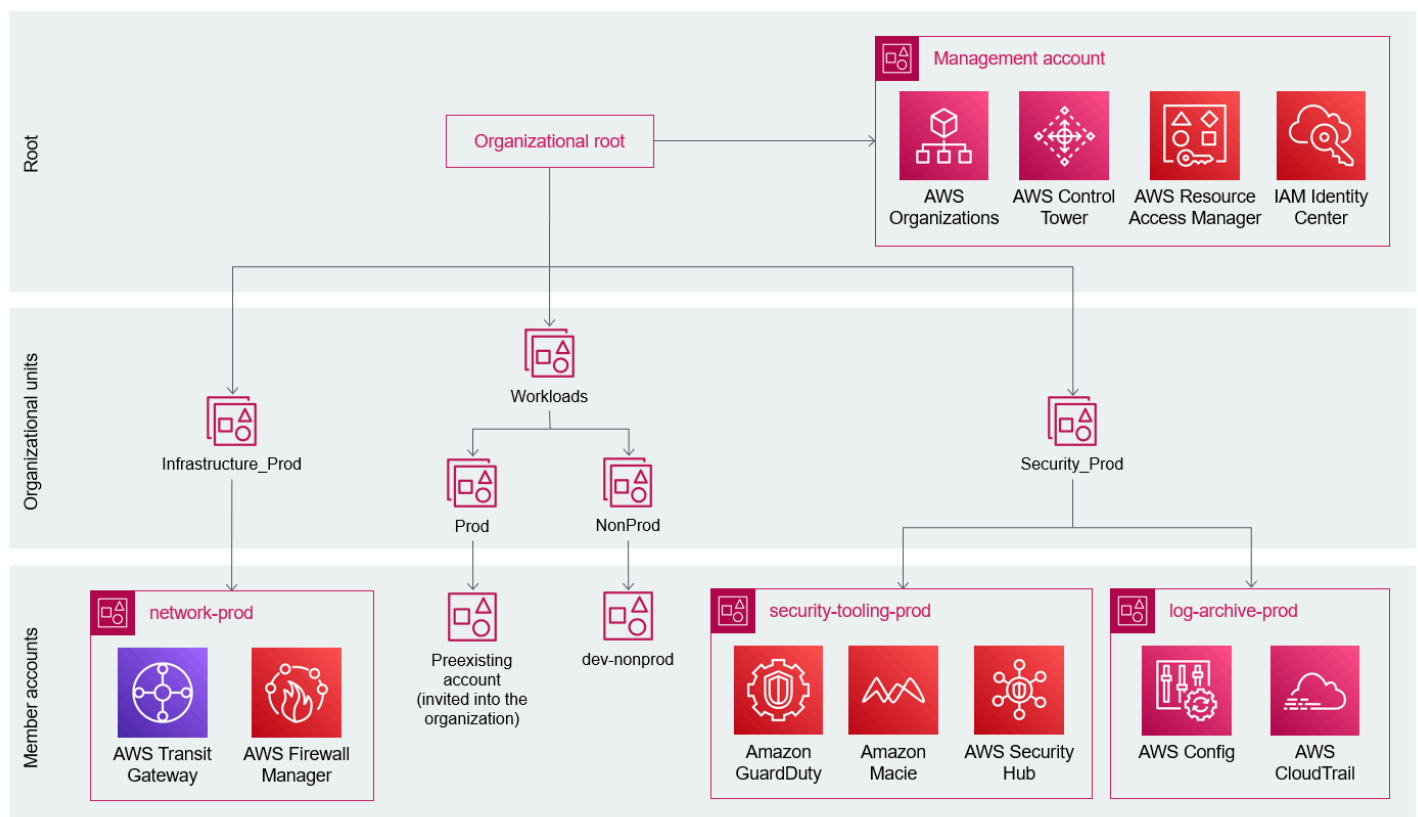
包括的なバックアップ戦略は、セキュリティイベントによって持続する可能性のあるあらゆる影響に耐え、回復し、影響を軽減するために、企業のデータ保護計画にとって欠かせないものです。バックアップポリシーは、組織内のアカウント全体で、リソースのバックアップ戦略を標準化し、実装するのに役立ちます。リソースのバックアッププランの設定とデプロイは、バックアップポリシーで行うことができます。詳細については、[「バックアップポリシー \(AWS Organizations ドキュメント\)」](#)を参照してください。詳細については、[「Top 10 security best practices for secure backups in AWS \(AWS Prescriptive Guidance\)」](#)を参照してください。

# マルチアカウントアーキテクチャに移行する際のアカウントの移行

[既存のアカウントを招待する](#) で、既存のアカウントをワークロード > Prod 組織単位に参加するよう招待しました。このアカウントは、組織の一部として管理されます。

また、ワークロード > NonProd 組織単位に新しい dev-nonprod アカウントをプロビジョニングしました。これで、チームメンバーはを通じて適切なアカウントにアクセスできるようになります AWS IAM アイデンティティセンター。AWS Identity and Access Management (IAM) で個々のユーザーアカウントを削除します。

このガイドの推奨事項に従った場合、組織は次のような構造になります。



既存のアカウント内で実行中のワークロードがある場合は、[スコーピングの基準を定義する](#) で設定した基準に従って、これらのワークロードを独立したアカウントに移行します。非本番稼働用のワークロードを新しい dev-nonprod 組織単位に移行し、本番環境用のワークロードを network-prod アカウントに移行します。一般的な AWS リソースの移行の詳細については、このガイドの次のセクション「」を参照してください [リソース移行](#)。

# リソースレプリケーションまたは 間の移行 AWS アカウント

単一アカウントアーキテクチャから AWS アカウント マルチアカウントアーキテクチャに移行した後は、本番ワークロードと非本番ワークロードが既存のアカウントで実行されることが一般的です。これらのリソースを専用の本番用アカウントと非本番用アカウント、または組織単位に移行することで、これらのワークロードへのアクセスとネットワークを管理しやすくなります。一般的な AWS リソースを別のリソースに移行するためのオプションを以下に示します AWS アカウント。

このセクションでは、AWS アカウント間でデータを複製する戦略に焦点を当てます。アカウント間でコンピューティングリソースを複製する必要がないように、ワークロードはできるだけステートレスになるように努める必要があります。また、環境を別の AWS アカウントに再プロビジョニングできるように、Infrastructure as Code (IaC) からリソースを管理することも有益です。

このセクションでは、以下のデータリソースを移行するためのオプションについて説明します。

- [AWS AppConfig 設定と環境](#)
- [AWS Certificate Manager 証明書](#)
- [Amazon CloudFront デイストリビューション](#)
- [AWS CodeArtifact ドメインとリポジトリ](#)
- [Amazon DynamoDB テーブル](#)
- [Amazon EBS ボリューム](#)
- [Amazon EC2 インスタンスまたは AMI](#)
- [Amazon ECR レジストリ](#)
- [Amazon EFS ファイルシステム](#)
- [Amazon ElastiCache \(Redis OSS\) クラスター](#)
- [AWS Elastic Beanstalk 環境](#)
- [Elastic IP アドレス](#)
- [AWS Lambda レイヤー](#)
- [Amazon Lightsail インスタンス](#)
- [Amazon Neptune クラスター](#)
- [Amazon OpenSearch Service ドメイン](#)
- [Amazon RDS スナップショット](#)

- [Amazon Redshift クラスター](#)
- [Amazon Route 53 のドメインとホストゾーン](#)
- [Amazon S3 バケット](#)
- [Amazon SageMaker AI モデル](#)
- [AWS WAF ウェブ ACLs](#)

## AWS AppConfig 設定と環境

AWS AppConfig では、設定を別の AWS アカウントに直接コピーすることはできません。ただし、環境をホスト AWS アカウントしているとは別に AWS AppConfig 設定と環境を管理するのがベストプラクティスです。詳細については、「[Cross-account configuration with AWS AppConfig \(AWS blog post\)](#)」を参照してください。

## AWS Certificate Manager 証明書

証明書のプライベートキーの暗号化に使用される AWS Certificate Manager (ACM) キーは AWS リージョンおよびアカウントごとに一意であるため、(ACM AWS Key Management Service AWS KMS) 証明書のあるアカウントから別のアカウントに直接エクスポートすることはできません。ただし、複数のアカウントとリージョンにある同じドメイン名の複数の証明書を同時にプロビジョニングできます。ACM は DNS (推奨) または E メールによるドメイン所有権の検証をサポートしています。DNS 検証を使用して新しい証明書を作成すると、ACM は証明書のすべてのドメインに固有の CNAME レコードを生成します。CNAME レコードはアカウントごとに異なるため、証明書を適切に検証するには 72 時間以内に Amazon Route 53 ホストゾーンまたは DNS プロバイダーに追加する必要があります。

## Amazon CloudFront デイストリビューション

Amazon CloudFront は、ある AWS アカウントから別の AWS アカウントへのデイストリビューションの移行をサポートしていません。ただし、代替ドメイン名 (CNAME) をデイストリビューションから別のデイストリビューションへ移行することは CloudFront でサポートしています。詳細については、「[CloudFront デイストリビューション \(Knowledge Center\) の CNAME エイリアスを設定するときに CNAMEAlreadyExists エラーを解決する方法](#)」を参照してください。AWS

## AWS CodeArtifact ドメインとリポジトリ

1つの組織が複数のドメインを使用することもできますが、公開されたアーティファクトをすべて含む1つの本番ドメインを使用することをお勧めします。これにより、開発チームは組織全体でパッケージを見つけて共有できます。ドメインを所有 AWS アカウント するは、ドメインに関連付けられたリポジトリを所有するアカウントとは異なる場合があります。パッケージはリポジトリ間でコピーできますが、同じドメインに属している必要があります。詳細については、「[Copy packages between repositories](#)」(CodeArtifact ドキュメント)を参照してください。

## Amazon DynamoDB テーブル

Amazon DynamoDB テーブルを別の AWS アカウントに移行するには、次のサービスのいずれかを使用できます。

- AWS Backup
- Amazon S3 への DynamoDB インポートとエクスポート
- Amazon S3 と AWS Glue
- AWS Data Pipeline
- Amazon EMR

詳細については、「[Amazon DynamoDB テーブル AWS アカウント を別のテーブルに移行する方法 \(AWS ナレッジセンター\)](#)」を参照してください。

## Amazon EBS ボリューム

既存の Amazon Elastic Block Store (Amazon EBS) ボリュームのスナップショットを取得し、そのスナップショットをターゲットアカウントと共有して、ターゲットアカウントでボリュームのコピーを作成できます。これにより、ボリュームをあるアカウントから別のアカウントに効果的に移行します。詳細については、「[暗号化された Amazon EBS スナップショットまたはボリュームを別の \(ナレッジセンター\) と共有する方法 AWS アカウント](#)」を参照してください。AWS

## Amazon EC2 インスタンスまたは AMI

既存の Amazon Elastic Compute Cloud (Amazon EC2) インスタンスや Amazon マシンイメージ (AMI) を別の AWS アカウントに直接転送することはできません。代わりに、ソースアカウントでカ

スタム AMI を作成し、その AMI をターゲットアカウントと共有して、ターゲットアカウントの共有 AMI から新しい EC2 インスタンスを起動し、共有 AMI の登録を解除できます。

## Amazon ECR レジストリ

Amazon Elastic Container Registry (Amazon ECR) は、クロスアカウントレプリケーションとクロスリージョンレプリケーションの両方をサポートしています。ソースレジストリでレプリケーションを設定して、ターゲットレジストリでレジストリのアクセス許可ポリシーを設定します。詳細については、「[クロスアカウントレプリケーションの設定](#)」および「[ソースアカウントのルートユーザーにすべてのリポジトリのレプリケーションを許可する](#)」(Amazon ECR ドキュメント) を参照してください。

## Amazon EFS ファイルシステム

Amazon Elastic File System (Amazon EFS) は、クロスアカウントレプリケーションとクロスリージョンレプリケーションをサポートしています。ソースファイルシステムでレプリケーションを設定できます。詳細については、「[ファイルシステムのレプリケーション](#)」(Amazon EFS ドキュメント) を参照してください。

## Amazon ElastiCache (Redis OSS) クラスター

Amazon ElastiCache (Redis OSS) データベースクラスターのバックアップを使用して、別のアカウントに移行できます。詳細については、「[What are best practices for migrating my ElastiCache \(Redis OSS\) cluster](#)」(AWS Knowledge Center) を参照してください。

## AWS Elastic Beanstalk 環境

では AWS Elastic Beanstalk、[保存された設定](#) (Elastic Beanstalk ドキュメント) を使用して、環境を別のアカウントに移行できます AWS アカウント。詳細については、「[Elastic Beanstalk 環境 AWS アカウントを移行する方法 AWS アカウント](#)」(AWS ナレッジセンター) を参照してください。

## Elastic IP アドレス

Elastic IP アドレス AWS アカウント は、同じリージョンにある間で転送できます AWS リージョン。詳細については、「[Elastic IP アドレスを移管する](#)」(Amazon VPC ドキュメント) を参照してください。

## AWS Lambda レイヤー

デフォルトでは、作成した AWS Lambda レイヤーは に対してプライベートです AWS アカウント。ただし、必要に応じてレイヤーを他の と共有 AWS アカウント したり、公開したりできます。レイヤーをコピーするには、別の で再プロビジョニングします AWS アカウント。詳細については、「[レイヤー権限の設定](#)」(Lambda ドキュメント) を参照してください。

