aws

で Essential Eight 成熟度に達する AWS

AWS 規範ガイダンス

AWS 規範ガイダンス: で Essential Eight 成熟度に達する AWS

Table of Contents

Essential Eight 成熟度に達する AWS: オーストラリア組織のセキュリティとコンプライアンス

Amazon Web Services (寄稿者)

2024 年 11 月 (ドキュメント履歴)

Australian Signals Directorate (ASD) は、組織がサイバーセキュリティ脅威のリスクを軽減できるよう支援する戦略を作成し、それらの優先順位付けを行いました。その戦略のうち 8 つが Essential Eight フレームワーク形成のために選択され、オーストラリアの公共部門および民間部門組織の多くが、Essential Eight フレームワークの成熟度に到達することを求められています。

Essential Eight フレームワークを作成したのは、Australian Cyber Security Centre (ACSC) ですが、その目的は、Microsoft ベースのインターネット接続ネットワークの保護を支援することでした。しかし、組織の多くが、オンプレミスとクラウドのいずれでもすべての環境で Essential Eight の成熟度への到達を求められています。

Essential Eight フレームワークには、組織が段階的なイテレーションを通じてフレームワークを実装できるように設計された成熟度モデルも定義されています。このモデルでは、成熟度レベル 0～3 が概説されており、成熟度レベル 3 は、高度なサイバーセキュリティ戦術や、ターゲットが非常に絞られた攻撃に対し、レジリエンスがあることを意味します。このガイドでは、Essential Eight の成熟度レベル 3 を達成するのに役立つ、具体的な意見に基づいたガイダンスを提供します AWS。

オーストラリア組織のセキュリティおよびコンプライアンス

オーストラリアの多くの組織は AWS クラウド 、 を使用して機密データの保存、機密取引の処理、重要なサービスの構築を行います。

このガイドでは、Essential Eight フレームワークのクラウドへの適応方法について説明しますが、AWS では、組織のセキュリティおよびコンプライアンス要件を満たすのに有用な認定やモデルも提供しています。例を挙げましょう。

- Information Security Registered Assessors Program
- ホスティング認定フレームワーク
- AWS 責任共有モデル
- AWS Well-Architected フレームワーク

Information Security Registered Assessors Program

AWS のサービス は、オーストラリアサイバーセキュリティセンター (ACSC) 情報セキュリティ登録評価プログラム (IRAP) の下で保護レベルで評価されています。オーストラリア信号局 (ASD) 認定の独立系 IRAP 評価者が IRAP 評価を完了しました AWS。この評価により、 AWS 製品およびサービスに関して、PROTECTED レベルのワークロードに適用可能なコントロールが実装されることが保証されます。

IRAP PROTECTED AWS パッケージは、 から入手できますAWS Artifact。IRAP レポートは、ACSC クラウドセキュリティガイダンス (ACSC ウェブサイト) を使用して開発されました。対象範囲内の AWS のサービス の完全なリストについては、「AWS のサービス in scope: IRAP」を参照してください。

ホスティング認定フレームワーク

オーストラリアのホスティング認定フレームワークは、政府のシステムおよびデータの安全な管理を支援するために開発されました。このフレームワークは、組織がサプライチェーンとデータセンターの所有権リスクを軽減するのに役立つことを目的としています。 AWS は、認定戦略レベルで認定されました。これにより、政府機関は が AWS 政府の要件を満たしていることを知り、急速なイノベーションを続けることができます。

AWS 責任共有モデル

責任AWS 共有モデルは、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 と責任を共有する方法を定義します。 は、 で提供されているすべてのサービスを実行するインフラストラクチャ AWS を保護するとともに AWS クラウド、データやアプリケーションなど、これらのサービスの使用を保護する責任があります。

この責任共有モデルでは、コンプライアンスと運用上の負担が軽減されます。なぜなら、ホストオペレーティングシステムと仮想化レイヤーから、サービスが稼働する施設の物理的なセキュリティに至る各種コンポーネントの運用、管理、制御は、 AWS が行うからです。お客様は、ゲストオペレーティングシステム (更新およびセキュリティパッチを含む) やその他の関連アプリケーションソフトウェアの管理に加え、 AWS が提供するセキュリティグループファイアウォールの設定を行う責任を負います。

Essential Eight の成熟度に近づくときは、責任 AWS 共有モデルを理解することが重要です AWS。お客様の責任は、利用するサービス、それらのサービスの IT 環境への統合、適用される法律や規制によって異なります。

AWS Well-Architected フレームワーク

AWS Well-Architected は、クラウドアーキテクトがさまざまなアプリケーションや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。AWS Well-Architected フレームワークは、システムの設計、構築、運用に役立つアーキテクチャのベストプラクティスを提供します AWS。運用上の優秀性、セキュリティ、信頼性、パフォーマンス効率、コスト最適化、持続可能性という 6 つの柱を中心にして構築されています。

AWS は、ワークロードを確認するサービスも提供します。は、 AWS Well-Architected フレームワークを使用してアーキテクチャを確認および評価するAWS Well-Architected Toolのに役立ちます。このツールにより、ワークロードの信頼性、安全性、効率性、費用対効果を高めるための推奨事項を得られます。

クラウドを考慮した、Essential Eight 戦略の再解釈

元の Essential Eight 緩和戦略を以下に示します。これらは、Microsoft ベースのインターネット接続ネットワーク向けに設計されています。

- アプリケーション制御
- アプリケーションへのパッチ適用
- Microsoft Office マクロ設定の構成
- ユーザーアプリケーションの強化
- 管理者権限の制限
- オペレーティングシステムへのパッチ適用
- 多要素認証
- 定期バックアップ

前述のとおり、Essential Eight フレームワークは、クラウド環境向けには設計されていません。ただし、基礎となる原則が適用され、Essential Eight 戦略と AWS Well-Architected Framework のベストプラクティスの間には重複があります。

クラウドネイティブのさまざまなアプローチを取ると、セキュリティが向上し、コンプライアンス上の負担が大幅に軽減されます。オンプレミス環境では、あらゆるセキュリティを自社の責任で確保する必要があるものの、管理機能があらかじめ用意されているわけではありません。クラウドでワークロードを実行する場合、 AWS は当社のサービスを実行するインフラストラクチャを保護する責任があります。自動化およびマネージドサービスを利用すると、コンプライアンス上の負担を軽減することも可能です。マネージドサービスは抽象化サービスとも呼ばれ、 AWS のサービス が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、プラットフォームを AWS 運用し、ユーザーが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。詳細については、このガイドの「テーマ 1: マネージドサービスの使用」セクションを参照してください。

したがって、Essential Eight 戦略を AWS上のワークロードに当てはまるように変更するには、いくつかの再解釈が必要です。このガイドでは、Essential Eight 戦略を AWS テーマに変換します。

テーマの使用

このガイドは 8 つのテーマに分かれています。各 Essential Eight 戦略は、次のテーマの 1 つ以上にマッピングされ、各テーマは AWS Well-Architected フレームワークの 1 つ以上のベストプラクティスにマッピングされます。

- テーマ 1: マネージドサービスの使用
- テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理
- テーマ 3: 自動化によるミュータブルなインフラストラクチャの管理
- テーマ 4: ID の管理
- テーマ 5: データ境界を確立する
- テーマ 6: バックアップの自動化
- テーマ 7: ログ記録およびモニタリングの一元化
- テーマ 8: 手動プロセスの仕組みの実装

各テーマには、トピックの概要、関連する AWS Well-Architected Framework のベストプラクティス、Essential Eight の成熟度を達成し、コンプライアンスをモニタリングする方法の手順が含まれています。こうした手順は、手動での手順や、AWS Config ルールによる自動化の設定方法を説明するものです。手動の手順には、検出結果に対処するための仕組みが必要です。詳細については、「」を参照してくださいテーマ 8: 手動プロセスの仕組みの実装。 AWS Config ルールは、非準拠のリソースを修復するために同様の監視または自動化を必要とします。 https://docs.aws.amazon.com/config/latest/developerguide/remediation.htmlこれらのテーマに沿ったガイダンスに従うと、クラウドのメリットが最大化するアプローチで Essential Eight の成熟度に到達できます。

クラウドを考慮した、Essential Eight 戦略の再解釈

Essential Eight フレームワークは、クラウド環境向けには設計されていないため、各 Essential Eight 戦略の基本原則を実装する際には、クラウドネイティブなアプローチを取ることが重要です。このアプローチは、2 つの重要な質問から導き出される現状によって異なります。

どのようなサービスを利用していますか?

AWS 責任共有モデル は、コンプライアンスと運用上の負担を軽減するのに有用です。マネージドサービスは、デプロイされたサービスの可用性、パフォーマンス、セキュリティ最適化 AWS を維持する責任を に移行します。サービス維持に伴う運用上および管理上の負担も軽減できるため、イノベーションへの取り組みにより多くの時間を割けるようになります。

マネージドサービスでは、Amazon API Gateway、AWS Lambda、DynamoDB などのサーバーレスサービスも提供されます。Amazon Relational Database Service (Amazon RDS) でデータベースを稼働させると、Amazon Elastic Compute Cloud (Amazon EC2) で稼働させる場合よりも、運用上の責任が少なくなります。

例えば、Patch オペレーティングシステムの Essential Eight 戦略をクラウドに適応させる場合は、使用しているサービスと、それらのリソースにパッチを適用する責任があるかどうかを考慮する必要があります。 AWS は、Lambda や DynamoDB などのフルマネージドサービスのパッチ適用を担当します。Amazon RDS や Amazon Redshift などのサービスでは、メンテナンスウィンドウの間に、お客様側でパッチ管理が必要になる場合があります。https://docs.aws.amazon.com/redshift/latest/gsg/new-user-serverless.html

どのようなデプロイモデルを使用していますか?