## Amazon Lightsail インスタンス

Amazon Lightsail インスタンスのスナップショットを作成し、そのスナップショットを Amazon マシンイメージ (AMI) と Amazon EBS ボリュームの暗号化されたスナップショットにエクスポートできます。詳細については、「[Amazon Lightsail スナップショットを Amazon EC2 にエクスポートする](#)」(Lightsail ドキュメント) を参照してください。デフォルトでは、スナップショットは AWS Key Management Service () で作成された AWS マネージドキーで暗号化されます AWS KMS。ただし、このタイプの KMS キーは共有できません AWS アカウント。代わりに、ターゲットアカウントから使用できるカスタマー管理キーを使用して AMI のコピーを手動で暗号化します。詳細については、「[他のアカウントのユーザーに KMS キーの使用を許可する](#) (AWS KMS ドキュメント)」を参照してください。その後、コピーした AMI をターゲットと共有 AWS アカウント し、コピーした AMI Lightsail から用の新しい EC2 インスタンスを起動できます。詳細については、「[新しいインスタンス起動ウィザードを使用してインスタンスを起動する](#)」(Amazon EC2 ドキュメント) を参照してください。

## Amazon Neptune クラスター

Amazon Neptune データベースクラスターの自動スナップショットを別の AWS アカウントにコピーできます。詳細については、「[Copying a database \(DB\) cluster snapshot](#)」(Neptune ドキュメント) を参照してください。

手動スナップショットは最大 20 の AWS アカウント と共有して、そのスナップショットから DB クラスターを直接復元することもできます。詳細については、「[Sharing a DB Cluster Snapshot](#)」(Neptune ドキュメント) を参照してください。

## Amazon OpenSearch Service ドメイン

Amazon OpenSearch Service ドメイン間でデータをコピーするには、Amazon S3 を使用してソースドメインのスナップショットを作成し、そのスナップショットを別の AWS アカウントのターゲット

ドメインに復元できます。詳細については、「[How do I restore data from an Amazon OpenSearch Service domain in another AWS アカウント](#) (AWS Knowledge Center)」を参照してください。

間にネットワーク接続がある場合は AWS アカウント、OpenSearch Service の[クラスター間レプリケーション](#) (OpenSearch Service ドキュメント) OpenSearch 機能を使用することもできます。

## Amazon RDS スナップショット

Amazon Relational Database Service (Amazon RDS) では、DB インスタンスまたはクラスターの手動スナップショットを最大 20 の AWS アカウントと共有できます。共有スナップショットから DB インスタンスまたは DB クラスターを復元できます。詳細については、「[How do I share manual Amazon RDS DB snapshots or Aurora DB cluster snapshots with another \(Knowledge Center\) AWS アカウント](#)」を参照してください。AWS

AWS Database Migration Service (AWS DMS) を使用して、異なるアカウントのデータベースインスタンス間の継続的なレプリケーションを設定することもできます。ただし、これには VPC ピアリングやトランジットゲートウェイなど、アカウント間のネットワーク接続が必要です。

## Amazon Redshift クラスター

Amazon Redshift クラスターを別のクラスターに移行するには AWS アカウント、ソースアカウントでクラスターの手動スナップショットを作成し、スナップショットをターゲットと共有してから AWS アカウント、スナップショットからクラスターを復元します。詳細については、「[Amazon Redshift でプロビジョニングされたクラスターを別のクラスター \(ナレッジセンター\) にコピーする方法 AWS アカウント](#)」を参照してください。AWS

## Amazon Route 53 のドメインとホストゾーン

Amazon Route 53 のドメインは AWS アカウント間で移管できます。詳細については、「[異なる AWS アカウントへのドメインの移管](#)」(Route 53 ドキュメント) を参照してください。

Route 53 ホストゾーンを別のホストゾーンに移行することもできます AWS アカウント。これが推奨される場合や必要な場合の詳細については、「[別の AWS アカウントにホストゾーンを移管する](#)」(Route 53 ドキュメント) を参照してください。ホストゾーンを移行する場合、ターゲットの AWS アカウントにホストゾーンを再作成します。手順については、「[別の AWS アカウントへのホストゾーンの移行](#)」(Route 53 ドキュメント) を参照してください。

## Amazon S3 バケット

Amazon Simple Storage Service (Amazon S3) の同一リージョンレプリケーションを使用して、同じ AWS リージョンにある S3 バケット間でオブジェクトをコピーできます。詳細については、「[オブジェクトのレプリケーション](#)」(Amazon S3 ドキュメント)を参照してください。次の点に注意してください。

- レプリカの所有権を、レプリケート先バケットを所有 AWS アカウント する に変更します。手順については、「[レプリカ所有者の変更](#)」(Amazon S3 ドキュメント)を参照してください。
- バケット所有者条件を更新して、ターゲットバケットの AWS アカウント ID を反映します。詳細については、「[バケット所有者条件によるバケット所有者の確認](#)」(Amazon S3 ドキュメントの)を参照してください。
- 2023 年 4 月現在、新しく作成されたバケットではバケット所有者強制設定が有効になっているため、バケットアクセスコントロールリスト (ACL) とオブジェクト ACL は無効になっています。詳細については、[Amazon S3 セキュリティの変更が近づいている](#)」(AWS ブログ記事)を参照してください。
- [S3 バッチレプリケーション](#) (Amazon S3 ドキュメント) を使用してレプリケーションが設定される前に存在していたオブジェクトをレプリケートできます。

## Amazon SageMaker AI モデル

SageMaker AI モデルは、トレーニング中に Amazon S3 バケットに保存されます。ターゲットアカウントから S3 バケットへのアクセスを許可することで、ソースアカウントに保存されているモデルをターゲットアカウントにデプロイできます。詳細については、[Amazon SageMaker AI モデルを別の \(ナレッジセンター\) にデプロイする方法 AWS アカウント](#)」を参照してください。AWS

## AWS WAF ウェブ ACLs

AWS WAF ウェブアクセスコントロールリスト (ウェブ ACLs) は、Amazon CloudFront ディストリビューション、Application Load Balancer、Amazon API Gateway REST APIs、AWS AppSync GraphQL APIs など、関連付けられているリソースと同じアカウントに存在する必要があります。を使用して AWS Firewall Manager、組織全体 AWS Organizations の AWS WAF ウェブ ACLs リージョン間で一元管理できます。詳細については、「[AWS Firewall Manager AWS WAF ポリシーの開始方法](#)」(Firewall Manager ドキュメント)を参照してください。

# マルチアカウントアーキテクチャに移行する際の請求に関する考慮事項

複数の への移行 AWS Organizations に を使用する場合は AWS アカウント、[一括請求機能](#) (AWS Organizations ドキュメント) を使用できます。この機能は、複数のアカウントの料金を示す 1 つの請求書をまとめたものです。

以下は、複数のアカウントに移行するための請求のベストプラクティスと推奨事項です。

- 過去の請求データにアクセスする必要がある場合は、組織への招待を受け入れる前に、[コストと使用状況レポート](#) (AWS Cost and Usage Report ドキュメント) を作成して、アカウントの過去の請求データを Amazon Simple Storage Service (Amazon S3) バケットにエクスポートします。組織への招待を承諾すると、アカウントの請求履歴データにアクセスできなくなります。
- 合併や買収など、2 つの組織を組み合わせる必要がある場合は、(AWS ソリューションライブラリ) の [アカウント評価 AWS Organizations](#) を使用して、各組織のリソースベースのポリシーを評価し、それらを組み合わせる前に潜在的な問題を特定できます。

## 結論

単一のアカウントから AWS アカウント 複数のアカウントへの移行は、導入戦略がないと最初は圧倒的になることがあります。マルチアカウント戦略を導入することで、シングル AWS アカウントを使用する企業が直面している次のような多くの課題に対処することができます。

- 本番稼働用データを開発用データに偏らせる – 異なるアクセス許可とアクセス許可を付与するには、異なるアクセス許可セット AWS IAM アイデンティティセンター で本番稼働用と非本番稼働用の組織単位を使用します。本番稼働用データベースにアクセスできるのは権限の高いユーザーのみとし、アクセスは期間限定で、かつ監査を受ける必要があります。
- 本番稼働環境へのデプロイが他のビジネス運用に影響を及ぼす — 複数のアカウントと複数の環境を使用することで、ステークホルダーを分離することができます。例えば非本番稼働用アカウント内に専用のセールスデモ環境を作成して、デモが行われていないときにデプロイとリリースを計画できます。
- 開発ワークロードをテストするときの本番ワークロードのパフォーマンスの低下 – 各 AWS アカウントには、各サービスを管理する独立したサービスクォータがあります。複数のアカウントを使用することで、1つの環境が別の環境に与える影響の範囲を制限できます。
- 本番稼働用コストと開発コストを区別する — 組織の一括請求 (コンソリデेटィッドビルギング) では、すべてのコストが AWS アカウント レベルでまとめられるため、財務チームは、開発環境、テスト環境、デモ環境などの非本番環境と比較して、本番稼働用コストがどれくらいかを確認することができます。タグとタグ付けポリシーを使用して、アカウント内のコストを分けることもできます。
- 機密データへのアクセスを制限する — IAM Identity Center では、特定のアカウントに関連するユーザーグループに個別のアクセスポリシーを設定できます。
- コスト管理 — マルチアカウントアーキテクチャでサービスコントロールポリシー (SCP) を使用することで、組織にとって高いコストが発生する可能性のある特定の AWS のサービス へのアクセスを禁止することができます。SCP は、特定のサービスへのすべてのアクセスを拒否したり、作成可能な Amazon Elastic Compute Cloud (Amazon EC2) インスタンスのタイプを制限するなど、サービスの使用を特定のタイプに制限したりすることができます。

## 寄稿者

本ドキュメントの寄稿者は次のとおりです。

- Justin Plock、Principal Solutions Architect、AWS (Principal Author)
- Emily Arnautovic、プリンシパルアーキテクト、AWS
- Jason DiDomenico、シニアソリューションアーキテクト、AWS
- マイケル・レイティ、セキュリティスペシャリスト・ソリューション・アーキテクト、AWS
- Jesse Lepich、シニアセキュリティスペシャリストソリューションアーキテクト、AWS
- Rodney Lester、Principal Solutions Architect、AWS
- Israel Lopez Moriano、ソリューションアーキテクト、AWS
- ジョージ・ロルトン、シニアソリューションアーキテクト、AWS
- Alex Torres、シニアソリューションアーキテクト、AWS
- Dave Walker、Principal Solutions Architect、AWS

# リソース

## AWS 規範ガイド

- [AWS セキュリティリファレンスアーキテクチャ \(AWS SRA\)](#)
- [でバックアップを保護するためのセキュリティのベストプラクティスのトップ 10 AWS](#)

## AWS ブログ投稿

- [How Setting Up IAM Users and IAM Roles Can Help Keep Your Startup Secure](#)
- [How to let builders create IAM resources while improving security and agility for your organization](#)

## AWS ホワイトペーパー

- [複数のアカウントを使用した AWS 環境の整理](#)
- [での Cloud Foundation の確立 AWS](#)
- [スケーラブルで安全なマルチ VPC AWS ネットワークインフラストラクチャの構築](#)

## AWS コードサンプル

- [Automate the setup of security services with AWS Control Tower \(GitHub\)](#)

## ドキュメント履歴

以下の表は、本ガイドの重要な変更点について説明したものです。今後の更新に関する通知を受け取る場合は、[RSS フィード](#) をサブスクライブできます。

変更	説明	日付
<a href="#">リソースコントロールポリシー</a>	リソースコントロールポリシーに関する情報を「 <a href="#">組織のセットアップ</a> 」セクションに追加しました。	2024 年 11 月 20 日
<a href="#">一元的な出力のベストプラクティス</a>	出力トラフィックを保護するための <a href="#">ベストプラクティス</a> を更新しました。	2024 年 5 月 6 日
<a href="#">組織のベストプラクティス</a>	AWS Organizationsでの組織作成に対する <a href="#">ベストプラクティス</a> を更新しました。	2023 年 12 月 4 日
<a href="#">請求に関する考慮事項</a>	「 <a href="#">請求に関する考慮事項</a> 」セクションを追加しました。	2023 年 9 月 20 日
<a href="#">リソースの移行、アプリケーションの接続、Amazon VPC Lattice</a>	また、「 <a href="#">Resource migration</a> 」と「 <a href="#">アプリケーションの接続</a> 」のセクションも追加しました。また、新しい AWS のサービスである Amazon Virtual Private Cloud (Amazon VPC) Lattice に関する情報も追加しました。	2023 年 4 月 27 日
<a href="#">アカウント履歴と ABAC</a>	「 <a href="#">ランディングゾーンの作成</a> 」セクションを改訂し、新しい AWS アカウントの使用履歴を確認してランディングゾーンに追加できるようにする方法についての情報を	2023 年 1 月 6 日