組織では、ミュータブルまたはイミュータブルなインフラストラクチャのアプローチを取っていますか?

ミュータブルインフラストラクチャモデルでは、本番ワークロードの既存インフラストラクチャを更新し、修正します。クラウド導入前には、こうした標準的なデプロイ方法が取られます。サーバーインフラストラクチャの置き換えにはコストと時間が非常にかかると考えると、最も実用的なアプローチは、既存の本番サーバーを変更することだからです。クラウドでのミュータブルなアプローチの例には、アプリケーションの変更を、手動、あるいは、AWS Systems Manager Run Command や AWS CodeDeploy などのソフトウェアデプロイメントサービスを使用して、稼働中の EC2 インスタンスに直接デプロイする方法があります。

イミュータブルなインフラストラクチャモデルでは、既存のインフラストラクチャに更新、パッチ適用、変更などを行わず、本番環境のワークロード向けに新しいインフラストラクチャをデプロイします。イミュータブルなアプローチの例として、AWS CloudFormation または AWS Cloud Development Kit (AWS CDK) でアプリケーションスタックを定義することが挙げられます。こうしたサービスを使用すると、継続的インテグレーションおよび継続的デリバリー (CI/CD) パイプラインを通じてアプリケーションスタックをデプロイできます。このアプローチでは、ローリングデプロイやブルー/グリーンデプロイなどのデプロイ手法を使用します。このアプローチの詳細については、AWS Well-Architected フレームワークにある「イミュータブルな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。

例えば、オペレーションシステムへのパッチ適用の Essential Eight 戦略をクラウドに適応させる場合は、パッチ適用をデプロイモデルにどのように当てはめるかを考察する必要があります。ミュータブルなインフラストラクチャでは、リソースに手動でパッチを適用することも、自動化によって運用を効率化することもできます。イミュータブルなインフラストラクチャを採用している場合は、CI/

CD パイプラインを通じて、最新のオペレーティングシステムバージョンを使用し、新しいインフラストラクチャをデプロイします。厳密に言うと、このモデルでは、パッチ適用という呼び方は誤っています。パッチ適用ではなく、置き換えによってインフラストラクチャを更新するからです。

テーマ 1: マネージドサービスの使用

> **Essential Eight 戦略の対象**
>
> アプリケーションへのパッチ適用、管理権限の制限、オペレーティングシステムへのパッチ適用

マネージドサービスは、 AWS がパッチ適用や脆弱性管理などの一部のセキュリティタスクを管理できるようにすることで、コンプライアンス義務の軽減に役立ちます。

AWS 責任共有モデル セクションで説明したように、クラウドのセキュリティとコンプライアンス AWS の責任は と共有します。これにより、 はホストオペレーティングシステムや仮想化レイヤーから、サービスが動作する施設の物理的なセキュリティまで、コンポーネントを AWS 運用、管理、制御するため、運用上の負担を軽減できます。

お客様の責任には、Amazon Relational Database Service (Amazon RDS) や Amazon Redshift などのマネージドサービスのメンテナンスウィンドウの管理、 AWS Lambda コードやコンテナイメージの脆弱性のスキャンが含まれます。また、これは、本ガイドのどのテーマにも当てはまる運用ですが、モニタリングとコンプライアンスレポートもお客様の責任で行う必要があります。Amazon Inspector を使用すると、すべての AWS アカウントに存在する脆弱性をレポートでき、のルールを使用して、Amazon RDS や Amazon Redshift などのサービスでマイナーな更新とメンテナンスウィンドウが有効になってい AWS Config ることを確認できます。

例えば、Amazon EC2 インスタンスが稼働している場合、以下については、お客様の責任で実施します。

- アプリケーション制御
- アプリケーションへのパッチ適用
- Amazon EC2 のコントロールプレーンおよびオペレーティングシステム (OS) に対する管理者権限の制限
- OS へのパッチ適用
- AWS コントロールプレーンと OS にアクセスするための多要素認証 (MFA) の強制
- データおよび設定のバックアップ

一方、Lambda 関数を実行している場合、責任は軽減され、以下がお客様側での運用対象となります。

- アプリケーション制御
- ライブラリの最新状態の維持
- Lambda コントロールプレーンに対する理者権限の制限
- AWS コントロールプレーンへのアクセスを MFA に強制する
- Lambda 関数のコードおよび設定のバックアップ

AWS Well-Architected フレームワークの関連するベストプラクティス

- SEC01-BP05 セキュリティ管理の範囲を縮小する

このテーマの実装

パッチ適用の有効化

- Amazon RDS の更新を適用する
- でマネージド更新を有効にする AWS Elastic Beanstalk
- Amazon Redshift クラスターのメンテナンスウィンドウの認識

脆弱性のスキャン

- Amazon Inspector による、Amazon Elastic Container Registry (Amazon ECR) コンテナイメージのスキャン
- Amazon Inspector による、Lambda 関数のスキャン

このテーマのモニタリング

ガバナンスチェックの実装

- で ACSC Essential 8 コンフォーマンスパックの運用上のベストプラクティスを有効にする AWS Config

Amazon Inspector のモニタリング

- アカウントレベルのカバレッジ評価
- 複数アカウントの管理

次の AWS Config ルールを実装する

- RDS_AUTOMATIC_MINOR_VERSION_UPGRADE_ENABLED
- ELASTIC_BEANSTALK_MANAGED_UPDATES_ENABLED
- REDSHIFT_CLUSTER_MAINTENANCESETTINGS_CHECK
- EC2_MANAGEDINSTANCE_PATCH_COMPLIANCE_STATUS_CHECK
- EKS_CLUSTER_SUPPORTED_VERSION

テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理

> **Essential Eight 戦略の対象**
>
> アプリケーション制御、アプリケーションへのパッチ適用、オペレーティングシステムへのパッチ適用

イミュータブルインフラストラクチャの場合は、システム変更のデプロイパイプラインを保護する必要があります。 AWS 著名エンジニアである Colm MacC árthaigh は、2022 AWS re:Invent カンファレンスの「ゼロ特権オペレーション: データにアクセスできないサービスの実行YouTube」プレゼンテーションでこの原則について説明しました。

AWS リソースを設定するための直接アクセスを制限することで、承認済み、セキュア、自動化されたパイプラインを通じてすべてのリソースをデプロイまたは変更することを要求できます。具体的には、通常、AWS Identity and Access Management (IAM) ポリシーを作成し、ユーザーがデプロイパイプラインをホストするアカウントにのみアクセスできるようにし、限られた数のユーザーにブレークグラスアクセスを許可する IAM ポリシーも設定します。また、手動の変更を防ぐために、セキュリティグループを使用して、SSH および Windows リモートデスクトッププロトコル (RDP) によるサーバーアクセスをブロックできます。の一機能である Session Manager は AWS Systems Manager、インバウンドポートを開いたり踏み台ホストを維持したりすることなく、インスタンスへのアクセスを提供できます。

Amazon マシンイメージ (AMI) とコンテナイメージは、安全で反復可能な方法で構築しなければなりません。Amazon EC2 インスタンスの場合、EC2 Image Builder を使用して、インスタンス検出、アプリケーション制御、ログ記録などのセキュリティ機能を組み込んだ AMI を構築できます。アプリケーション制御の詳細については、ACSC ウェブサイトの「Implementing Application Control」を参照してください。また、Image Builder でコンテナイメージを構築して、Amazon Elastic Container Registry (Amazon ECR) を使用し、それらのイメージをアカウント間で共有することも可能です。中央のセキュリティチームでは、こうした AMI とコンテナイメージを構築する自動プロセスを承認でき、これによって、作成された AMI またはコンテナイメージをアプリケーションチームで使用することが承認されます。

アプリケーションは、AWS CloudFormation や AWS Cloud Development Kit (AWS CDK) などのサービスを使用して、Infrastructure as Code (IaC) で定義する必要があります。cfn-nag AWS

CloudFormation Guardや cdk-nag などのコード分析ツールは、承認されたパイプラインのセキュリティのベストプラクティスに照らしてコードを自動的にテストできます。

テーマ 1: マネージドサービスの使用 と同様に、Amazon Inspector を使用すると、 AWS アカウント全体の脆弱性をレポートできます。一元化されたクラウドおよびセキュリティチームでは、こうした情報を使用して、アプリケーションチームがセキュリティおよびコンプライアンス要件を満たしていることを確認できます。

コンプライアンスをモニタリングし報告するには、IAM リソースとログを継続的にレビューします。 AWS Config ルールを使用して、承認された AMIs のみを使用し、Amazon Inspector が Amazon ECR リソースの脆弱性をスキャンするように設定されていることを確認します。

AWS Well-Architected フレームワークの関連するベストプラクティス

- OPS05-BP04 構築およびデプロイ管理システムを使用する
- REL08-BP04 イミュータブルなインフラストラクチャを使用してデプロイする
- SEC06-BP03 手動管理とインタラクティブアクセスを削減する

このテーマの実装

AMI およびコンテナビルドパイプラインの実装

- EC2 Image Builder を使用して、AMI に以下を組み込みます。
  - インスタンスの検出と管理に使用される AWS Systems Manager エージェント (SSM エージェント)
  - Security Enhanced Linux (SELinux) (GitHub)、ファイルアクセスポリシーデーモン (fapolicyd) (GitHub)、OpenSCAP などのアプリケーション制御用のセキュリティツール
  - Amazon CloudWatch エージェント (ログ記録に使用する)
- すべての EC2 インスタンスを対象に、Systems Manager がインスタンスアクセスに使用するインスタンスプロファイルまたは IAM ロールに、`CloudWatchAgentServerPolicy` および `AmazonSSMManagedInstanceCore` ポリシーを組み込みます。
- 組織全体と AMI を共有する
- EC2 Image Builder リソースを共有する
- アプリケーションチームで、最新の AMI が参照されていることを確認する

- パッチ管理に AMI パイプラインを使用する
- コンテナビルドパイプラインを実装します。
  - EC2 Image Builder コンソールウィザードを使用してコンテナイメージパイプラインを作成する
  - Amazon ECR をソースとして使用してコンテナイメージの継続的な配信パイプラインを構築する (AWS ブログ記事)
- マルチアカウントおよびマルチリージョンアーキテクチャを使用して、組織全体で ECR コンテナイメージを共有する

安全なアプリケーションビルドパイプラインの実装

- EC2 Image Builder や AWS CodePipeline を使用するなど、IaC のビルドパイプラインを実装する (AWS ブログ記事)
- CI/CD パイプラインで AWS CloudFormation Guard、cfn-nag (GitHub)、cdk-nag (GitHub) などのコード分析ツールを使用して、次のようなベストプラクティス違反を検出します。
  - ワイルドカードを使用するポリシーなど、許可範囲が広すぎる IAM ポリシー
  - ワイルドカードを使用したり、SSH アクセスを許可したりするなど、許可範囲が広すぎるセキュリティグループルール
  - 有効になっていないアクセスログ
  - 有効になっていない暗号化
  - パスワードの直接的な記述
- パイプラインにスキャンツールを実装する (AWS ブログ記事)
- パイプライン AWS Identity and Access Management Access Analyzer (ブログ記事) で を使用して、CloudFormation テンプレートで定義されている IAM ポリシーを検証するAWS
- IAM ポリシーとサービス制御ポリシーを設定して、パイプラインの使用や変更のためのアクセス時に最小特権が付与されるようにする

脆弱性スキャンの実装

- 組織内のすべてのアカウントで Amazon Inspector を有効にする
- Amazon Inspector を使用して、AMI ビルドパイプライン内の AMI をスキャンします。
  - EC2 Image Builder (GitHub) で AMI のライフサイクルを管理する
- Amazon Inspector を使用して、Amazon ECR リポジトリの拡張スキャンを設定する

- セキュリティ関連の検出結果をトリアージして修正する脆弱性管理プログラムを構築する

このテーマのモニタリング

IAM とログの継続的なモニタリング

- IAM ポリシーを定期的に見直し、以下を確認します。
  - デプロイパイプラインのみがリソースに直接アクセスできる
  - 承認済みサービスのみがデータに直接アクセスできる
  - ユーザーは、リソースまたはデータに直接アクセスできない
- AWS CloudTrail ログをモニタリングして、ユーザーがパイプラインを介してリソースを変更しており、リソースを直接変更したりデータにアクセスしたりしていないことを確認します。
- IAM Access Analyzer の検出結果を定期的に確認する
- AWS アカウント のルートユーザー認証情報が使用された場合にその旨が通知されるアラートを設定する

次の AWS Config ルールを実装する

- `APPROVED_AMIS_BY_ID`
- `APPROVED_AMIS_BY_TAG`
- `ECR_PRIVATE_IMAGE_SCANNING_ENABLED`

テーマ 3: 自動化によるミュータブルなインフラストラクチャの管理

> **Essential Eight 戦略の対象**
>
> アプリケーション制御、アプリケーションへのパッチ適用、オペレーティングシステムへのパッチ適用

イミュータブルなインフラストラクチャと同様に、ミュータブルなインフラストラクチャも、IaC として管理し、自動プロセスを通じて変更または更新します。イミュータブルなインフラストラクチャを実装する手順の多くは、ミュータブルなインフラストラクチャにも当てはまりますが、ミュータブルなインフラストラクチャでは、手動制御も実装して、変更済みワークロードがベストプラクティスに準拠している状態を維持しなければなりません

変更可能なインフラストラクチャでは、 の一機能である Patch Manager を使用してパッチ管理を自動化できます AWS Systems Manager。Patch Manager は、 AWS 組織内のすべてのアカウントで有効にしてください。

また、SSH および RDP による直接アクセスを防止し、ユーザーに Systems Manager の機能でもある Session Manager または Run Command の使用を求めます。SSH や RDP とは異なり、これらの機能では、システムへのアクセスおよび変更が口グに記録されます。

コンプライアンスをモニタリングし報告するには、パッチ適用上のコンプライアンスを継続的にレビューする必要があります。 AWS Config ルールを使用して、すべての Amazon EC2 インスタンスが Systems Manager によって管理され、必要なアクセス許可とインストールされたアプリケーションがあり、パッチに準拠していることを確認できます。

AWS Well-Architected フレームワークの関連するベストプラクティス

- SEC06-BP03 手動管理とインタラクティブアクセスを削減する
- SEC06-BP05 コンピューティング保護を自動化する

このテーマの実装

パッチ適用を自動化する

- AWS 組織内のすべてのアカウントで Patch Manager を有効にする手順を実行する
- すべての EC2 インスタンスのインスタンスプロファイルまたは IAM ロールに、`CloudWatchAgentServerPolicy` と `AmazonSSMManagedInstanceCore` を追加します。これらは、Systems Manager によるインスタンスアクセスに使用されます。

手動ではなく、自動プロセスを使用する

- テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理 の「AMI およびコンテナビルドパイプラインの実装」のガイダンスを実行する
- SSH または RDP による直接アクセスではなく、Session Manager または Run Command を使用する

自動化によって EC2 インスタンスに以下をインストールする

- インスタンスの検出と管理に使用される AWS Systems Manager エージェント (SSM エージェント)
- Security Enhanced Linux (SELinux) (GitHub)、ファイルアクセスポリシーデーモン (fapolicyd) (GitHub)、OpenSCAP などのアプリケーション制御用のセキュリティツール
- Amazon CloudWatch エージェント (ログ記録に使用する)

リリース前にピアレビューを行い、変更がベストプラクティスに準拠していることを確認します。

- ワイルドカードを使用するポリシーなど、許可範囲が広すぎる IAM ポリシー
- ワイルドカードを使用したり、SSH アクセスを許可したりするなど、許可範囲が広すぎるセキュリティグループルール
- 有効になっていないアクセスログ
- 有効になっていない暗号化
- パスワードの直接的な記述

- IAM ポリシーのセキュリティ強化

ID レベルの制御を使用する

- ユーザーに自動プロセスでのリソース変更を求め、手動設定を防ぐには、ユーザーが引き受けられるロールに読み取り専用アクセス許可を付与する
- Systems Manager によって使用されるロールなど、サービスロールにのみ、リソース変更に必要なアクセス許可を付与する

脆弱性スキャンの実装

- テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理 の「脆弱性スキャンの実装」に記載のガイダンスを実装する
- Amazon Inspector を使用して EC2 インスタンスをスキャンする

このテーマのモニタリング

パッチ適用上のコンプライアンスを継続的にモニタリングする

- 自動化とダッシュボードを使用して、パッチ適用上のコンプライアンスを報告する
- ダッシュボードでパッチ適用上のコンプライアンスを確認する仕組みを実装する

IAM とログの継続的なモニタリング

- IAM ポリシーを定期的に見直し、以下を確認します。
  - デプロイパイプラインのみがリソースに直接アクセスできる
  - 承認済みサービスのみがデータに直接アクセスできる
  - ユーザーは、リソースまたはデータに直接アクセスできない
- AWS CloudTrail ログをモニタリングして、ユーザーがパイプラインを介してリソースを変更しており、リソースを直接変更したりデータにアクセスしたりしていないことを確認します。
- 検出 AWS Identity and Access Management Access Analyzer 結果を定期的に確認する
- AWS アカウント のルートユーザー認証情報が使用された場合にその旨が通知されるアラートを設定する

次の AWS Config ルールを実装する

- EC2_MANAGEDINSTANCE_PATCH_COMPLIANCE_STATUS_CHECK
- EC2_INSTANCE_MANAGED_BY_SSM
- EC2_MANAGEDINSTANCE_APPLICATIONS_REQUIRED - SELinux/fapolicyd/OpenSCAP, CW Agent
- EC2_MANAGEDINSTANCE_APPLICATIONS_BLACKLISTED - any unsupported apps
- IAM_ROLE_MANAGED_POLICY_CHECK - CW Logs, SSM
- EC2_MANAGEDINSTANCE_ASSOCIATION_COMPLIANCE_STATUS_CHECK
- REQUIRED_TAGS
- RESTRICTED_INCOMING_TRAFFIC - 22, 3389

テーマ 4: ID の管理

Essential Eight 戦略の対象

管理者権限の制限、多要素認証

ID およびアクセス許可の堅固な管理は、クラウドでセキュリティを管理する上で、きわめて重要です。強力な ID 管理プラクティスを実装すると、必要なアクセス権限と最小特権のバランスが取れるため、開発チームが、セキュリティを損なわずに、作業を迅速化できるようになります。

ID フェデレーションを使用して、ID 管理を一元化します。これによって、アクセス権限を一元管理することで、複数のアプリケーションやサービス全体で、アクセス管理が容易になります。また、一時的な権限や多要素認証 (MFA) も実装しやすくなります。