	<p>追加しました AWS Control Tower。また、「<a href="#">Add initial users</a>」セクションを改訂し、属性ベースのアクセス制御 (ABAC) を使用して、外部の SAML ベースの IdP から AWS IAM アイデンティティセンターに認証方法を渡す方法についての情報を追加しました。</p>	
<a href="#">エグレストラフィックネットワーク</a>	<p>Amazon Route 53 Resolver DNS Firewall を使用して送信トラフィックを特定のドメイン名に制限する情報を追加するために、「<a href="#">集中型送信</a>」セクションを改訂しました。</p>	2022 年 10 月 13 日
<a href="#">エグレストラフィックのセキュリティ</a>	<p>「<a href="#">Best practices for securing egress traffic</a>」を追加しました。</p>	2022 年 10 月 6 日
<a href="#">アクセス許可の境界</a>	<p><a href="#">アクセス許可の境界</a>の定義を改善し、「リソース」セクションにこのトピックに関する詳細情報のリンクを追加しました。</p>	2022 年 9 月 22 日
<a href="#">初版発行</a>	—	2022 年 9 月 6 日

# AWS 規範ガイドの用語集

以下は、AWS 規範ガイドによって提供される戦略、ガイド、パターンで一般的に使用される用語です。エントリを提案するには、用語集の最後のフィードバックの提供リンクを使用します。

## 数字

### 7 Rs

アプリケーションをクラウドに移行するための 7 つの一般的な移行戦略。これらの戦略は、ガートナーが 2011 年に特定した 5 Rs に基づいて構築され、以下で構成されています。

- リファクタリング/アーキテクチャの再設計 — クラウドネイティブ特徴を最大限に活用して、俊敏性、パフォーマンス、スケーラビリティを向上させ、アプリケーションを移動させ、アーキテクチャを変更します。これには、通常、オペレーティングシステムとデータベースの移植が含まれます。例: オンプレミスの Oracle データベースを Amazon Aurora PostgreSQL 互換エディションに移行する。
- リプラットフォーム (リフトアンドリシェイプ) — アプリケーションをクラウドに移行し、クラウド機能を活用するための最適化レベルを導入します。例: お客様のオンプレミスの Oracle データベースを AWS クラウドの Oracle 用の Amazon Relational Database Service (Amazon RDS) に移行する。
- 再購入 (ドロップアンドショップ) — 通常、従来のライセンスから SaaS モデルに移行して、別の製品に切り替えます。例: 顧客関係管理 (CRM) システムを Salesforce.com に移行する。
- リホスト (リフトアンドシフト) — クラウド機能を活用するための変更を加えずに、アプリケーションをクラウドに移行します。例: お客様のオンプレミスの Oracle データベースを AWS クラウドの EC2 インスタンス上の Oracle に移行する。
- 再配置 (ハイパーバイザーレベルのリフトアンドシフト) — 新しいハードウェアを購入したり、アプリケーションを書き換えたり、既存の運用を変更したりすることなく、インフラストラクチャをクラウドに移行できます。オンプレミスプラットフォームから同じプラットフォームのクラウドサービスにサーバーを移行します。例: Microsoft Hyper-V アプリケーションをに移行します AWS。
- 保持 (再アクセス) — アプリケーションをお客様のソース環境で保持します。これには、主要なリファクタリングを必要とするアプリケーションや、お客様がその作業を後日まで延期したいアプリケーション、およびそれらを行き移るためのビジネス上の正当性がないため、お客様が保持するレガシーアプリケーションなどがあります。
- 廃止 — お客様のソース環境で不要になったアプリケーションを停止または削除します。

# A

## ABAC

[「属性ベースのアクセス制御」](#)をご覧ください。

## 抽象化されたサービス

[「マネージドユーザー」](#)をご覧ください。

## ACID

[「原子性、一貫性、分離性、耐久性 \(ACID\)」](#)をご覧ください。

## アクティブ/アクティブ移行

(双方向レプリケーションツールまたは二重書き込み操作を使用して) ソースデータベースとターゲットデータベースを同期させ、移行中に両方のデータベースが接続アプリケーションからのトランザクションを処理するデータベース移行方法。この方法では、1 回限りのカットオーバーの必要がなく、管理された小規模なバッチで移行できます。[アクティブ/パッシブ移行](#)よりも柔軟な方法ですが、さらに多くの作業が必要となります。

## アクティブ/パッシブ移行

ソースデータベースとターゲットデータベースを同期させながら、データがターゲットデータベースにレプリケートされている間、接続しているアプリケーションからのトランザクションをソースデータベースのみで処理するデータベース移行方法。移行中、ターゲットデータベースはトランザクションを受け付けません。

## 集計関数

複数行に処理を行い、グループ全体を対象に単一の戻り値を計算する SQL 関数。集計関数の例としては、SUM や MAX などがあります。

## AI

[「人工知能」](#)をご覧ください。

## AIOps

[「AI オペレーション」](#)をご覧ください。

## 匿名化

データセット内の個人情報を完全に削除するプロセス。匿名化は個人のプライバシー保護に役立ちます。匿名化されたデータは、もはや個人データとは見なされません。

## アンチパターン

繰り返し起こる問題に対して頻繁に用いられる解決策で、その解決策が逆効果であったり、効果がなかったり、代替案よりも効果が低かったりするもの。

### アプリケーション制御

マルウェアからシステムを保護するために、承認されたアプリケーションのみを使用できるようにするセキュリティアプローチ。

### アプリケーションポートフォリオ

アプリケーションの構築と維持にかかるコスト、およびそのビジネス価値を含む、組織が使用する各アプリケーションに関する詳細情報の集まり。この情報は、[ポートフォリオの検出と分析プロセス](#)の重要な要素であり、移行、モダナイズ、最適化するアプリケーションを特定し、優先順位を付けるのに役立ちます。

### 人工知能 (AI)

コンピューティングテクノロジーを使用し、学習、問題の解決、パターンの認識など、通常は人間に関連づけられる認知機能の実行に特化したコンピュータサイエンスの分野。詳細については、「[人工知能 \(AI\) とは何ですか?](#)」をご覧ください。

### AI オペレーション (AIOps)

機械学習技術を使用して運用上の問題を解決し、運用上のインシデントと人の介入を減らし、サービス品質を向上させるプロセス。AWS 移行戦略での AIOps の使用方法については、[オペレーション統合ガイド](#)を参照してください。

### 非対称暗号化

暗号化用のパブリックキーと復号用のプライベートキーから成る 1 組のキーを使用した、暗号化のアルゴリズム。パブリックキーは復号には使用されないため共有しても問題ありませんが、プライベートキーの利用は厳しく制限する必要があります。

### 原子性、一貫性、分離性、耐久性 (ACID)

エラー、停電、その他の問題が発生した場合でも、データベースのデータ有効性と運用上の信頼性を保証する一連のソフトウェアプロパティ。

### 属性ベースのアクセス制御 (ABAC)

部署、役職、チーム名など、ユーザーの属性に基づいてアクセス許可をきめ細かく設定する方法。詳細については、AWS Identity and Access Management (IAM) ドキュメントの「[の ABAC AWS](#)」を参照してください。

## 信頼できるデータソース

最も信頼性のある情報源とされるデータのプライマリーバージョンを保存する場所。匿名化、編集、仮名化など、データを処理または変更する目的で、信頼できるデータソースから他の場所にデータをコピーすることができます。

## アベイラビリティゾーン (AZ)

他のアベイラビリティゾーンの障害から AWS リージョン 隔離され、同じリージョン内の他のアベイラビリティゾーンへの低コストで低レイテンシーのネットワーク接続を提供する 内の別の場所。

## AWS クラウド導入フレームワーク (AWS CAF)

組織がクラウドへの移行を成功させるための効率的で効果的な計画を立て AWS するための、のガイドラインとベストプラクティスのフレームワークです。AWS CAF は、ビジネス、人材、ガバナンス、プラットフォーム、セキュリティ、運用という 6 つの重点分野にガイダンスを整理しています。ビジネス、人材、ガバナンスの観点では、ビジネススキルとプロセスに重点を置き、プラットフォーム、セキュリティ、オペレーションの視点は技術的なスキルとプロセスに焦点を当てています。例えば、人材の観点では、人事 (HR)、人材派遣機能、および人材管理を扱うステークホルダーを対象としています。この観点から、AWS CAF は、クラウド導入を成功させるための組織の準備に役立つ人材開発、トレーニング、コミュニケーションに関するガイダンスを提供します。詳細については、[AWS CAF ウェブサイト](#)と [AWS CAF のホワイトペーパー](#) を参照してください。

## AWS ワークロード認定フレームワーク (AWS WQF)

データベース移行ワークロードを評価し、移行戦略を推奨し、作業見積もりを提供するツール。AWS WQF は AWS Schema Conversion Tool (AWS SCT) に含まれています。データベーススキーマとコードオブジェクト、アプリケーションコード、依存関係、およびパフォーマンス特性を分析し、評価レポートを提供します。

# B

## 不正なボット

個人や組織に混乱や損害を与えることを目的とした [ボット](#)。

## BCP

「[ビジネス継続性計画 \(BCP\)](#)」をご覧ください。

## 動作グラフ

リソースの動作とインタラクションを経時的に示した、一元的なインタラクティブビュー。Amazon Detective の動作グラフを使用すると、失敗したログオンの試行、不審な API 呼び出し、その他同様のアクションを調べることができます。詳細については、Detective ドキュメントの「[動作グラフのデータ](#)」を参照してください。

## ビッグエンディアンシステム

最上位バイトを最初に格納するシステム。「[エンディアン性](#)」もご覧ください。

## 二項分類

バイナリ結果 (2 つの可能なクラスのうちの一つ) を予測するプロセス。例えば、お客様の機械学習モデルで「この E メールはスパムですか、それともスパムではありませんか」などの問題を予測する必要があるかもしれません。または「この製品は書籍ですか、車ですか」などの問題を予測する必要があるかもしれません。

## ブルームフィルター

要素がセットのメンバーであるかどうかをテストするために使用される、確率的でメモリ効率の高いデータ構造。

## ブルー/グリーンデプロイ

それぞれが独立しているが、同一の環境を 2 つ作成するデプロイ戦略。現在のアプリケーションバージョンを 1 つの環境 (ブルー) で実行し、新しいアプリケーションバージョンを別の環境 (グリーン) で実行します。この戦略は、最小限の影響で迅速にロールバックするのに役立ちます。

## ボット

インターネット経由で自動タスクを実行し、人間のアクティビティややり取りをシミュレートするソフトウェアアプリケーション。インターネット上の情報のインデックスを作成するウェブクローラーなど、一部のボットは有用または有益です。悪質なボットと呼ばれる他のボットの中には、個人や組織を混乱させたり、損害を与えたりすることを意図したものもあります。

## ボットネット

[マルウェア](#)に感染しており、ボットハーダーまたはボットオペレーターと呼ばれる単一の当事者によって制御されている[ボット](#)のネットワーク。ボットネットは、ボットとその影響力を拡大する仕組みとして、非常によく知られています。

## ブランチ

コードリポジトリに含まれる領域。リポジトリに最初に作成するブランチは、メインブランチといます。既存のブランチから新しいブランチを作成し、その新しいブランチで機能を開発した

り、バグを修正したりできます。機能を構築するために作成するブランチは、通常、機能ブランチと呼ばれます。機能をリリースする準備ができたなら、機能ブランチをメインブランチに統合します。詳細については、「[ブランチの概要](#)」(GitHub ドキュメント)を参照してください。

## ブレイクグラスアクセス

例外的な状況では、承認されたプロセスを通じて、ユーザーが AWS アカウント 通常アクセス許可を持たないにすばやくアクセスできるようにします。詳細については、AWS Well-Architected ガイドの「[ブレイクグラス手順の実装](#)」インジケータを参照してください。

## ブラウフィールド戦略

環境の既存インフラストラクチャ。システムアーキテクチャにブラウフィールド戦略を導入する場合、現在のシステムとインフラストラクチャの制約に基づいてアーキテクチャを設計します。既存のインフラストラクチャを拡張している場合は、ブラウフィールド戦略と[グリーンフィールド](#)戦略を融合させることもできます。

## バッファキャッシュ

アクセス頻度が最も高いデータが保存されるメモリ領域。

## ビジネス能力

価値を生み出すためにビジネスが行うこと (営業、カスタマーサービス、マーケティングなど)。マイクロサービスのアーキテクチャと開発の決定は、ビジネス能力によって推進できます。詳細については、[AWSでのコンテナ化されたマイクロサービスの実行](#)ホワイトペーパーの「[ビジネス機能を中心に組織化](#)」セクションを参照してください。

## ビジネス継続性計画 (BCP)

大規模移行など、中断を伴うイベントが運用に与える潜在的な影響に対処し、ビジネスを迅速に再開できるようにする計画。

# C

## CAF

「[AWS クラウド導入フレームワーク](#)」を参照してください

## カナリアデプロイ

エンドユーザーへのバージョンリリースを、時間をかけて段階的に行うこと。確信が持てたら新規バージョンをデプロイして、現在のバージョン全体を置き換えます。

## CCoE

「[Cloud Center of Excellence](#)」を参照してください。

## CDC

「[変更データキャプチャ](#)」を参照してください。

### 変更データキャプチャ (CDC)

データソース (データベーステーブルなど) の変更を追跡し、その変更に関するメタデータを記録するプロセス。CDC は、ターゲットシステムでの変更を監査またはレプリケートして同期を維持するなど、さまざまな目的に使用できます。