ユーザーには、タスク実行に必要なアクセス権限のみを付与してください。 AWS Identity and Access Management Access Analyzer を使用すると、ポリシーの検証や、パブリックアクセスおよびクロスアカウントアクセスの確認を行えます。 AWS Organizations サービスコントロールポリシー (SCPs)、IAM ポリシー条件、IAM アクセス許可の境界、 AWS IAM アイデンティティセンターアクセス許可セットなどの機能は、きめ細かなアクセスコントロール (FGAC) の設定に役立ちます。

認証の種類にかかわらず、一時的な認証情報の使用を強くお勧めします。これにより、認証情報が誤って開示または共有されたり、盗まれたりするなどのリスクを軽減または排除します。IAM ユーザーではなく、IAM ロールを使用してください。

強力なサインインの仕組み (MFA など) を使用すると、サインインの認証情報が誤って開示されたり、簡単に推測されたりするリスクを軽減できます。ルートユーザーには MFA を求めます。MFA は、フェデレーションレベルで要求することも可能です。どうしても、IAM ユーザーの使用が必要な場合は、MFA を強制します。

コンプライアンスをモニタリングし報告するには、アクセス許可の継続的な削減、IAM Access Analyzer から取得した検出結果のモニタリング、使用されていない IAM リソースの削除などを行う必要があります。 AWS Config ルールを使用して、強力なサインインメカニズムが適用され、認証情報の有効期限が短く、IAM リソースが使用されていることを確認します。

AWS Well-Architected フレームワークの関連するベストプラクティス

- SEC02-BP01 強力なサインインメカニズムを使用する
- SEC02-BP02 一時的な認証情報を使用する
- SEC02-BP03 シークレットを安全に保存して使用する
- SEC02-BP04 一元化された ID プロバイダーを利用する
- SEC02-BP05 定期的に認証情報を監査およびローテーションする
- SEC02-BP06 ユーザーグループと属性を採用する
- SEC03-BP01 アクセス要件を定義する
- SEC03-BP02 最小特権のアクセスを付与する
- SEC03-BP03 緊急アクセスプロセスを確立する
- SEC03-BP04 アクセス許可を継続的に削減する
- SEC03-BP05 組織のアクセス許可ガードレールを定義する
- SEC03-BP06 ライフサイクルに基づいてアクセスを管理する
- SEC03-BP07 パブリックおよびクロスアカウントアクセスの分析
- SEC03-BP08 組織内でリソースを安全に共有する

このテーマの実装

ID フェデレーションの実装

- 一時的な認証情報を使用して AWS にアクセスする人間のユーザーには、ID プロバイダーとのフェデレーションを求めている
- AWS 環境への一時的な昇格アクセスを実装する

最小特権アクセス許可を適用する

- ルートユーザーの認証情報を保護し、日常的なタスクには使用しない
- IAM Access Analyzer を使用して、アクセスアクティビティに基づいて最小特権ポリシーを生成する

- IAM Access Analyzer を使用してリソースへのパブリックアクセスとクロスアカウントアクセスを検証する
- IAM Access Analyzer を使用して IAM ポリシーを検証し、安全で機能的なアクセス許可を行う
- 複数のアカウントでアクセス許可ガードレールを確立する
- アクセス許可の境界を使用して、ID ベースのポリシーで付与可能な権限の上限を設定する
- IAM ポリシーの条件を使用してアクセスをさらに制限する
- 未使用のユーザー、ロール、アクセス許可、ポリシー、認証情報を定期的に確認して削除する
- AWS 管理ポリシーの使用を開始し、最小特権のアクセス許可に移行する
- IAM Identity Center のアクセス許可セット機能を使用する

認証情報のローテーション

- ワークロードが にアクセスするために IAM ロールを使用するよう要求する AWS
- 使用されていない IAM ロールの削除を自動化する
- 長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションする

MFA の強制

- ルートユーザーに MFA を求める
- IAM Identity Center を通じて MFA を要求する
- サービス固有の API アクションに MFA を要求することを検討する

このテーマのモニタリング

最小特権アクセスのモニタリング

- IAM Access Analyzer の検出結果を に送信する AWS Security Hub CSPM
- 重要な IAM Identity Center から得た重要な検出結果の通知設定を検討する
- の認証情報レポートを定期的に確認する AWS アカウント

次の AWS Config ルールを実装する

- ACCESS_KEYS_ROTATED
- IAM_ROOT_ACCESS_KEY_CHECK
- IAM_USER_MFA_ENABLED
- IAM_USER_UNUSED_CREDENTIALS_CHECK
- IAM_PASSWORD_POLICY
- ROOT_ACCOUNT_HARDWARE_MFA_ENABLED

テーマ 5: データ境界を確立する

Essential Eight 戦略の対象

管理者権限の制限

データ境界 とは、 AWS 環境における予防的ガードレールのセットであり、これによって、想定ネットワークをアクセス元とする信頼できる ID のみ、信頼できるリソースにアクセスできるようにします。これらのガードレールは、 AWS アカウント および リソースの幅広いセットにわたってデータを保護するのに役立つ常時オンの境界として機能します。しかし、組織全体をカバーするこうしたガードレールは、既存のきめ細かなアクセスコントロールに代わるものではなく、代わりに、すべての AWS Identity and Access Management (IAM) ユーザー、ロール、リソースが定義された一連のセキュリティ標準に準拠していることを確認することで、セキュリティ戦略の改善に役立ちます。

データ境界は、組織境界外からのアクセスを防止するポリシー (一般的に、 AWS Organizationsで作成) を使用して確立できます。データ境界の確立に使用する主な境界認可条件には、以下の 3 つがあります。

- 信頼できる ID – 内のプリンシパル (IAM ロールまたはユーザー) AWS アカウント、またはユーザーに代わって AWS のサービス 行動するプリンシパル。
- 信頼できるリソース – 内のリソース、 AWS アカウント またはユーザーに代わって AWS のサービス 管理されるリソース。
- 予想されるネットワーク — オンプレミスのデータセンターと仮想プライベートクラウド (VPCsまたはユーザーに代わって AWS のサービス 動作するネットワーク。

`OFFICIAL:SENSITIVE` または `PROTECTED` といった異なるデータ分類の環境間、あるいは、開発、テスト、本番といった異なるリスクレベルの環境間にデータ境界を実装することを検討してください。詳細については、「 でのデータ境界 AWSの構築 (AWS ホワイトペーパー) 」および「 でのデータ境界の確立 AWS: 概要 (AWS ブログ記事) 」を参照してください。

AWS Well-Architected フレームワークの関連するベストプラクティス

- SEC03-BP05 組織のアクセス許可ガードレールを定義する

- SEC07-BP02 データ機密性に基づいてデータ保護コントロールを適用する

このテーマの実装

ID コントロールを実装する

- 信頼できる ID にのみリソースへのアクセスを許可する – 条件キー `aws:PrincipalOrgID` および `aws:PrincipalIsAWSService` を指定したリソースベースのポリシーを使用します。これにより、 AWS 組織と のプリンシパルのみが AWS リソースにアクセスできるようになります。
- お客様ネットワークを送信元とする信頼できる ID にのみアクセスを許可する – 条件キー `aws:PrincipalOrgID` および `aws:PrincipalIsAWSService` を指定した VPC エンドポイントポリシーを使用します。これにより、 AWS 組織と のプリンシパルのみが VPC AWS エンドポイントを介してサービスにアクセスできます。

リソースコントロールを実装する

- ID に、信頼できるリソースのみへのアクセスを許可する – 条件キー `aws:ResourceOrgID` を指定したサービスコントロールポリシー (SCP) を使用します。これにより、ID は AWS 組織内のリソースにのみアクセスできます。
- お客様ネットワークを送信元とする場合にのみ信頼できるリソースへのアクセスを許可する – 条件キー `aws:ResourceOrgID` を指定した VPC エンドポイントポリシーを使用します。これにより、この ID は、 AWS 組織に属する VPC エンドポイントを介してのみサービスにアクセスできます。

ネットワークコントロールを実装する

- 想定ネットワークを送信元とする ID にのみリソースへのアクセスを許可する – 条件キー `aws:SourceIp`、`aws:SourceVpc`、`aws:SourceVpce`、`aws:ViaAWSService` を指定した SCP を使用します。これにより、ID は、予想される IP アドレス、VPCs、VPC エンドポイントから、および を介してのみリソースにアクセスできます AWS のサービス。
- 想定ネットワークを送信元とする場合にのみお客様リソースへのアクセスを許可する – 条件キー `aws:SourceIp`、`aws:SourceVpc`、`aws:SourceVpce`、`aws:ViaAWSService`、`aws:PrincipalIsA` を指定した、リソースベースのポリシーを使用します。これにより、リソースへのアクセスは、予想される IPs、予想される VPCs、予想される VPC エンドポイント、 から、 AWS のサービスまたは呼び出し元の ID が である場合にのみ許可されます AWS のサービス。

このテーマのモニタリング

ポリシーをモニタリング

- SCP、IAM ポリシー、VPC エンドポイントポリシーを確認する仕組みを実装する

次の AWS Config ルールを実装する

- `SERVICE_VPC_ENDPOINT_ENABLED`

テーマ 6: バックアップの自動化

Essential Eight 戦略の対象

定期バックアップ

「障害は避けられないものであり、時間が経つにつれ、あらゆる要素に故障が生じます。これは、ルーターからハードディスク、オペレーティングシステムからメモリユニット、破損した TCP パケットに至るどの要素にも当てはまり、一時的なエラーから恒久的な障害まで、規模もさまざまです。最高品質のハードウェア、あるいは最低価格のコンポーネントを使用していても、避けられないのです」 —Amazon CTO、ワーナー ヴォゲルス、「All Things Distributed」

データのバックアップと復旧は、システムの信頼性にとって重要な部分です。 AWS は、バックアップの作成、バックアップデータの耐久性の維持、バックアップデータの復旧を容易にするように設計されています。

AWS Backup は、フルマネージドサービスであり、これを利用すると、 AWS のサービスサービス全体のデータバックアップを一元管理および自動化できます。複数の AWS リソースタイプをサポートし、まとめてバックアップする必要がある複数の AWS リソースを使用するワークロードのバックアップ戦略を実装および維持するのに役立ちます。 AWS Backup また、 は、複数の AWS リソースのバックアップおよび復元オペレーションをまとめてモニタリングするのに役立ちます。

AWS Backup ボールトロックはバックアップボールトのオプション機能であり、セキュリティと制御を強化できます。コンプライアンスモードで、ロックが有効になっているときに、猶予期間が終了すると、ユーザー、アカウント、データ所有者、 AWSによる、ボールト設定の変更または削除が行えなくなります。各ボールトには 1 つのボールトロックを設定できます。これにより、Write-Once、Read-Many (WORM) の設定と、保持期間の適用が可能になります。

現在の設定ガイダンスに従うと、 は 11 nines とも呼ばれる 99.999999999% の年間耐久性を提供 AWS Backup できます。グローバル AWS インフラストラクチャを使用して、複数のアベイラビリティーゾーンにバックアップをレプリケートします。詳細については、「AWS Backupの耐障害性」を参照してください。

AWS Backup は、バックアップされたデータの復旧とテストを自動化して、バックアップの整合性とプロセスを検証するのに役立ちます。

AWS Well-Architected フレームワークの関連するベストプラクティス

- SEC09-BP01 安全なキーと証明書の管理を実装する
- SEC09-BP02 伝送中に暗号化を適用する
- SEC09-BP03 ネットワーク通信を認証する

このテーマの実装

データのバックアップおよび復旧を自動化する

- にデータバックアップを実装する AWS
- 大規模なデータバックアップの自動化 (AWS ブログ記事)
- によるデータ復旧検証の自動化 AWS Backup (AWS ブログ記事)

AWS Backup 成果全体にガバナンスを実装する

- 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 ブログ記事)
- AWS Backup ボールトロックを使用してバックアップボールトのセキュリティを向上させる
- AWS Backup Audit Manager を使用して、 AWS Backup ポリシーのコンプライアンスを監査する

このテーマのモニタリング

次の AWS Config ルールを実装する

- RDS_IN_BACKUP_PLAN
- RDS_LAST_BACKUP_RECOVERY_POINT_CREATED
- RDS_RESOURCES_PROTECTED_BY_BACKUP_PLAN
- REDSHIFT_BACKUP_ENABLED
- AURORA_LAST_BACKUP_RECOVERY_POINT_CREATED
- AURORA_RESOURCES_PROTECTED_BY_BACKUP_PLAN

- BACKUP_PLAN_MIN_FREQUENCY_AND_MIN_RETENTION_CHECK
- BACKUP_RECOVERY_POINT_ENCRYPTED
- BACKUP_RECOVERY_POINT_MANUAL_DELETION_DISABLED
- BACKUP_RECOVERY_POINT_MINIMUM_RETENTION_CHECK
- DB_INSTANCE_BACKUP_ENABLED
- DYNAMODB_IN_BACKUP_PLAN
- DYNAMODB_LAST_BACKUP_RECOVERY_POINT_CREATED
- DYNAMODB_RESOURCES_PROTECTED_BY_BACKUP_PLAN
- EBS_IN_BACKUP_PLAN
- EBS_LAST_BACKUP_RECOVERY_POINT_CREATED
- EBS_RESOURCES_PROTECTED_BY_BACKUP_PLAN
- EC2_LAST_BACKUP_RECOVERY_POINT_CREATED
- S3_LAST_BACKUP_RECOVERY_POINT_CREATED
- S3_RESOURCES_PROTECTED_BY_BACKUP_PLAN
- STORAGEGATEWAY_LAST_BACKUP_RECOVERY_POINT_CREATED
- STORAGEGATEWAY_RESOURCES_PROTECTED_BY_BACKUP_PLAN
- VIRTUALMACHINE_LAST_BACKUP_RECOVERY_POINT_CREATED
- VIRTUALMACHINE_RESOURCES_PROTECTED_BY_BACKUP_PLAN

テーマ 7: ログ記録およびモニタリングの一元化

Essential Eight 戦略の対象

アプリケーション制御、アプリケーションへのパッチ適用、管理権限の制限、多要素認証

AWS には、 環境で何が起こっているかを確認できるようにするツールと機能が用意されています AWS 。具体的には次のとおりです。

- AWS CloudTrail は、、SDK、コマンドラインツールを通じて行われた AWS API コールなど AWS マネジメントコンソール、アカウントの API コールの履歴証跡を作成することで、 AWS デプロイをモニタリングするのに役立ちます。 AWS SDKs CloudTrail に対応したサービスでは、サービスの API を呼び出したユーザーおよびアカウント、呼び出し元の IP アドレス、呼び出しの発生時刻も特定できます。
- Amazon CloudWatch は、 AWS リソースと で実行するアプリケーションのメトリクスを AWS リアルタイムでモニタリングするのに役立ちます。
- Amazon CloudWatch Logs は、すべてのシステム、アプリケーション、 AWS のサービス からのログを一元化するのに役立ちます。一元化により、ログを監視して安全にアーカイブできます。
- Amazon GuardDuty は、継続的なセキュリティモニタリングサービスであり、これを利用すると、ログを分析および処理し、 AWS 環境における予期しないアクティビティや、不正の可能性があるアクティビティを特定できます。GuardDuty は、自動応答や、担当者への通知を行えるよう、Amazon EventBridge と統合されています。
- AWS Security Hub CSPM は、 のセキュリティ状態の包括的なビューを提供します AWS。また、セキュリティ業界標準とベストプラクティスに照らして AWS環境を確認するのにも役立ちます。

こうしたツールと機能は、可視性を高め、環境に悪影響が出る前に問題に対処できるよう設計されています。これにより、クラウドにおける組織のセキュリティ体制を強化し、環境のリスクプロファイルを減らすことができます。

AWS Well-Architected フレームワークの関連するベストプラクティス

- SEC04-BP01 サービスとアプリケーションの口グ記録を設定する

- SEC04-BP02 標準化された場所でログ、検出結果、メトリクスをキャプチャする

このテーマの実装

ログ記録の有効化

- CloudWatch エージェントを使用して、システムレベルのログを CloudWatch Logs に発行する
- GuardDuty による検出結果にアラートを設定する
- CloudTrail で組織の証跡を作成する

ログ記録のベストプラクティスを実装する

- CloudTrail セキュリティのベストプラクティスを実装する
- SCPs を使用して、ユーザーがセキュリティサービスを無効にできないようにする (AWS ブログ記事)
- を使用して CloudWatch Logs のログデータを暗号化する AWS Key Management Service

ログを一元管理する

- 複数のアカウントから CloudTrail ログを受信する
- ログアーカイブアカウントにログを送信する
- 監査と分析のために CloudWatch Logs をアカウントで一元管理する (AWS ブログ記事)
- Amazon Inspector の管理を一元化する
- で組織全体のアグリゲータを作成する AWS Config (AWS ブログ記事)
- Security Hub CSPM の管理を一元化する
- GuardDuty の管理を一元化する
- Amazon Security Lake の使用を検討する

このテーマのモニタリング

モニタリングの仕組みを実装する

- ログによる検出結果を確認する仕組みを確立する

- Security Hub CSPM の検出結果を確認するメカニズムを確立する
- GuardDuty による検出結果に対応する仕組みを確立する

次の AWS Config ルールを実装する

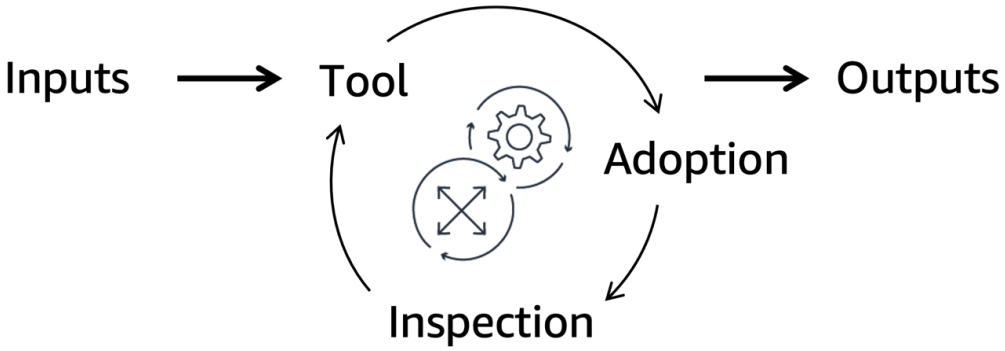
- `CLOUDTRAIL_SECURITY_TRAIL_ENABLED`
- `GUARDDUTY_ENABLED_CENTRALIZED`
- `SECURITYHUB_ENABLED`
- `ACCOUNT_PART_OF_ORGANIZATIONS`