## カオスエンジニアリング

障害や破壊的なイベントを意図的に導入して、システムの耐障害性をテストすること。[AWS Fault Injection Service \(AWS FIS\)](#) を使用して、AWS ワークロードにストレスを与え、その応答を評価する実験を実行できます。

## CI/CD

「[継続的インテグレーションと継続的デリバリー](#)」を参照してください。

## 分類

予測を生成するのに役立つ分類プロセス。分類問題の機械学習モデルは、離散値を予測します。離散値は、常に互いに区別されます。例えば、モデルがイメージ内に車があるかどうかを評価する必要がある場合があります。

## クライアント側の暗号化

ターゲットがデータ AWS のサービスを受信する前のローカルでのデータの暗号化。

## Cloud Center of Excellence (CCoE)

クラウドのベストプラクティスの作成、リソースの移動、移行のタイムラインの確立、大規模変革を通じて組織をリードするなど、組織全体のクラウド導入の取り組みを推進する学際的なチーム。詳細については、AWS クラウド エンタープライズ戦略ブログの [CCoE 投稿](#) を参照してください。

## クラウドコンピューティング

リモートデータストレージと IoT デバイス管理に通常使用されるクラウドテクノロジー。クラウドコンピューティングは、一般的に、[エッジコンピューティング](#)に接続されています。

## クラウド運用モデル

IT 組織において、1 つ以上のクラウド環境を構築、成熟、最適化するために使用される運用モデル。詳細については、「[クラウド運用モデルの構築](#)」を参照してください。

### 導入のクラウドステージ

組織が、AWS クラウドへの移行時に通常実行する 4 つの段階。

- プロジェクト — 概念実証と学習を目的として、クラウド関連のプロジェクトをいくつか実行する
- 基礎固め — お客様のクラウドの導入を拡大するための基礎的な投資 (ランディングゾーン の作成、CCoE の定義、運用モデルの確立など)
- 移行 — 個々のアプリケーションの移行
- 再発明 — 製品とサービスの最適化、クラウドでのイノベーション

これらのステージは、AWS クラウド エンタープライズ戦略ブログのブログ記事「[クラウドファーストへのジャーニー](#)」と「[導入のステージ](#)」で Stephen Orban によって定義されました。移行戦略との関連性については、AWS「[移行準備ガイド](#)」を参照してください。

### CMDB

「[構成管理データベース \(CMDB\)](#)」を参照してください。

### コードリポジトリ

ソースコードやその他の資産 (ドキュメント、サンプル、スクリプトなど) が保存され、バージョン管理プロセスを通じて更新される場所。一般的なクラウドリポジトリには、GitHub や Bitbucket Cloud があります。コードの各バージョンはブランチと呼ばれます。マイクロサービスの構造では、各リポジトリは 1 つの機能専用です。1 つの CI/CD パイプラインで複数のリポジトリを使用できます。

### コールドキャッシュ

空である、または、かなり空きがある、もしくは、古いデータや無関係なデータが含まれているバッファキャッシュ。データベースインスタンスはメインメモリまたはディスクから読み取る必要があり、バッファキャッシュから読み取るよりも時間がかかるため、パフォーマンスに影響します。

### コールドデータ

めったにアクセスされず、通常は過去のデータです。この種類のデータをクエリする場合、通常は低速なクエリでも問題ありません。このデータを低パフォーマンスで安価なストレージ階層またはクラスに移動すると、コストを削減することができます。

## コンピュータビジョン (CV)

機械学習を使用してデジタルイメージやビデオといった、ビジュアル形式の情報を分析および抽出する [AI](#) の分野。例えば、Amazon SageMaker AI では、CV 用の画像処理アルゴリズムを利用できます。

## 設定ドリフト

ワークロードにおいて、設定が想定した状態から変化すること。これによって、ワークロードが非準拠になる可能性があります。この状態は、徐々に生じ、意図的なものではありません。

## 構成管理データベース (CMDB)

データベースとその IT 環境 (ハードウェアとソフトウェアの両方のコンポーネントとその設定を含む) に関する情報を保存、管理するリポジトリ。通常、CMDB のデータは、移行のポートフォリオの検出と分析の段階で使用します。

## コンフォーマンスパック

コンプライアンスチェックとセキュリティチェックをカスタマイズするためにアセンブルできる AWS Config ルールと修復アクションのコレクション。YAML テンプレートを使用して、コンフォーマンスパックを AWS アカウント および リージョンの単一のエンティティとしてデプロイすることも、組織全体にデプロイすることもできます。詳細については、AWS Config ドキュメントの「[コンフォーマンスパック](#)」を参照してください。

## 継続的インテグレーションと継続的デリバリー (CI/CD)

ソフトウェアリリースプロセスのソース、ビルド、テスト、ステージング、本番の各ステージを自動化するプロセス。CI/CD は一般的にパイプラインと呼ばれます。プロセスの自動化、生産性の向上、コード品質の向上、配信の加速化を可能にします。詳細については、「[継続的デリバリーの利点](#)」を参照してください。CD は継続的デプロイ (Continuous Deployment) の略語でもあります。詳細については「[継続的デリバリーと継続的なデプロイ](#)」を参照してください。

## CV

[「コンピュータビジョン」](#) を参照してください。

## D

### 保管中のデータ

ストレージ内にあるデータなど、常に自社のネットワーク内にあるデータ。

## データ分類

ネットワーク内のデータを重要度と機密性に基づいて識別、分類するプロセス。データに適した保護および保持のコントロールを判断する際に役立つため、あらゆるサイバーセキュリティのリスク管理戦略において重要な要素です。データ分類は、AWS Well-Architected フレームワークのセキュリティの柱のコンポーネントです。詳細については、「[データ分類](#)」を参照してください。

## データドリフト

実稼働データと ML モデルのトレーニングに使用されたデータとの間に有意な差異が生じたり、入力データが時間の経過と共に有意に変化したりすることです。データドリフトは、ML モデル予測の全体的な品質、精度、公平性を低下させる可能性があります。

## 転送中のデータ

ネットワーク内 (ネットワークリソース間など) を活発に移動するデータ。

## データメッシュ

非一元的で分散型のデータ所有権を持つとともに、一元的な管理およびガバナンスを行えるアーキテクチャフレームワーク。

## データ最小化

厳密に必要なデータのみを収集し、処理するという原則。でデータ最小化を実践 AWS クラウドすることで、プライバシーリスク、コスト、分析のカーボンフットプリントを削減できます。

## データ境界

AWS 環境内の一連の予防ガードレール。信頼された ID のみが、期待されるネットワークから信頼されたリソースにアクセスできるようにします。詳細については、「[でのデータ境界の構築 AWS](#)」を参照してください。

## データの前処理

raw データをお客様の機械学習モデルで簡単に解析できる形式に変換すること。データの前処理とは、特定の列または行を削除して、欠落している、矛盾している、または重複する値に対処することを意味します。

## データ出所

データの生成、送信、保存の方法など、データのライフサイクル全体を通じてデータの出所と履歴を追跡するプロセス。

## データ件名

データを収集、処理している個人。

## データウェアハウス

分析などのビジネスインテリジェンスをサポートするデータ管理システム。データウェアハウスには、一般的に、大量の履歴データが含まれており、多くの場合、それらはクエリや分析に使用されます。

## データベース定義言語 (DDL)

データベース内のテーブルやオブジェクトの構造を作成または変更するためのステートメントまたはコマンド。

## データベース操作言語 (DML)

データベース内の情報を変更 (挿入、更新、削除) するためのステートメントまたはコマンド。

## DDL

「[データベース定義言語](#)」を参照してください。

## ディープアンサンブル

予測のために複数の深層学習モデルを組み合わせます。ディープアンサンブルを使用して、より正確な予測を取得したり、予測の不確実性を推定したりできます。

## 深層学習

人工ニューラルネットワークの複数層を使用して、入力データと対象のターゲット変数の間のマッピングを識別する機械学習サブフィールド。

## 多層防御

一連のセキュリティメカニズムとコントロールをコンピュータネットワーク全体に層状に重ねて、ネットワークとその内部にあるデータの機密性、整合性、可用性を保護する情報セキュリティの手法。この戦略をに採用するときは AWS、リソースの保護に役立つように、AWS Organizations 構造の異なるレイヤーに複数のコントロールを追加します。たとえば、多層防御アプローチでは、多要素認証、ネットワークセグメンテーション、暗号化を組み合わせることができます。

## 委任管理者

では AWS Organizations、互換性のあるサービスが AWS メンバーアカウントを登録して組織のアカウントを管理し、そのサービスのアクセス許可を管理できます。このアカウントを、そのサービスの委任管理者と呼びます。詳細、および互換性のあるサービスの一覧は、AWS

Organizations ドキュメントの「[AWS Organizationsで利用できるサービス](#)」を参照してください。

## トラブルシューティング

アプリケーション、新機能、コードの修正をターゲットの環境で利用できるようにするプロセス。デプロイでは、コードベースに変更を施した後、アプリケーションの環境でそのコードベースを構築して実行します。

## 開発環境

「[環境](#)」を参照してください。

## 検出管理

イベントが発生したときに、検出、ログ記録、警告を行うように設計されたセキュリティコントロール。これらのコントロールは副次的な防衛手段であり、実行中の予防的コントロールをすり抜けたセキュリティイベントをユーザーに警告します。詳細については、「AWSでのセキュリティコントロールの実装」の「[検出的コントロール](#)」を参照してください。

## 開発バリューストリームマッピング (DVSM)

ソフトウェア開発ライフサイクルのスピードと品質に悪影響を及ぼす制約を特定し、優先順位を付けるために使用されるプロセス。DVSM は、もともとリーンマニファクチャリング・プラクティスのために設計されたバリューストリームマッピング・プロセスを拡張したものです。ソフトウェア開発プロセスを通じて価値を創造し、動かすために必要なステップとチームに焦点を当てています。

## デジタルツイン

建物、工場、産業機器、生産ラインなど、現実世界のシステムを仮想的に表現したものです。デジタルツインは、予知保全、リモートモニタリング、生産最適化をサポートします。

## ディメンションテーブル

[スタースキーマ](#)において、ファクトテーブルの定量データに関するデータ属性が含まれる小さいテーブル。ディメンションテーブルの属性は、通常、テキストフィールド、またはテキストのように扱える個別の数値で示されます。これらの属性は、一般的に、クエリの制約、フィルタリング、結果セットのラベル付けに使用されます。

## デザスタ

ワークロードまたはシステムが、導入されている主要な場所でのビジネス目標の達成を妨げるイベント。これらのイベントは、自然災害、技術的障害、または意図しない設定ミスやマルウェア攻撃などの人間の行動の結果である場合があります。

## ディザスタリカバリ (DR)

[ディザスタ](#)によるダウンタイムとデータ損失を最小限に抑えるための戦略とプロセス。詳細については、AWS Well-Architected フレームワークの「[でのワークロードのディザスタリカバリ](#)」[AWS: クラウドでのリカバリ](#)」を参照してください。

## DML

「[データベース操作言語](#)」を参照してください。

## ドメイン駆動型設計

各コンポーネントが提供している変化を続けるドメイン、またはコアビジネス目標にコンポーネントを接続して、複雑なソフトウェアシステムを開発するアプローチ。この概念は、エリック・エヴァンスの著書、Domain-Driven Design: Tackling Complexity in the Heart of Software (ドメイン駆動設計:ソフトウェアの中心における複雑さへの取り組み) で紹介されています (ポストン: Addison-Wesley Professional、2003)。strangler fig パターンでドメイン駆動型設計を使用する方法の詳細については、「[コンテナと Amazon API Gateway を使用して、従来の Microsoft ASP.NET \(ASMX\) ウェブサービスを段階的にモダナイズ](#)」を参照してください。

## DR

「[ディザスタリカバリ](#)」を参照してください。

## ドリフト検出

ベースライン設定からの偏差を追跡します。たとえば、AWS CloudFormation を使用して[システムリソースのドリフトを検出](#)したり、を使用して AWS Control Tower、ガバナンス要件への準拠に影響する[ランディングゾーンの変更を検出](#)したりできます。

## DVSM

「[開発バリューSTREAMマッピング](#)」を参照してください。

## E

### EDA

「[探索的データ分析](#)」を参照してください。

### EDI

「[電子データ交換](#)」を参照してください。

## エッジコンピューティング

IoT ネットワークのエッジにあるスマートデバイスの計算能力を高めるテクノロジー。[クラウドコンピューティング](#)と比較すると、エッジコンピューティングは通信レイテンシーを短縮し、応答時間を改善できます。

## 電子データ交換 (EDI)

組織間で行う、ビジネスドキュメントの自動交換。詳細については、[「電子データ交換とは」](#)を参照してください。

## 暗号化

人間が読み取り可能なプレーンテキストデータを暗号文に変換するコンピューティング処理。

## 暗号化キー

暗号化アルゴリズムが生成した、ランダム化されたビットからなる暗号文字列。キーの長さは決まっておらず、各キーは予測できないように、一意になるように設計されています。

## エンディアン

コンピュータメモリにバイトが格納される順序。ビッグエンディアンシステムでは、最上位バイトが最初に格納されます。リトルエンディアンシステムでは、最下位バイトが最初に格納されます。