テーマ 8: 手動プロセスの仕組みの実装

> **Essential Eight 戦略の対象**
>
> アプリケーション制御、アプリケーションへのパッチ適用

Amazon では、良い意図は機能しません。メカニズムは機能します (AWS ブログ記事) 。望ましい結果を実現するために必要なのは、最善の努力ではなく、自動化され、反復可能で、スケーラブルなプロセスとツールなのです。

次の図に示すように、仕組みとは包括的なプロセスであり、これに従って、ツールを作成し、導入を促進します。その後、結果を検査し、それに応じた調整を行います。この仕組みは、運用しながらそれ自体を強化し改善していくサイクルでもあります。このサイクルでは、制御可能な入力を受け取り、それを継続的な出力に変換することで、繰り返し発生するビジネス課題に対処します。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。



AWS Well-Architected フレームワークの関連するベストプラクティス

- OPS02-BP01 リソースには特定の所有者が存在する
- OPS02-BP02 プロセスと手順に特定の所有者が存在する
- OPS02-BP03 パフォーマンスに責任を持つ所有者が運用アクティビティに存在する
- OPS02-BP04 責任と所有権を管理するためのメカニズムが存在する
- OPS03-BP01 エグゼクティブスポンサーシップを提供する
- OPS03-BP03 エスカレーションが推奨されている

このテーマの実装

- コンプライアンス上の不備を確認し対処する仕組みを確立する
- セキュリティポリシー更新の仕組みを確立する
- サポート対象外のアプリケーションを削除し、 AWS Config ルールの拒否リストに追加する
- を使用してアクセスポリシーを検証する AWS Identity and Access Management Access Analyzer
- Amazon Inspector を有効にして、脆弱性登録を自動的に最新状態に保つ
- アプリケーション制御のルールセットを少なくとも毎年確認する
- 手動プロセスの負担を軽減するために、AWS Config ルールといった自動化の実装を検討する
- ソフトウェアポリシーに必要なソフトウェアが稼働しているインスタンスを可視化するために、AWS Systems Manager Inventory の使用を検討する

このテーマのモニタリング

- エグゼクティブスポンサーの監督体制を確立して、コンプライアンスの確保、不備の検査、仕組みの評価などを行い、目標達成の進捗状況を追跡します。

で Essential Eight 成熟度を達成するための示唆的なケーススタディ AWS

この章では、 AWSで Essential Eight の成熟度到達を目指す政府機関の示唆的な導入事例について説明します。

この章のセクション:

- シナリオおよびアーキテクチャの概要
- ワークロードの例: サーバーレスデータレイク
- ワークロードの例: コンテナ化されたウェブサービス
- ワークロードの例: Amazon EC2 で稼働する COTS ソフトウェア

シナリオおよびアーキテクチャの概要

この政府機関は、 AWS クラウドで 3 つのワークロードを運用しています。

- ストレージと抽出、変換、ロード (ETL) オペレーション AWS Lambda に Amazon Simple Storage Service (Amazon S3) を使用するサーバーレスデータレイク
- コンテナ化されたウェブサービス: Amazon Elastic Container Service (Amazon ECS) で稼働させ、Amazon Relational Database Service (Amazon RDS) のデータベースを使用している
- Commercial Off-The-Shelf (COTS) ソフトウェア: Amazon EC2 で稼働させている

クラウドチームは、組織の一元化されたプラットフォームを提供し、 AWS 環境のコアサービスを実行します。クラウドチームは、 AWS 環境にコアサービスを提供します。各ワークロードは、独立したアプリケーションチーム (開発チームまたはデリバリーチームとも呼ばれる) が所有しています。

コアアーキテクチャ

クラウドチームでは、 AWS クラウドで以下の機能を構築済みです。

- ID Microsoft フェデレーションは、Entra ID (以前の Azure Active Directory) インスタンス AWS IAM アイデンティティセンター にリンクします。フェデレーションでは、MFA、ユーザーアカウントの自動有効期限、および AWS Identity and Access Management (IAM) ロールを介した有効期間の短い認証情報の使用が適用されます。

- 一元化され、EC2 Image Builder を使用する AMI パイプラインで、OS とコアアプリケーションにパッチを適用しています。
- Amazon Inspector を利用して、脆弱性を特定しており、セキュリティ関連の検出結果はすべて Amazon GuardDuty に送信し、一元管理しています。
- アプリケーション制御ルールの更新、サイバーセキュリティイベントへの対応、コンプライアンス上の不備確認を行う仕組みが確立されています。
- AWS CloudTrail はログ記録とモニタリングに使用されます。
- セキュリティイベント (ルートユーザーのログインなど) が発生したら、アラートが発行されます。
- SCP および VPC エンドポイントポリシーにより、 AWS 環境のデータ境界を確立しています。
- SCP により、アプリケーションチームが CloudTrail や AWS Configなどのセキュリティおよびログ記録サービスを無効化できないようにしています。
- AWS Config 検出結果は AWS 組織全体から 1 つの に集約され、セキュリティ AWS アカウントが確保されます。
- AWS Config ACSC Essential 8 コンフォーマンスパックは、組織内のすべての AWS アカウントで有効になっています。

ワークロードの例: サーバーレスデータレイク

このワークロードは、テーマ 1: マネージドサービスの使用 の例を示すものです。

データレイクは、ストレージに Amazon S3 を使用し、ETL AWS Lambda に Amazon S3 を使用します。これらのリソースは AWS Cloud Development Kit (AWS CDK) アプリで定義されます。システムへの変更は、 を通じてデプロイされます AWS CodePipeline。このパイプラインの使用は、アプリケーションチームに制限されています。このチームでコードリポジトリにプルリクエストを行う場合は、2 人制が適用されます。

このワークロードに対し、アプリケーションチームは、Essential Eight 戦略に対応できるよう、以下のアクションを実行します。

アプリケーション制御

- GuardDuty の Lambda Protection と Amazon Inspector の Lambda スキャンを有効にします。
- Amazon Inspector の検出結果を検査および管理する仕組みを実装します。

アプリケーションへのパッチ適用

- Amazon Inspector の Lambda スキャンを有効にし、非推奨ライブラリや脆弱なライブラリにアラートを設定します。
- アプリケーションチームは、 AWS Config がアセット検出の AWS リソースを追跡できるようにします。

管理者権限の制限

- 「コアアーキテクチャ」セクションで説明したように、アプリケーションチームでは、デプロイパイプラインの承認ルールによって、本番環境にデプロイする際のアクセスを既に制限しています。
- ともに一元化された ID フェデレーションおよびログ記録ソリューション (「コアアーキテクチャ」セクションを参照) を活用します。
- アプリケーションチームは証 AWS CloudTrail 跡と Amazon CloudWatch フィルターを作成します。
- アプリケーションチームは、CodePipeline デプロイと AWS CloudFormation スタック削除の Amazon Simple Notification Service (Amazon SNS) アラートを設定します。

オペレーティングシステムへのパッチ適用

- Amazon Inspector の Lambda スキャンを有効にし、非推奨ライブラリや脆弱なライブラリにアラートを設定します。

多要素認証

- 一元化された ID フェデレーションソリューション (「コアアーキテクチャ」セクションを参照) を活用します。このソリューションによって、MFA の適用や認証のログ記録を行い、疑わしい MFA イベントが発生した際は、アラートを生成するかそれらに自動的に対応します。

定期バックアップ

- アプリケーションチームは、 AWS CDK アプリケーションや Lambda 関数、設定などのコードをコードリポジトリに保存します。
- バージョニングと Amazon S3 Object Lock を有効にして、オブジェクトの削除や変更を防ぎます。
- データセット全体を別の AWS リージョンに複製せず、Amazon S3 が備える耐久性を活用します。

- アプリケーションチームは、データ主権 AWS リージョン 要件を満たす別の でワークロードのコピーを実行します。Amazon DynamoDB グローバルテーブルと Amazon S3 クロスリージョンレプリケーションを使用して、プライマリリージョンからセカンダリリージョンにデータを自動的にレプリケートします。

ワークロードの例: コンテナ化されたウェブサービス

このワークロードは、テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理 の例を示すものです。

このウェブサービスは Amazon ECS 上で稼働し、Amazon RDS のデータベースが使用されています。アプリケーションチームは CloudFormation 、 テンプレートでこれらのリソースを定義します。コンテナは EC2 Image Builder で構築し、Amazon ECR に保存します。アプリケーションチームは、 を通じてシステムに変更をデプロイします AWS CodePipeline。このパイプラインの使用は、アプリケーションチームに制限されています。このチームでコードリポジトリにプルリクエストを行う場合は、2 人制が適用されます。

このワークロードに対し、アプリケーションチームは、Essential Eight 戦略に対応できるよう、以下のアクションを実行します。

アプリケーション制御

- Amazon Inspector で Amazon ECR コンテナイメージのスキャンを有効にします。
- ファイルアクセスポリシーデーモン (fapolicyd) セキュリティツールを EC2 Image Builder パイプラインに構築します。詳細については、ACSC ウェブサイトの「Implementing Application Control」を参照してください。
- ログの出力が Amazon CloudWatch Logs に記録されるように、Amazon ECS タスク定義を設定します。
- Amazon Inspector の検出結果を検査および管理する仕組みを実装します。

アプリケーションへのパッチ適用

- Amazon Inspector で Amazon ECR コンテナイメージのスキャンを有効にし、非推奨ライブラリや脆弱なライブラリにアラートを設定します。
- Amazon Inspector の検出結果への対応を自動化します。新しいイベントが検出されると、Amazon EventBridge がトリガーとなり、ターゲットのデプロイパイプライン、つまり、CodePipeline が開始されます。