## エンドポイント

[「サービスエンドポイント」](#)を参照してください。

## エンドポイントサービス

仮想プライベートクラウド (VPC) 内でホストして、他のユーザーと共有できるサービス。を使用してエンドポイントサービスを作成し AWS PrivateLink、他の AWS アカウント または AWS Identity and Access Management (IAM) プリンシパルにアクセス許可を付与できます。これらのアカウントまたはプリンシパルは、インターフェイス VPC エンドポイントを作成することで、エンドポイントサービスにプライベートに接続できます。詳細については、Amazon Virtual Private Cloud (Amazon VPC) ドキュメントの [「エンドポイントサービスを作成する」](#)を参照してください。

## エンタープライズリソースプランニング (ERP)

エンタープライズの主要なビジネスプロセス (会計、[MES](#)、プロジェクト管理など) を自動化および管理するシステム。

## エンベロープ暗号化

暗号化キーを、別の暗号化キーを使用して暗号化するプロセス。詳細については、AWS Key Management Service (AWS KMS) ドキュメントの「[エンベロープ暗号化](#)」を参照してください。

### 環境

実行中のアプリケーションのインスタンス。クラウドコンピューティングにおける一般的な環境の種類は以下のとおりです。

- 開発環境 — アプリケーションのメンテナンスを担当するコアチームのみが利用できる、実行中のアプリケーションのインスタンス。開発環境は、上位の環境に昇格させる変更をテストするときに使用します。このタイプの環境は、テスト環境と呼ばれることもあります。
- 下位環境 — 初期ビルドやテストに使用される環境など、アプリケーションのすべての開発環境。
- 本番環境 — エンドユーザーがアクセスできる、実行中のアプリケーションのインスタンス。CI/CD パイプラインでは、本番環境が最後のデプロイ環境になります。
- 上位環境 — コア開発チーム以外のユーザーがアクセスできるすべての環境。これには、本番環境、本番前環境、ユーザー承認テスト環境などが含まれます。

### エピック

アジャイル方法論で、お客様の作業の整理と優先順位付けに役立つ機能カテゴリ。エピックでは、要件と実装タスクの概要についてハイレベルな説明を提供します。例えば、AWS CAF セキュリティエピックには、ID とアクセスの管理、検出コントロール、インフラストラクチャセキュリティ、データ保護、インシデント対応が含まれます。AWS 移行戦略のエピックの詳細については、[プログラム実装ガイド](#)を参照してください。

### ERP

「[エンタープライズリソース計画](#)」を参照してください。

### 探索的データ分析 (EDA)

データセットを分析してその主な特性を理解するプロセス。お客様は、データを収集または集計してから、パターンの検出、異常の検出、および前提条件のチェックのための初期調査を実行します。EDA は、統計の概要を計算し、データの可視化を作成することによって実行されます。

## F

### ファクトテーブル

[スタースキーマ](#)の中央にあるテーブル。ビジネスオペレーションに関する定量的データが保存されます。一般的に、ファクトテーブルは、2種類の列で構成されます。1つは測定値が含まれる列、もう1つはディメンションテーブルへの外部キーが含まれる列です。

### フェイルファスト

開発ライフサイクルを短縮するために、頻繁かつ段階的にテストを行う哲学であり、アジャイルアプローチでは、この考え方がきわめて重要です。

### 障害分離境界

では AWS クラウド、障害の影響を制限し、ワークロードの耐障害性を高めるのに役立つアベイラビリティゾーン AWS リージョン、コントロールプレーン、データプレーンなどの境界。詳細については、「[AWS 障害分離境界](#)」を参照してください。

### 機能ブランチ

「[ブランチ](#)」を参照してください。

### 特徴量

お客様が予測に使用する入力データ。例えば、製造コンテキストでは、特徴量は製造ラインから定期的にキャプチャされるイメージの可能性もあります。

### 特徴量重要度

モデルの予測に対する特徴量の重要性。これは通常、Shapley Additive Deskonations (SHAP) や積分勾配など、さまざまな手法で計算できる数値スコアで表されます。詳細については、「[を使用した機械学習モデルの解釈可能性 AWS](#)」を参照してください。

### 機能変換

追加のソースによるデータのエンリッチ化、値のスケーリング、単一のデータフィールドからの複数の情報セットの抽出など、機械学習プロセスのデータを最適化すること。これにより、機械学習モデルはデータの恩恵を受けることができます。例えば、「2021-05-27 00:15:37」の日付を「2021年」、「5月」、「木」、「15」に分解すると、学習アルゴリズムがさまざまなデータコンポーネントに関連する微妙に異なるパターンを学習するのに役立ちます。

### 数ショットプロンプト

[LLM](#) に、タスクと望ましい出力を示す例を少数提示した後に、類似のタスクを実行させること。この手法は、プロンプトに記述された例(ショット)からモデルが学習する「インコンテキスト学

習」の一種です。数ショットプロンプトは、特定のフォーマット、推論、専門知識が必要なタスクに効果的です。「[ゼロショットプロンプト](#)」も参照してください。

## FGAC

「[きめ細かなアクセス制御](#)」を参照してください。

### きめ細かなアクセス制御 (FGAC)

複数の条件を使用してアクセス要求を許可または拒否すること。

## フラッシュカット移行

[変更データのキャプチャ](#)による継続的なデータ複製を利用して、段階的なアプローチではなく、可能な限り短時間でデータを移行するデータベース移行方法。目的はダウンタイムを最小限に抑えることです。

## FM

「[基盤モデル](#)」を参照してください。

### 基盤モデル (FM)

大規模な深層学習ニューラルネットワークであり、一般化およびラベル付けされていないデータからなる大規模データセットでトレーニングされています。FMにより、言語理解、テキストおよび画像生成、自然言語での会話といった、一般的な各種タスクを実行できます。詳細については、「[基盤モデルとは何ですか?](#)」を参照してください。

## G

### 生成 AI

[AI](#) モデルのサブセット。大量のデータでトレーニングされており、シンプルなテキストプロンプトを使用して、画像、動画、テキスト、オーディオなどの新しいコンテンツやアーティファクトを作成できます。詳細については、「[生成 AI とは何ですか?](#)」を参照してください。

### ジオブロッキング

「[地理的制限](#)」を参照してください。

### 地理的制限 (ジオブロッキング)

特定の国のユーザーがコンテンツ配信にアクセスできないようにするための、Amazon CloudFront のオプション。アクセスを許可する国と禁止する国は、許可リストまたは禁止リスト

を使って指定します。詳細については、CloudFront ドキュメントの「[コンテンツの地理的ディストリビューションの制限](#)」を参照してください。

## Gitflow ワークフロー

下位環境と上位環境が、ソースコードリポジトリでそれぞれ異なるブランチを使用する方法。Gitflow ワークフローは古いと見なされている方法であり、[トランクベースのワークフロー](#)は推奨されている新しい方法です。

## ゴールデンイメージ

システムまたはソフトウェアのスナップショットであり、システムまたはソフトウェアの新規インスタンスをデプロイするテンプレートとして使用されます。製造の例で言えば、ゴールデンイメージを使用すると、複数のデバイスにソフトウェアをプロビジョニングして、デバイス製造オペレーションの速度、スケーラビリティ、生産性を向上させることができます。

## グリーンフィールド戦略

新しい環境に既存のインフラストラクチャが存在しないこと。システムアーキテクチャにグリーンフィールド戦略を導入する場合、既存のインフラストラクチャ (別名 [ブラウンフィールド](#)) との互換性の制約を受けることなく、あらゆる新しいテクノロジーを選択できます。既存のインフラストラクチャを拡張している場合は、ブラウンフィールド戦略とグリーンフィールド戦略を融合させることもできます。

## ガードレール

組織単位 (OU) 全般のリソース、ポリシー、コンプライアンスを管理するのに役立つ概略的なルール。予防ガードレールは、コンプライアンス基準に一致するようにポリシーを実施します。これらは、サービスコントロールポリシーと IAM アクセス許可の境界を使用して実装されます。検出ガードレールは、ポリシー違反やコンプライアンス上の問題を検出し、修復のためのアラートを発信します。これらは AWS Config、AWS Security Hub CSPM、Amazon GuardDuty、AWS Trusted Advisor Amazon Inspector、およびカスタム AWS Lambda チェックを使用して実装されます。

# H

## HA

「[高可用性](#)」を参照してください。

## 異種混在データベースの移行

別のデータベースエンジンを使用するターゲットデータベースへお客様の出典データベースの移行 (例えば、Oracle から Amazon Aurora)。異種間移行は通常、アーキテクチャの再設計作業の一部であり、スキーマの変換は複雑なタスクになる可能性があります。[AWS は、スキーマの変換に役立つ AWS SCTを提供します。](#)

### 高可用性 (HA)

課題や災害が発生した場合に、介入なしにワークロードを継続的に運用できること。HA システムは、自動的にフェイルオーバーし、一貫して高品質のパフォーマンスを提供し、パフォーマンスへの影響を最小限に抑えながらさまざまな負荷や障害を処理するように設計されています。

### ヒストリアンのモダナイゼーション

製造業のニーズによりよく応えるために、オペレーションテクノロジー (OT) システムをモダナイズし、アップグレードするためのアプローチ。ヒストリアンは、工場内のさまざまなソースからデータを収集して保存するために使用されるデータベースの一種です。

### ホールドアウトデータ

[機械学習](#) モデルのトレーニング用データセットから保留される、ラベル付き履歴データの一部。ホールドアウトデータを使用すると、モデル予測をホールドアウトデータと比較して、モデルのパフォーマンスを評価できます。

### 同種データベースの移行

お客様の出典データベースを、同じデータベースエンジンを共有するターゲットデータベース (Microsoft SQL Server から Amazon RDS for SQL Server など) に移行する。同種間移行は、通常、リホストまたはリプラットフォーム化の作業の一部です。ネイティブデータベースユーティリティを使用して、スキーマを移行できます。

### ホットデータ

リアルタイムデータや最近の翻訳データなど、頻繁にアクセスされるデータ。通常、このデータには高速なクエリ応答を提供する高性能なストレージ階層またはクラスが必要です。

### ホットフィックス

本番環境の重大な問題を修正するために緊急で配布されるプログラム。緊急性が高いため、通常の DevOps のリリースワークフローからは外れた形で実施されます。

## ハイパーケア期間

カットオーバー直後、移行したアプリケーションを移行チームがクラウドで管理、監視して問題に対処する期間。通常、この期間は 1~4 日です。ハイパーケア期間が終了すると、アプリケーションに対する責任は一般的に移行チームからクラウドオペレーションチームに移ります。

## I

### laC

「[Infrastructure as Code](#)」を参照してください。

### ID ベースのポリシー

AWS クラウド 環境内のアクセス許可を定義する 1 つ以上の IAM プリンシパルにアタッチされたポリシー。

### アイドル状態のアプリケーション

90 日間の平均的な CPU およびメモリ使用率が 5~20% のアプリケーション。移行プロジェクトでは、これらのアプリケーションを廃止するか、オンプレミスに保持するのが一般的です。

## IIoT

「[インダストリアル IoT](#)」を参照してください。

### イミュータブルインフラストラクチャ

既存インフラストラクチャの更新、パッチ適用、変更などを行わずに、本番環境ワークロードに使用する新規インフラストラクチャをデプロイするモデル。本質的に、イミュータブルインフラストラクチャは、[ミュータブルインフラストラクチャ](#)よりも一貫性、信頼性、予測性に優れています。詳細については、AWS Well-Architected フレームワークにある「[イミュータブルインフラストラクチャを使用してデプロイする](#)」のベストプラクティスを参照してください。

### インバウンド (受信) VPC

AWS マルチアカウントアーキテクチャでは、アプリケーションの外部からネットワーク接続を受け入れ、検査し、ルーティングする VPC。[AWS Security Reference Architecture](#) では、アプリケーションとより広範なインターネット間の双方向のインターフェイスを保護するために、インバウンド、アウトバウンド、インスペクションの各 VPC を使用してネットワークアカウントを設定することを推奨しています。

## 増分移行

アプリケーションを 1 回ですべてカットオーバーするのではなく、小さい要素に分けて移行するカットオーバー戦略。例えば、最初は少数のマイクロサービスまたはユーザーのみを新しいシステムに移行する場合があります。すべてが正常に機能することを確認できたら、残りのマイクロサービスやユーザーを段階的に移行し、レガシーシステムを廃止できるようにします。この戦略により、大規模な移行に伴うリスクが軽減されます。

## インダストリー 4.0

2016 年に [Klaus Schwab](#) 氏が提唱した用語で、接続、リアルタイムデータ、オートメーション、分析、AI/ML の進歩による、ビジネスプロセスのモダナイズを意味します。

## インフラストラクチャ

アプリケーションの環境に含まれるすべてのリソースとアセット。

## Infrastructure as Code (IaC)

アプリケーションのインフラストラクチャを一連の設定ファイルを使用してプロビジョニングし、管理するプロセス。IaC は、新しい環境を再現可能で信頼性が高く、一貫性のあるものにするため、インフラストラクチャを一元的に管理し、リソースを標準化し、スケールを迅速に行えるように設計されています。

## インダストリアル IoT (IIoT)

製造、エネルギー、自動車、ヘルスケア、ライフサイエンス、農業などの産業部門におけるインターネットに接続されたセンサーやデバイスの使用。詳細については、「[インダストリアル IoT \(IIoT\) デジタルトランスフォーメーション戦略の構築](#)」を参照してください。