- アプリケーションチームは、 AWS Config がアセット検出の AWS リソースを追跡できるようにします。

管理者権限の制限

- アプリケーションチームでは、デプロイパイプラインの承認ルールによって、本番環境にデプロイする際のアクセスを既に制限しています。
- 一元化した、クラウドチームの ID フェデレーションを活用して、認証情報のローテーションと一元的なログ記録を行います。
- CloudTrail 証跡と CloudWatch フィルターを作成します。
- CodePipeline によるデプロイと CloudFormation スタック削除に Amazon SNS アラートを設定します。

オペレーティングシステムへのパッチ適用

- Amazon Inspector で Amazon ECR コンテナイメージのスキャンを有効にし、OS のパッチ更新にアラートを設定します。
- Amazon Inspector の検出結果への対応を自動化します。新しいイベントが検出されると、EventBridge がトリガーとなり、ターゲットのデプロイパイプライン、つまり、CodePipeline が開始されます。
- Amazon RDS イベント通知をサブスクライブして、更新情報が通知されるようにし、こうした更新を手動で適用するか、Amazon RDS で自動適用するかについて、ビジネス所有者と共に、リスクに基づく判断を下します。
- メンテナンスイベントの影響を軽減するために、Amazon RDS インスタンスをマルチアベイラビリティーゾーンクラスターとして設定します。

多要素認証

- 一元化された ID フェデレーションソリューション (「コアアーキテクチャ」セクションを参照) を活用します。このソリューションによって、MFA の適用や認証のログ記録を行い、疑わしい MFA イベントが発生した際は、アラートを生成するかそれらに自動的に対応します。

定期バックアップ

- アプリケーションチームは、Amazon RDS クラスターのデータのバックアップを自動化 AWS Backup するように を設定します。
- CloudFormation テンプレートをコードリポジトリに保存します。
- アプリケーションチームは、別のリージョンでワークロードのコピーを作成し、自動テストを実行する自動パイプラインを開発します (AWS ブログ記事) 。このパイプラインでは、自動テストの実行後、スタックを破棄します。このパイプラインを 1 か月に 1 回自動的に実行し、復旧手順の有効性を検証します。

ワークロードの例: Amazon EC2 で稼働する COTS ソフトウェア

このワークロードは、テーマ 3: 自動化によるミュータブルなインフラストラクチャの管理 の例を示すものです。

Amazon EC2 で稼働するワークロードは、 AWS マネジメントコンソールを使用して手動で構築したものです。開発者は、EC2 インスタンスにログインしてソフトウェアを更新することで、システムを手動で更新しています。

このワークロードに対し、クラウドチームとアプリケーションチームは、Essential Eight 戦略に対応できるよう、以下のアクションを実行します。

アプリケーション制御

- クラウドチームは、エージェント (SSM AWS Systems Manager エージェント) 、CloudWatch エージェント、SELinux をインストールして設定するように、一元化された AMI パイプラインを設定します。また、作成した AMI を、組織内のすべてのアカウントで共有します。
- クラウドチームは AWS Config ルールを使用して、実行中のすべての EC2 インスタンスが Systems Manager によって管理され、SSM エージェント、CloudWatch エージェント、SELinux がインストールされていることを確認します。
- クラウドチームは、Amazon CloudWatch Logs の出力を、Amazon OpenSearch Service 上で稼働する一元化されたセキュリティ情報およびイベント管理 (SIEM) ソリューションに送信します。
- アプリケーションチームは、、GuardDuty AWS Config、Amazon Inspector からの検出結果を検査および管理するためのメカニズムを実装します。クラウドチームは、アプリケーションチームが見逃した検出結果を特定する独自の仕組みを実装します。脆弱性管理プログラムを作成して検出結果に対処する方法について、詳細なガイダンスを確認したい場合は、「AWSでのスケーラブルな脆弱性管理プログラムの構築」を参照してください。

アプリケーションへのパッチ適用

- アプリケーションチームは、Amazon Inspector の検出結果に基づいてインスタンスにパッチを適用します。
- クラウドチームは、ベース AMI にパッチを適用します。その AMI が変更された場合、アプリケーションチームはアラートを受け取ります。
- アプリケーションチームは、ワークロードに必要なポートでのみトラフィックが許可されるようにセキュリティグループルールを設定し、EC2 インスタンスへの直接アクセスを制限します。
- アプリケーションチームは、個々のインスタンスにログインせず、Patch Manager を使用してインスタンスにパッチを適用します。
- EC2 インスタンスグループで任意のコマンドを実行するために、アプリケーションチームは、Run Command を使用します。
- まれに、アプリケーションチームがインスタンスに直接アクセスしなければならない場合は、Session Manager を使用します。このアクセスアプローチでは、フェデレーティッド ID を使用し、監査目的でセッションアクティビティを記録します。

管理者権限の制限

- アプリケーションチームは、ワークロードに必要なポートでのみトラフィックが許可されるようにセキュリティグループルールを設定します。これにより、Amazon EC2 インスタンスへの直接アクセスを制限し、Session Manager を介した EC2 インスタンスへのアクセスをユーザーに求めます。
- 一元化した、クラウドチームの ID フェデレーションを活用して、認証情報のローテーションと一元的なログ記録を行います。
- CloudTrail 証跡と CloudWatch フィルターを作成します。
- CodePipeline によるデプロイと CloudFormation スタック削除に Amazon SNS アラートを設定します。

オペレーティングシステムへのパッチ適用

- クラウドチームは、ベース AMI にパッチを適用します。その AMI が変更された場合、アプリケーションチームはアラートを受け取ります。アプリケーションチームは、この AMI を使用して新しいインスタンスをデプロイし、Systems Manager の機能であるステートマネージャーを使用して必要なソフトウェアをインストールします。
- アプリケーションチームは、個々のインスタンスにログインせず、Patch Manager を使用してインスタンスにパッチを適用します。

- EC2 インスタンスグループで任意のコマンドを実行するために、アプリケーションチームは、Run Command を使用します。
- まれに、アプリケーションチームが直接アクセスしなければならない場合は、Session Manager を使用します。

多要素認証

- 一元化された ID フェデレーションソリューション (「コアアーキテクチャ」セクションを参照) を活用します。このソリューションによって、MFA の適用や認証の口グ記録を行い、疑わしい MFA イベントが発生した際は、アラートを生成するかそれらに自動的に対応します。

定期バックアップ

- アプリケーションチームは、EC2 インスタンスと Amazon Elastic Block Store (Amazon EBS) ボリュームの AWS Backup プランを作成します。
- アプリケーションチームは、バックアップからの復元を毎月手動で実行する仕組みを実装します。

リソース

AWS ドキュメント

- AWS Security Reference ArchitectureAWS (SRA)
- AWS セキュリティドキュメント
- セキュリティの柱 - AWS Well-Architected フレームワーク

その他の AWS リソース

- AWS クラウドのセキュリティ
- AWS クラウド導入フレームワーク (セキュリティ観点)

Australian Cyber Security Centre のリソース

- Essential Eight Explained
- Essential Eight Maturity Model
- Essential Eight Assessment Process Guide

寄稿者

本ドキュメントの寄稿者は次のとおりです。

- AWS Solutions Architecture、Senior Solutions Architect、James Kingsmill
- AWS Solutions Architecture、Senior Solutions Architect、Chris Harding
- AWS Solutions Architecture、Advisory Solutions Architect、Jess Modini
- AWS Security Assurance、Security Assurance Principal、Justin Bowden
- AWS Solutions Architecture、Senior Solutions Architect、Rob Powell
- AWS Professional Services、Senior Cloud Architect、Tony Mihaljevic
- AWS Global Services Security、Principal Security Advisor、Volker Rath

付録: Essential Eight コントロールのマトリックス

次の表は、Essential Eight 戦略を AWS Well-Architected フレームワークの AWS 実装ガイダンスおよび関連するベストプラクティスにリンクしています。で適用されない Essential Eight コントロールの場合 AWS クラウド、表にはオーストラリアサイバーセキュリティセンター (ACSC) からの追加のガイダンスへのリンクが含まれています。

コントロールマトリックス:

- アプリケーション制御
- アプリケーションへのパッチ適用
- Microsoft Office マクロ設定の構成
- ユーザーアプリケーションの強化
- 管理者権限の制限
- オペレーティングシステムへのパッチ適用
- 多要素認証
- 定期バックアップ

アプリケーション制御

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
アプリケーション制御は、ワークステーションとサーバーに実装するもので、これによって、次の項目の実行を組織の承認済みセットに制限します: 実行可能ファイル、ソフトウェアライブラリ、スクリプト、インストーラ、コンパイル済み	テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理: AMI およびコンテナビルドパイプラインの実装	EC2 Image Builder を使用して、以下を構築する • AWS Systems Manager エージェント (SSM エージェント) • Security Enhanced Linux (SELinux) (GitHub)、ファイルアクセスポリシー	SEC06-BP02 強化イメージからコンピューティングをプロビジョニングする

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
HTML、HTML アプリケーション、コントロールパネルのアプレット、ドライバー		デーモン (fapolicyd) (GitHub)、OpenSCAP などのアプリケーション制御用のセキュリティツール Amazon CloudWatch エージェント 組織全体と AMI を共有する アプリケーションチームで、最新の AMI が参照されていることを確認する パッチ管理に AMI パイプラインを使用する	
Microsoft の「推奨ブロックルール」が実装済みです。 Microsoft の「推奨ドライバーブロックルール」が実装済みです。	「Implementing Application Control」(ACSC ウェブサイト) を参照	該当しない	該当しない