## インスペクション VPC

AWS マルチアカウントアーキテクチャでは、VPC (同一または異なる 内 AWS リージョン)、インターネット、オンプレミスネットワーク間のネットワークトラフィックの検査を管理する一元化された VPCs。 [AWS Security Reference Architecture](#) では、アプリケーションとより広範なインターネット間の双方向のインターフェイスを保護するために、インバウンド、アウトバウンド、インスペクションの各 VPC を使用してネットワークアカウントを設定することを推奨しています。

## IoT

インターネットまたはローカル通信ネットワークを介して他のデバイスやシステムと通信する、センサーまたはプロセッサが組み込まれた接続済み物理オブジェクトのネットワーク。詳細については、「[IoT とは](#)」を参照してください。

## 解釈可能性

機械学習モデルの特性で、モデルの予測がその入力にどのように依存するかを人間が理解できる度合いを表します。詳細については、[「を使用した機械学習モデルの解釈可能性 AWS」](#)を参照してください。

## IoT

[「IoT」](#)を参照してください。

## IT 情報ライブラリ (ITIL)

IT サービスを提供し、これらのサービスをビジネス要件に合わせるための一連のベストプラクティス。ITIL は ITSM の基盤を提供します。

## IT サービス管理 (ITSM)

組織の IT サービスの設計、実装、管理、およびサポートに関連する活動。クラウドオペレーションと ITSM ツールの統合については、[オペレーション統合ガイド](#)を参照してください。

## ITIL

[「IT 情報ライブラリ」](#)を参照してください。

## ITSM

[「IT サービス管理」](#)を参照してください。

## L

## ラベルベースアクセス制御 (LBAC)

強制アクセス制御 (MAC) の実装で、ユーザーとデータ自体にそれぞれセキュリティラベル値が明示的に割り当てられます。ユーザーセキュリティラベルとデータセキュリティラベルが交差する部分によって、ユーザーに表示される行と列が決まります。

## ランディングゾーン

ランディングゾーンは、スケーラブルで安全な、適切に設計されたマルチアカウント AWS 環境です。これは、組織がセキュリティおよびインフラストラクチャ環境に自信を持ってワークロードとアプリケーションを迅速に起動してデプロイできる出発点です。ランディングゾーンの詳細については、[「安全でスケーラブルなマルチアカウント AWS 環境のセットアップ」](#)を参照してください。

## 大規模言語モデル (LLM)

大量のデータで事前トレーニングされた深層学習 AI モデル。LLM では、質問への回答、ドキュメントの要約、他言語へのテキスト翻訳、文を完成させるなど、さまざまなタスクを実行できます。詳細については、「[大規模言語モデル \(LLM\) とは何ですか?](#)」を参照してください。

### 大規模な移行

300 台以上のサーバの移行。

### LBAC

「[ラベルベースアクセス制御](#)」を参照してください。

### 最小特権

タスクの実行には必要最低限の権限を付与するという、セキュリティのベストプラクティス。詳細については、IAM ドキュメントの「[最小特権アクセス許可を適用する](#)」を参照してください。

### リフトアンドシフト

「[7 Rs](#)」を参照してください。

### リトルエンディアンシステム

最下位バイトを最初に格納するシステム。「[エンディアン性](#)」もご覧ください。

### LLM

「[大規模言語モデル](#)」を参照してください。

### 下位環境

「[環境](#)」を参照してください。

## M

### 機械学習 (ML)

パターン認識と学習にアルゴリズムと手法を使用する人工知能の一種。ML は、モノのインターネット (IoT) データなどの記録されたデータを分析して学習し、パターンに基づく統計モデルを生成します。詳細については、「[機械学習](#)」を参照してください。

### メインブランチ

「[ブランチ](#)」を参照してください。

## マルウェア

コンピュータのセキュリティやプライバシーを侵害するように設計されたソフトウェア。マルウェアは、コンピュータシステムの中断、機密情報の漏洩、不正アクセスを招く可能性があります。マルウェアの例には、ウイルス、ワーム、ランサムウェア、トロイの木馬、スパイウェア、キーロガーなどがあります。

## マネージドサービス

AWS のサービスはインフラストラクチャレイヤー、オペレーティングシステム、プラットフォーム AWS を運用し、エンドポイントにアクセスしてデータを保存および取得します。マネージドサービスの例として、Amazon Simple Storage Service (Amazon S3) と Amazon DynamoDB が挙げられます。このサービスは、抽象化されたサービスとも呼ばれます。

## 製造実行システム (MES)

生産プロセスを追跡、モニタリング、文書化、制御するソフトウェアシステムであり、工場では、これによって、原材料から製品を完成させます。

## MAP

[「Migration Acceleration Program」](#) を参照してください。

## メカニズム

ツールを作成してその導入を推進し、導入結果を調べて調整を行うための包括的なプロセス。メカニズムとは、運用中にそれ自体を強化し改善するサイクルを意味します。詳細については、AWS 「Well-Architected フレームワーク」の [「メカニズムの構築」](#) を参照してください。

## メンバーアカウント

組織の一部である管理アカウント AWS アカウント 以外のすべて AWS Organizations。アカウントが組織のメンバーになることができるのは、一度に 1 つのみです。

## MES

[「製造実行システム」](#) を参照してください。

## Message Queuing Telemetry Transport (MQTT)

[発行/サブスクリプション](#) のパターンに基づく、軽量のマシンツーマシン (M2M) 通信プロトコルであり、リソースに限りのある [IoT](#) デバイスに使用されます。

## マイクロサービス

明確に定義された API を介して通信し、通常は小規模な自己完結型のチームが所有する、小規模で独立したサービスです。例えば、保険システムには、販売やマーケティングなどのビジネス

機能、または購買、請求、分析などのサブドメインにマッピングするマイクロサービスが含まれる場合があります。マイクロサービスの利点には、俊敏性、柔軟なスケーリング、容易なデプロイ、再利用可能なコード、回復力などがあります。詳細については、[AWS「サーバーレスサービスを使用したマイクロサービスの統合」](#)を参照してください。

## マイクロサービスアーキテクチャ

各アプリケーションプロセスをマイクロサービスとして実行する独立したコンポーネントを使用してアプリケーションを構築するアプローチ。これらのマイクロサービスは、軽量 API を使用して、明確に定義されたインターフェイスを介して通信します。このアーキテクチャの各マイクロサービスは、アプリケーションの特定の機能に対する需要を満たすように更新、デプロイ、およびスケーリングできます。詳細については、「[でのマイクロサービスの実装 AWS](#)」を参照してください。

## Migration Acceleration Program (MAP)

組織がクラウドに移行するための強力な運用基盤を構築し、移行の初期コストを相殺するのに役立つコンサルティングサポート、トレーニング、サービスを提供する AWS プログラム。MAP には、組織的な方法でレガシー移行を実行するための移行方法論と、一般的な移行シナリオを自動化および高速化する一連のツールが含まれています。

## 大規模な移行

アプリケーションポートフォリオの大部分を次々にクラウドに移行し、各ウェーブでより多くのアプリケーションを高速に移動させるプロセス。この段階では、以前の段階から学んだベストプラクティスと教訓を使用して、移行ファクトリー チーム、ツール、プロセスのうち、オートメーションとアジャイルデリバリーによってワークロードの移行を合理化します。これは、[AWS 移行戦略](#) の第 3 段階です。

## 移行ファクトリー

自動化された俊敏性のあるアプローチにより、ワークロードの移行を合理化する部門横断的なチーム。移行ファクトリーチームには、通常、運用、ビジネスアナリストおよび所有者、移行エンジニア、デベロッパー、およびスプリントで作業する DevOps プロフェッショナルが含まれます。エンタープライズアプリケーションポートフォリオの 20~50% は、ファクトリーのアプローチによって最適化できる反復パターンで構成されています。詳細については、このコンテンツセットの[移行ファクトリーに関する解説](#)と [Cloud Migration Factory ガイド](#)を参照してください。

## 移行メタデータ

移行を完了するために必要なアプリケーションおよびサーバーに関する情報。移行パターンごとに、異なる一連の移行メタデータが必要です。移行メタデータの例としては、ターゲットサブネット、セキュリティグループ、AWS アカウントなどがあります。

## 移行パターン

移行戦略、移行先、および使用する移行アプリケーションまたはサービスを詳述する、反復可能な移行タスク。例: AWS Application Migration Service を使用して Amazon EC2 への移行をリホストします。

## Migration Portfolio Assessment (MPA)

オンラインツール。これによって、AWS クラウドに移行するビジネスケースの検証に必要な情報を得られます。MPA は、詳細なポートフォリオ評価 (サーバーの適切なサイジング、価格設定、TCO 比較、移行コスト分析) および移行プラン (アプリケーションデータの分析とデータ収集、アプリケーションのグループ化、移行の優先順位付け、およびウェーブプランニング) を提供します。[MPA ツール](#) (ログインが必要) は、すべての AWS コンサルタントと APN パートナー コンサルタントが無料で利用できます。

## 移行準備状況評価 (MRA)

AWS CAF を使用して、組織のクラウド準備状況に関するインサイトを取得し、長所と短所を特定し、特定されたギャップを埋めるためのアクションプランを構築するプロセス。詳細については、[移行準備状況ガイド](#)を参照してください。MRA は、[AWS 移行戦略](#)の第一段階です。

## 移行戦略

ワークロードを AWS クラウドに移行するために使用するアプローチ。詳細については、この用語集の [7 Rs](#) エントリと、「[組織を動員して大規模な移行を加速する](#)」を参照してください。

## ML

「[機械学習](#)」を参照してください。

## モダナイゼーション

古い (レガシーまたはモノリシック) アプリケーションとそのインフラストラクチャをクラウド内の俊敏で弾力性のある高可用性システムに変換して、コストを削減し、効率を高め、イノベーションを活用します。詳細については、「[AWS クラウドでのアプリケーションのモダナイズ戦略](#)」を参照してください。

## モダナイゼーション準備状況評価

組織のアプリケーションのモダナイゼーションの準備状況を判断し、利点、リスク、依存関係を特定し、組織がこれらのアプリケーションの将来の状態をどの程度適切にサポートできるかを決定するのに役立つ評価。評価の結果として、ターゲットアーキテクチャのブループリント、モダナイゼーションプロセスの開発段階とマイルストーンを詳述したロードマップ、特定されたギャップに対処するためのアクションプランが得られます。詳細については、「[AWS クラウドでのアプリケーションのモダナイゼーションの準備状況を評価する](#)」を参照してください。

### モノリシックアプリケーション (モノリス)

緊密に結合されたプロセスを持つ単一のサービスとして実行されるアプリケーション。モノリシックアプリケーションにはいくつかの欠点があります。1つのアプリケーション機能エクスペリエンスの需要が急増する場合は、アーキテクチャ全体をスケーリングする必要があります。モノリシックアプリケーションの特徴を追加または改善することは、コードベースが大きくなると複雑になります。これらの問題に対処するには、マイクロサービスアーキテクチャを使用できます。詳細については、「[モノリスをマイクロサービスに分解する](#)」を参照してください。

### MPA

「[Migration Portfolio Assessment](#)」を参照してください。

### MQTT

「[Message Queuing Telemetry Transport](#)」を参照してください。

### 多クラス分類

複数のクラスの予測を生成するプロセス (2 つ以上の結果の 1 つを予測します)。例えば、機械学習モデルが、「この製品は書籍、自動車、電話のいずれですか?」または、「このお客様にとって最も関心のある商品のカテゴリはどれですか?」と聞くかもしれません。

### ミュータブルなインフラストラクチャ

本番ワークロードに使用する既存のインフラストラクチャを更新および変更するためのモデル。Well-Architected AWS フレームワークでは、一貫性、信頼性、予測可能性を向上させるために、[イミュータブルインフラストラクチャ](#)の使用をベストプラクティスとして推奨しています。

## O

### OAC

「[オリジンアクセス制御](#)」を参照してください。

## OAI

「[オリジンアクセスアイデンティティ](#)」を参照してください。

## OCM

「[組織変更管理](#)」を参照してください。

## オフライン移行

移行プロセス中にソースワークロードを停止させる移行方法。この方法はダウンタイムが長くなるため、通常は重要ではない小規模なワークロードに使用されます。

## OI

「[オペレーション統合](#)」を参照してください。

## Ola

「[オペレーショナルレベルアグリーメント](#)」を参照してください。

## オンライン移行

ソースワークロードをオフラインにせずにターゲットシステムにコピーする移行方法。ワークロードに接続されているアプリケーションは、移行中も動作し続けることができます。この方法はダウンタイムがゼロから最小限で済むため、通常は重要な本番稼働環境のワークロードに使用されます。

## OPC-UA

「[Open Process Communications - Unified Architecture](#)」を参照してください。

## Open Process Communications - Unified Architecture (OPC-UA)

産業オートメーション用のマシンツーマシン (M2M) 通信プロトコル。OPC-UA により、相互運用の際に、データ暗号化、認証、認可の各スキームを標準化できます。

## オペレーショナルレベルアグリーメント (OLA)

サービスレベルアグリーメント (SLA) をサポートするために、どの機能的 IT グループが互いに提供することを約束するかを明確にする契約。

## 運用準備状況レビュー (ORR)

質問と関連するベストプラクティスのチェックリスト。インシデントや起こり得る障害を理解、評価、防止したり、その範囲を縮小したりする際に役立ちます。詳細については、AWS Well-Architected フレームワークの「[Operational Readiness Reviews \(ORR\)](#)」を参照してください。

## 運用テクノロジー (OT)

産業オペレーション、機器、インフラストラクチャを制御するために物理環境と連携させるハードウェアおよびソフトウェアシステム。製造分野では、[Industry 4.0](#) への変革を進める上で、OT と情報技術 (IT) システムの統合に焦点が当てられています。

## オペレーション統合 (OI)

クラウドでオペレーションをモダナイズするプロセスには、準備計画、オートメーション、統合が含まれます。詳細については、[オペレーション統合ガイド](#)を参照してください。

## 組織の証跡

組織 AWS アカウント 内のすべてのイベント AWS CloudTrail をログに記録することによって作成された証跡 AWS Organizations。証跡は、組織に含まれている各 AWS アカウントに作成され、各アカウントのアクティビティを追跡します。詳細については、CloudTrail ドキュメントの「[組織の証跡の作成](#)」を参照してください。

## 組織変更管理 (OCM)

人材、文化、リーダーシップの観点から、主要な破壊的なビジネス変革を管理するためのフレームワーク。OCM は、変化の導入を加速し、移行問題に対処し、文化や組織の変化を推進することで、組織が新しいシステムと戦略の準備と移行するのを支援します。AWS 移行戦略では、クラウド導入プロジェクトに必要な変化のスピードにより、このフレームワークは人材アクセラレーションと呼ばれます。詳細については、[OCM ガイド](#)を参照してください。

## オリジンアクセス制御 (OAC)

Amazon Simple Storage Service (Amazon S3) コンテンツを保護するための、CloudFront のアクセス制限の強化オプション。OAC は AWS リージョン、すべての S3 バケット、AWS KMS (SSE-KMS) によるサーバー側の暗号化、S3 バケットへの動的 PUT および DELETE リクエストをサポートします。

## オリジンアクセスアイデンティティ (OAI)

CloudFront の、Amazon S3 コンテンツを保護するためのアクセス制限オプション。OAI を使用すると、CloudFront が、Amazon S3 に認証可能なプリンシパルを作成します。認証されたプリンシパルは、S3 バケット内のコンテンツに、特定の CloudFront ディストリビューションを介してのみアクセスできます。[OAC](#) も併せて参照してください。OAC では、より詳細な、強化されたアクセス制御が可能です。

## ORR

「[運用準備状況レビュー](#)」を参照してください。

## OT

「[運用テクノロジー](#)」を参照してください。

### アウトバウンド (送信) VPC

AWS マルチアカウントアーキテクチャでは、アプリケーション内から開始されたネットワーク接続を処理する VPC。AWS Security Reference Architecture では、アプリケーションとより広範なインターネット間の双方向のインターフェイスを保護するために、インバウンド、アウトバウンド、インスペクションの各 VPC を使用してネットワークアカウントを設定することを推奨しています。

## P

### アクセス許可の境界

ユーザーまたはロールが使用できるアクセス許可の上限を設定する、IAM プリンシパルにアタッチされる IAM 管理ポリシー。詳細については、IAM ドキュメントの[アクセス許可の境界](#)を参照してください。

### 個人を特定できる情報 (PII)

直接閲覧した場合、または他の関連データと組み合わせた場合に、個人の身元を合理的に推測するために使用できる情報。PII の例には、氏名、住所、連絡先情報などがあります。

## PII

「[個人を特定できる情報](#)」を参照してください。

### プレイブック

クラウドでのコアオペレーション機能の提供など、移行に関連する作業を取り込む、事前定義された一連のステップ。プレイブックは、スクリプト、自動ランブック、またはお客様のモダナイズされた環境を運用するために必要なプロセスや手順の要約などの形式をとることができます。

## PLC

「[プログラマブルロジックコントローラー](#)」を参照してください。

## PLM

「[製品ライフサイクル管理](#)」を参照してください。

## ポリシー

次の操作を可能にするオブジェクト: アクセス許可を定義する ([ID ベースのポリシー](#)を参照)。アクセス条件を指定する ([リソースベースのポリシー](#)を参照)。AWS Organizations の組織における全アカウントにアクセス許可の上限を定義する ([サービスコントロールポリシー](#)を参照)。

## 多言語の永続性

データアクセスパターンやその他の要件に基づいて、マイクロサービスのデータストレージテクノロジーを個別に選択します。マイクロサービスが同じデータストレージテクノロジーを使用している場合、実装上の問題が発生したり、パフォーマンスが低下する可能性があります。マイクロサービスは、要件に最も適合したデータストアを使用すると、より簡単に実装でき、パフォーマンスとスケーラビリティが向上します。

## ポートフォリオ評価

移行を計画するために、アプリケーションポートフォリオの検出、分析、優先順位付けを行うプロセス。詳細については、「[移行の準備状況の評価](#)」を参照してください。

## 述語

true または false を返すためのクエリ条件。一般的に、WHERE 句に記述されます。

## 述語プッシュダウン

データベースクエリを最適化する手法。これによって、転送前にクエリ内のデータをフィルタリングします。この手法を取ると、リレーショナルデータベースから取得し処理する必要のあるデータの量が減少するため、クエリのパフォーマンスが向上します。

## 予防的コントロール

イベントの発生を防ぐように設計されたセキュリティコントロール。このコントロールは、ネットワークへの不正アクセスや好ましくない変更を防ぐ最前線の防御です。詳細については、「AWSでのセキュリティコントロールの実装」の「[予防的コントロール](#)」を参照してください。

## プリンシパル

アクションを実行し AWS、リソースにアクセスできるエンティティ。このエンティティは通常、IAM AWS アカウントロール、またはユーザーのルートユーザーです。詳細については、IAM ドキュメントの「[ロールに関する用語と概念](#)」にあるプリンシパルを参照してください。

## プライバシーバイデザイン

開発プロセス全体を通してプライバシーが考慮されているシステムエンジニアリングのアプローチ。

## プライベートホストゾーン

1 つ以上の VPC 内のドメインとそのサブドメインへの DNS クエリに対し、Amazon Route 53 がどのように応答するかに関する情報を保持するコンテナ。詳細については、Route 53 ドキュメントの「[プライベートホストゾーンの使用](#)」を参照してください。

## プロアクティブコントロール

非準拠リソースのデプロイ防止を目的とした[セキュリティコントロール](#)。このコントロールにより、プロビジョニング前にリソースをスキャンします。コントロールに準拠していないリソースは、プロビジョニングされません。詳細については、AWS Control Tower ドキュメントの「[コントロールリファレンスガイド](#)」および「[セキュリティコントロールの実装](#)」の「[プロアクティブコントロール](#)」を参照してください。 AWS

## 製品ライフサイクル管理 (PLM)

製品の設計、開発、発売から、成長、成熟、衰退、廃棄に至る、製品のライフサイクル全体を通してデータとプロセスを管理すること。

## 本番環境

「[環境](#)」を参照してください。

## プログラマブルロジックコントローラー (PLC)

製造分野で使用される、信頼性と適応性に優れたコンピュータであり、これによって、マシンをモニタリングするとともに、製造プロセスを自動化します。

## プロンプトチェイニング

1 つの [LLM](#) プロンプトによる出力を次のプロンプトの入力に使用して、より良いレスポンスを生成します。この手法を使用すると、複雑なタスクをサブタスクに分割したり、事前レスポンスを繰り返し改良または拡張したりできます。これによって、モデルのレスポンスの精度と関連性が向上し、粒度の高いパーソナライズされた結果を得られます。

## 仮名化

データセット内の個人識別子をプレースホルダー値に置き換えるプロセス。仮名化は個人のプライバシー保護に役立ちます。仮名化されたデータは、依然として個人データとみなされます。

## 発行/サブスクライブ (pub/sub)

マイクロサービス間の非同期通信を可能にするパターン。これにより、スケーラビリティと応答性を向上させます。例えば、マイクロサービスベースの [MES](#) の場合、マイクロサービスは、他のマイクロサービスがサブスクライブ可能なチャンネルにイベントメッセージを発行できます。このシステムでは、発行サービスの変更なしに、新規マイクロサービスを追加できます。

## Q

### クエリプラン

手順などの一連のステップであり、SQL リレーショナルデータベースシステムのデータにアクセスするために使用されます。

### クエリプランのリグレッション

データベースサービスのオプティマイザーが、データベース環境に特定の変更が加えられる前に選択されたプランよりも最適性の低いプランを選択すること。これは、統計、制限事項、環境設定、クエリパラメータのバインディングの変更、およびデータベースエンジンの更新などが原因である可能性があります。

## R

### RACI マトリックス

「[実行責任者、説明責任者、協業先、報告先 \(RACI\)](#)」を参照してください。

### RAG

「[検索拡張生成](#)」を参照してください。

### ランサムウェア

決済が完了するまでコンピュータシステムまたはデータへのアクセスをブロックするように設計された、悪意のあるソフトウェア。

### RASCI マトリックス

「[実行責任者、説明責任者、協業先、報告先 \(RACI\)](#)」を参照してください。

### RCAC

「[行と列のアクセス制御](#)」を参照してください。

### リードレプリカ

読み取り専用で使用されるデータベースのコピー。クエリをリードレプリカにルーティングして、プライマリデータベースへの負荷を軽減できます。

### リアーキテクト

「[7 Rs](#)」を参照してください。

## 目標復旧時点 (RPO)

最後のデータリカバリポイントからの最大許容時間です。これにより、最後の回復時点からサービスが中断されるまでの間に許容できるデータ損失の程度が決まります。

## 目標復旧時間 (RTO)

サービスが中断から復旧までの最大許容遅延時間。

## リファクタリング

「[7 Rs](#)」を参照してください。

## リージョン

地理的エリア内の AWS リソースのコレクション。各 AWS リージョンは、耐障害性、安定性、耐障害性を提供するために、他のから分離され、独立しています。詳細については、「[アカウントが使用できる AWS リージョンを指定する](#)」を参照してください。

## リグレッション

数値を予測する機械学習手法。例えば、「この家はどれくらいの値段で売れるでしょうか?」という問題を解決するために、機械学習モデルは、線形回帰モデルを使用して、この家に関する既知の事実 (平方フィートなど) に基づいて家の販売価格を予測できます。

## リホスト

「[7 Rs](#)」を参照してください。

## リリース

デプロイプロセスで、変更を本番環境に昇格させること。

## 再配置

「[7 Rs](#)」を参照してください。

## リプラットフォーム

「[7 Rs](#)」を参照してください。

## 再購入

「[7 Rs](#)」を参照してください。

## 回復性

中断に抵抗または中断から回復するアプリケーションの機能。AWS クラウドでの回復力を計画する際には、一般的に、[高可用性](#)と[ディザスタリカバリ](#)が考慮されます。詳細については、「[AWS クラウドの耐障害性](#)」を参照してください。

## リソースベースのポリシー

Amazon S3 バケット、エンドポイント、暗号化キーなどのリソースにアタッチされたポリシー。このタイプのポリシーは、アクセスが許可されているプリンシパル、サポートされているアクション、その他の満たすべき条件を指定します。

## 実行責任者、説明責任者、協業先、報告先 (RACI) に基づくマトリックス

移行活動とクラウド運用に関わるすべての関係者の役割と責任を定義したマトリックス。マトリックスの名前は、マトリックスで定義されている責任の種類、すなわち責任 (R)、説明責任 (A)、協議 (C)、情報提供 (I) に由来します。サポート (S) タイプはオプションです。サポートが含まれる場合は RASCI マトリックスと呼ばれ、含まれない場合は RACI マトリックスと呼ばれます。

## レスポンスコントロール

有害事象やセキュリティベースラインからの逸脱について、修復を促すように設計されたセキュリティコントロール。詳細については、「AWSでのセキュリティコントロールの実装」の「[レスポンスコントロール](#)」を参照してください。

## 保持

「[7 Rs](#)」を参照してください。

## 廃止

「[7 Rs](#)」を参照してください。

## 検索拡張生成 (RAG)

[生成 AI](#) の技術。これにより、[LLM](#) では、レスポンスの生成前に、トレーニングデータソースの外部にある信頼できるデータソースが参照されます。例えば、RAG モデルによって、組織のナレッジベースまたはカスタムデータのセマンティック検索を実行できる場合があります。細については、「[RAG \(検索拡張生成\) とは何ですか?](#)」を参照してください。

## ローテーション

定期的に[シークレット情報](#)を更新して、攻撃者が認証情報にアクセスするのをより困難にするプロセス。

## 行と列のアクセス制御 (RCAC)

アクセスルールが定義された、基本的で柔軟な SQL 表現の使用。RCAC は行権限と列マスクで構成されています。

## RPO

「[目標復旧時点](#)」を参照してください。

## RTO

「[目標復旧時間](#)」を参照してください。

## ランブック

特定のタスクを実行するために必要な手動または自動化された一連の手順。これらは通常、エラー率の高い反復操作や手順を合理化するために構築されています。

## S

### SAML 2.0

多くの ID プロバイダー (IdP) が使用しているオープンスタンダード。この機能を使用すると、フェデレーテッドシングルサインオン (SSO) が有効になるため、ユーザーは組織内のすべてのユーザーを IAM で作成しなくても、AWS マネジメントコンソールにログインしたり AWS、API オペレーションを呼び出すことができます。SAML 2.0 ベースのフェデレーションの詳細については、IAM ドキュメントの「[SAML 2.0 ベースのフェデレーションについて](#)」を参照してください。

### SCADA

「[監視制御とデータ取得](#)」を参照してください。

### SCP

「[サービスコントロールポリシー](#)」を参照してください。

## シークレット

暗号化された形式で保存する AWS Secrets Manager パスワードやユーザー認証情報などの機密情報または制限付き情報。シークレット値とそのメタデータで構成されます。シークレット値には、バイナリ、1 つの文字列、複数の文字列を指定できます。詳細については、Secrets Manager ドキュメントの「[Secrets Manager シークレットの概要](#)」を参照してください。

## セキュリティバイデザイン

開発プロセス全体を通してセキュリティが考慮されているシステムエンジニアリングのアプローチ。

## セキュリティコントロール

脅威アクターによるセキュリティ脆弱性の悪用を防止、検出、軽減するための、技術上または管理上のガードレール。セキュリティコントロールには、主に 4 つの種類があります。4 つとは、[予防](#)、[検出](#)、[レスポンス](#)、[プロアクティブ](#)です。

### セキュリティ強化

アタックサーフェスを狭めて攻撃への耐性を高めるプロセス。このプロセスには、不要になったリソースの削除、最小特権を付与するセキュリティのベストプラクティスの実装、設定ファイル内の不要な機能の無効化、といったアクションが含まれています。

### Security Information and Event Management (SIEM) システム

セキュリティ情報管理 (SIM) とセキュリティイベント管理 (SEM) のシステムを組み合わせたツールとサービス。SIEM システムは、サーバー、ネットワーク、デバイス、その他ソースからデータを収集、モニタリング、分析して、脅威やセキュリティ違反を検出し、アラートを発信します。

### セキュリティレスポンスの自動化

セキュリティイベントへの自動レスポンスまたは自動修復を目的として、事前定義およびプログラムされたアクション。これらの自動化は、セキュリティのベストプラクティスを実装するのに役立つ[検出的](#)または[応答的](#)な AWS セキュリティコントロールとして機能します。自動レスポンスアクションの例には、VPC セキュリティグループの変更、Amazon EC2 インスタンスへのパッチ適用、認証情報の更新などがあります。

### サーバー側の暗号化

送信先で、それ AWS のサービスを受け取る によるデータの暗号化。

### サービスコントロールポリシー (SCP)

AWS Organizationsの組織内の、すべてのアカウントのアクセス許可を一元的に管理するポリシー。SCP は、管理者がユーザーまたはロールに委任するアクションに、ガードレールを定義したり、アクションの制限を設定したりします。SCP は、許可リストまたは拒否リストとして、許可または禁止するサービスやアクションを指定する際に使用できます。詳細については、AWS Organizations ドキュメントの「[サービスコントロールポリシー](#)」を参照してください。

### サービスエンドポイント

のエンドポイントの URL AWS のサービス。ターゲットサービスにプログラムで接続するには、エンドポイントを使用します。詳細については、「AWS 全般のリファレンス」の「[AWS のサービス エンドポイント](#)」を参照してください。

## サービスレベルアグリーメント (SLA)

サービスのアップタイムやパフォーマンスなど、IT チームがお客様に提供すると約束したものを明示した合意書。

## サービスレベルインジケータ (SLI)

エラー率、可用性、スループットといった、サービスパフォーマンス面の指標。

## サービスレベル目標 (SLO)

[サービスレベルインジケータ](#)によって測定され、サービスの状態を表すターゲットメトリクス。

## 責任共有モデル

クラウドのセキュリティとコンプライアンス AWS について と共有する責任を説明するモデル。AWS はクラウドのセキュリティを担当しますが、 はクラウドのセキュリティを担当します。詳細については、「[責任共有モデル](#)」を参照してください。

## SIEM

「[Security Information and Event Management システム](#)」を参照してください。

## 単一障害点 (SPOF)

特定のアプリケーションを構成する単一の重要なコンポーネントで発生し、システム稼働に支障をきたす可能性のある障害。

## SLA

「[サービスレベルアグリーメント](#)」を参照してください。

## SLI

「[サービスレベルインジケータ](#)」を参照してください。

## SLO

「[サービスレベルの目標](#)」を参照してください。

## スプリットアンドシードモデル

モダナイゼーションプロジェクトのスケーリングと加速のためのパターン。新機能と製品リリースが定義されると、コアチームは解放されて新しい製品チームを作成します。これにより、お客様の組織の能力とサービスの拡張、デベロッパーの生産性の向上、迅速なイノベーションのサポートに役立ちます。詳細については、「[AWS クラウドでのアプリケーションをモダナイズするための段階的アプローチ](#)」を参照してください。

## SPOF

「[単一障害点](#)」を参照してください。

## スタースキーマ

データベースの編成構造を意味し、1つの大きいファクトテーブルにトランザクションデータまたは測定データが保存され、1つ以上の小さいディメンションテーブルにデータ属性が保存されます。この構造は、[データウェアハウス](#)やビジネスインテリジェンスを用途とするように設計されています。

## strangler fig パターン

レガシーシステムが廃止されるまで、システム機能を段階的に書き換えて置き換えることにより、モノリシックシステムをモダナイズするアプローチ。このパターンは、宿主の樹木から根を成長させ、最終的にその宿主を包み込み、宿主に取って代わるイチジクのつるを例えています。そのパターンは、モノリシックシステムを書き換えるときのリスクを管理する方法として [Martin Fowler](#) により提唱されました。このパターンの適用方法の例については、「[コンテナと Amazon API Gateway を使用して、従来の Microsoft ASP.NET \(ASMX\) ウェブサービスを段階的にモダナイズ](#)」を参照してください。

## サブネット

VPC 内の IP アドレスの範囲。サブネットは、1つのアベイラビリティゾーンに存在する必要があります。

## 監視制御とデータ取得 (SCADA)

製造分野において、ハードウェアとソフトウェアを使用して物理アセットと本番運用をモニタリングするシステム。

## 対称暗号化

データの暗号化と復号に同じキーを使用する暗号化のアルゴリズム。

## 合成テスト

ユーザーとのやり取りをシミュレートして、起こり得る問題を検出したり、パフォーマンスをモニタリングしたりすることで、システムをテストします。[Amazon CloudWatch Synthetics](#) を使用すると、こうしたテストを作成できます。

## システムプロンプト

コンテキスト、指示、ガイドラインなどを提示して、[LLM](#) に動作を指示する手法。システムプロンプトは、コンテキストを設定して、ユーザーとやり取りするルールを確立するのに有用です。

## T

### タグ

AWS リソースを整理するためのメタデータとして機能するキーと値のペア。タグは、リソースの管理、識別、整理、検索、フィルタリングに役立ちます。詳細については、「[AWS リソースのタグ付け](#)」を参照してください。

### ターゲット変数

監督された機械学習でお客様が予測しようとしている値。これは、結果変数のことも指します。例えば、製造設定では、ターゲット変数が製品の欠陥である可能性があります。

### タスクリスト

ランブックの進行状況を追跡するために使用されるツール。タスクリストには、ランブックの概要と完了する必要がある一般的なタスクのリストが含まれています。各一般的なタスクには、推定所要時間、所有者、進捗状況が含まれています。

### テスト環境

「[環境](#)」を参照してください。

### トレーニング

お客様の機械学習モデルに学習するデータを提供すること。トレーニングデータには正しい答えが含まれている必要があります。学習アルゴリズムは入力データ属性をターゲット (お客様が予測したい答え) にマッピングするトレーニングデータのパターンを検出します。これらのパターンをキャプチャする機械学習モデルを出力します。そして、お客様が機械学習モデルを使用して、ターゲットがわからない新しいデータでターゲットを予測できます。

### トランジットゲートウェイ

VPC とオンプレミスネットワークを相互接続するために使用できる、ネットワークの中継ハブ。詳細については、AWS Transit Gateway ドキュメントの「[トランジットゲートウェイとは](#)」を参照してください。

### トランクベースのワークフロー

デベロッパーが機能ブランチで機能をローカルにビルドしてテストし、その変更をメインブランチにマージするアプローチ。メインブランチはその後、開発環境、本番前環境、本番環境に合わせて順次構築されます。

## 信頼されたアクセス

ユーザーに代わって AWS Organizations およびそのアカウントで組織内でタスクを実行するために指定したサービスにアクセス許可を付与します。信頼されたサービスは、サービスにリンクされたロールを必要とときに各アカウントに作成し、ユーザーに代わって管理タスクを実行します。詳細については、ドキュメントの「[Using AWS Organizations with other AWS services](#) AWS Organizations」を参照してください。

## チューニング

機械学習モデルの精度を向上させるために、お客様のトレーニングプロセスの側面を変更する。例えば、お客様が機械学習モデルをトレーニングするには、ラベル付けセットを生成し、ラベルを追加します。これらのステップを、異なる設定で複数回繰り返して、モデルを最適化します。

## ツーピザチーム

2 枚のピザを分け合えることができるくらい小さな DevOps チーム。ツーピザチームの規模では、ソフトウェア開発におけるコラボレーションに最適な機会が確保されます。

# U

## 不確実性

予測機械学習モデルの信頼性を損なう可能性がある、不正確、不完全、または未知の情報を指す概念。不確実性には、次の 2 つのタイプがあります。認識論的不確実性は、限られた、不完全なデータによって引き起こされ、弁論的不確実性は、データに固有のノイズとランダム性によって引き起こされます。

## 未分化なタスク

ヘビーリフティングとも呼ばれ、アプリケーションの作成と運用には必要だが、エンドユーザーに直接的な価値をもたらさなかったり、競争上の優位性をもたらしたりしない作業です。未分化なタスクの例としては、調達、メンテナンス、キャパシティプランニングなどがあります。

## 上位環境

「[環境](#)」を参照してください。

## V

### バキューミング

ストレージを再利用してパフォーマンスを向上させるために、増分更新後にクリーンアップを行うデータベースのメンテナンス操作。

### バージョンコントロール

リポジトリ内のソースコードへの変更など、変更を追跡するプロセスとツール。

### VPC ピアリング

プライベート IP アドレスを使用してトラフィックをルーティングできる、2 つの VPC 間の接続。詳細については、Amazon VPC ドキュメントの「[VPC ピア機能とは](#)」を参照してください。

### 脆弱性

システムのセキュリティを脅かすソフトウェアまたはハードウェアの欠陥。

## W

### ウォームキャッシュ

頻繁にアクセスされる最新の関連データを含むバッファキャッシュ。データベースインスタンスはバッファキャッシュから、メインメモリまたはディスクからよりも短い時間で読み取りを行うことができます。

### ウォームデータ

アクセス頻度の低いデータ。この種類のデータをクエリする場合、通常は適度に遅いクエリでも問題ありません。

### ウィンドウ関数

現在のレコードに何らかの形で関連している行のグループに計算を実行する SQL 関数。ウィンドウ関数は、移動平均を計算したり、現在の行の相対位置に基づいて他の行の値にアクセスするといったタスクの処理に役立ちます。

### ワークロード

ビジネス価値をもたらすリソースとコード (顧客向けアプリケーションやバックエンドプロセスなど) の総称。

## ワークストリーム

特定のタスクセットを担当する移行プロジェクト内の機能グループ。各ワークストリームは独立していますが、プロジェクト内の他のワークストリームをサポートしています。たとえば、ポートフォリオワークストリームは、アプリケーションの優先順位付け、ウェーブ計画、および移行メタデータの収集を担当します。ポートフォリオワークストリームは、これらの設備を移行ワークストリームで実現し、サーバーとアプリケーションを移行します。

## WORM

「[Write-Once-Read-Many](#)」を参照してください。

## WQF

「[AWS ワークロード資格フレームワーク](#)」を参照してください

## Write-Once-Read-Many (WORM)

データを 1 回のみ書き込むことで、データの削除や変更を防ぐストレージモデル。承認済みユーザーは、必要な回数だけデータを読み取ることができますが、変更することはできません。このデータストレージインフラストラクチャは、[イミュータブル](#)と見なされます。

## Z

### ゼロデイ 익스プロイト

[ゼロデイ脆弱性](#)を悪用した攻撃 (一般的にマルウェアによる)。

### ゼロデイ脆弱性

実稼働システムにおける未解決の欠陥または脆弱性。脅威アクターは、このような脆弱性を利用してシステムを攻撃する可能性があります。開発者は、よく攻撃の結果で脆弱性に気付きます。

### ゼロショットプロンプト

[LLM](#) にタスク実行の手順は提示するが、実行のガイドとして役立つ例 (ショット) は提示しない方法。LLM は、事前トレーニング済みの知識を使用してタスクを処理する必要があります。ゼロショットプロンプトの有効性は、タスクの複雑さとプロンプトの品質によって異なります。「[数ショットプロンプト](#)」も参照してください。

### ゾンビアプリケーション

平均 CPU およびメモリ使用率が 5% 未満のアプリケーション。移行プロジェクトでは、これらのアプリケーションを廃止するのが一般的です。

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。