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
アプリケーション制御ルールセットを、年 1 回以上の頻度で検証しています。	テーマ 8: 手動プロセスの仕組みの実装: セキュリティポリシー更新の仕組みを実装する	利用不可	SEC01-BP08 新しいセキュリティサービスと機能を定期的に評価して実装する
ワークステーションとサーバーで許可されている実行とブロックしている実行を一元的にログに記録し、それらのログを不正な変更や削除から保護しています。また、侵害の兆候がないかをモニタリングし、サイバーセキュリティイベントが検出されたらそれらに対処します。	テーマ 7: ログ記録およびモニタリングの一元化: ログ記録の有効化	CloudWatch エージェントを使用して、システムレベルのログを CloudWatch Logs に発行する GuardDuty による検出結果にアラートを設定する CloudTrail で組織の証跡を作成する バージョニングと S3 Object Lock を使用して、Amazon S3 に保存されているデータを保護する	SEC04-BP01 サービスとアプリケーションのログ記録を設定する SEC04-BP02 標準化された場所でログ、検出結果、メトリクスをキャプチャする

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
	テーマ 7: ログ記録およびモニタリングの一元化: ログ記録のベストプラクティスを実装する	CloudTrail セキュリティのベストプラクティスを実装する SCPs を使用して、ユーザーがセキュリティサービスを無効にできないようにする (AWS ブログ記事) を使用して CloudWatch Logs のログデータを暗号化する AWS Key Management Service	SEC04-BP01 サービスとアプリケーションのログ記録を設定する SEC04-BP02 標準化された場所でログ、検出結果、メトリクスをキャプチャする

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
	テーマ 7: ログ記録およびモニタリングの一元化: ログを一元管理する	複数のアカウントから CloudTrail ログを受信する ログアーカイブアカウントにログを送信する 監査と分析のために アカウントに CloudWatch Logs を一元化する (AWS ブログ記事) Amazon Inspector の管理を一元化する で組織全体のアグリゲータを作成する AWS Config (AWS ブログ記事) Security Hub CSPM の管理を一元化する GuardDuty の管理を一元化する Amazon Security Lake の使用を検討する	SEC04-BP02 標準化された場所でログ、検出結果、メトリクスをキャプチャする

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
	テーマ 8: 手動プロセスの仕組みの実装: コンプライアンス上の不備を確認し対処する仕組みを実装する	手動プロセスの負担を軽減するために、AWS Config ルールといった自動化の実装を検討する	OPS02-BP02 プロセスと手順には特定の所有者が存在する OPS02-BP03 パフォーマンスに責任を持つ所有者が運用アクティビティに存在する OPS02-BP04 責任と所有権を管理するためのメカニズムが存在する

アプリケーションへのパッチ適用

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
アセットを自動検出する方法を少なくとも 2 週間ごとに使用して、後続の脆弱性スキャンアクティビティでアセットを検出できるようにしています。	テーマ 1: マネージドサービスの使用: 脆弱性のスキャン テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理: 脆弱性スキャンの実装 テーマ 3: 自動化によるミュータブルなインフラストラクチャ	組織内のすべてのアカウントで Amazon Inspector を有効にする Amazon Inspector を使用して、Amazon ECR リポジトリの拡張スキャンを設定する セキュリティ関連の検出結果をトリアージして修正する脆弱	SEC06-BP01 脆弱性管理を実行する SEC06-BP05 コンピューティング保護を自動化する

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
	の管理: 脆弱性スキャンの実装	性管理プログラムを構築する	
	テーマ 7: ログ記録およびモニタリングの一元化: ログを一元管理する	複数のアカウントから CloudTrail ログを受信する ログアーカイブアカウントにログを送信する 監査と分析のために CloudWatch Logs をアカウントに一元化する (AWS ブログ記事) Amazon Inspector の管理を一元化する AWS Configで組織全体のアグリゲータを作成する (AWS ブログ記事) Security Hub CSPM の管理を一元化する GuardDuty の管理を一元化する Security Lake の使用を検討する	SEC04-BP02 標準化された場所でログ、検出結果、メトリクスをキャプチャする

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
脆弱性スキャン活動には、最新の脆弱性データベースを備えた脆弱性スキャナーを使用しています。	テーマ 1: マネージドサービスの使用: 脆弱性のスキャン テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理: 脆弱性スキャンの実装 テーマ 3: 自動化によるミュータブルなインフラストラクチャの管理: 脆弱性スキャンの実装	組織内のすべてのアカウントで Amazon Inspector を有効にする Amazon Inspector を使用して、Amazon ECR リポジトリの拡張スキャンを設定する セキュリティ関連の検出結果をトリアージして修正する脆弱性管理プログラムを構築する	SEC06-BP01 脆弱性管理を実行する SEC06-BP05 コンピューティング保護を自動化する
脆弱性スキャナーを、少なくとも毎日使用して、インターネット向けサービスのセキュリティ脆弱性対処に不足しているパッチおよび更新プログラムを特定しています。			
脆弱性スキャナーを、少なくとも毎週使用して、オフィスの生産性向上スイート、ウェブブラウザとその拡張機能、E メールクライアント、PDF ソフトウェア、セキュリティ製品などのセキュリティ脆弱性対処に不足しているパッチおよび更新プログラムを特定しています。	「Technical example: Patch applications」(ACSC ウェブサイト)」を参照してください。	該当しない	該当しない

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
脆弱性スキャナーを少なくとも 2 週間ごとに使用して、他のアプリケーションのセキュリティ脆弱性対処に不足しているパッチや更新プログラムを特定しています。	テーマ 1: マネージドサービスの使用: 脆弱性のスキャン テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理: 脆弱性スキャンの実装 テーマ 3: 自動化によるミュータブルなインフラストラクチャの管理: 脆弱性スキャンの実装	組織内のすべてのアカウントで Amazon Inspector を有効にする Amazon Inspector を使用して、Amazon ECR リポジトリの拡張スキャンを設定する セキュリティ関連の検出結果をトリアージして修正する脆弱性管理プログラムを構築する	SEC06-BP01 脆弱性管理を実行する SEC06-BP05 コンピューティング保護を自動化する
インターネット向けサービスのセキュリティ脆弱性に対処するパッチ、更新プログラム、ベンダーによる緩和を、リリースから 2 週間以内に、エクスプロイトが存在する場合は 48 時間以内に適用しています。	テーマ 1: マネージドサービスの使用: 脆弱性のスキャン テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理: 脆弱性スキャンの実装 テーマ 3: 自動化によるミュータブルなインフラストラクチャの管理: 脆弱性スキャンの実装	組織内のすべてのアカウントで Amazon Inspector を有効にする Amazon Inspector を使用して、Amazon ECR リポジトリの拡張スキャンを設定する セキュリティ関連の検出結果をトリアージして修正する脆弱性管理プログラムを構築する	SEC06-BP01 脆弱性管理を実行する

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
	テーマ 3: 自動化によるミュータブルなインフラストラクチャの管理: パッチ適用を自動化する	AWS 組織内のすべてのアカウントで Patch Manager を有効にする	SEC06-BP01 脆弱性管理を実行する SEC06-BP05 コンピューティング保護を自動化する
オフィスの生産性向上スイート、ウェブブラウザとその拡張機能、E メールクライアント、PDF ソフトウェア、セキュリティ製品のセキュリティ脆弱性に対処するパッチ、更新プログラム、ベンダーによる緩和を、リリースから 2 週間以内に、エクスプロイトが存在する場合は 48 時間以内に適用しています。	「Technical example: Patch applications」(ACSC ウェブサイト)」を参照してください。	該当しない	該当しない

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
他のアプリケーションのセキュリティ脆弱性に対処するパッチ、更新プログラム、ベンダーによる緩和を、リリースから 1 か月以内に適用しています。	テーマ 1: マネージドサービスの使用: 脆弱性のスキャン テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理: 脆弱性スキャンの実装 テーマ 3: 自動化によるミュータブルなインフラストラクチャの管理: 脆弱性スキャンの実装	組織内のすべてのアカウントで Amazon Inspector を有効にする Amazon Inspector を使用して、Amazon ECR リポジトリの拡張スキャンを設定する セキュリティ関連の検出結果をトリアージして修正する脆弱性管理プログラムを構築する	SEC06-BP01 脆弱性管理を実行する
	テーマ 3: 自動化によるミュータブルなインフラストラクチャの管理: パッチ適用を自動化する	AWS 組織内のすべてのアカウントで Patch Manager を有効にする	SEC06-BP01 脆弱性管理を実行する SEC06-BP05 コンピューティング保護を自動化する
ベンダーがサポートしなくなったアプリケーションは削除済みです。	テーマ 8: 手動プロセスの仕組みの実装: コンプライアンス上の不備を確認し対処する仕組みを実装する	ソフトウェアポリシーに必要なソフトウェアが稼働しているインスタンスを可視化するために、AWS Systems Manager Inventory の使用を検討する	SEC06-BP02 強化イメージからコンピューティングをプロビジョニングする

Microsoft Office マクロ設定の構成

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
ビジネス要件を実証していないユーザーに対しては、Microsoft Office マクロを無効にしています。	「Technical example: Configure macro settings」(ACSC ウェブサイト) を参照してください。	該当しない	該当しない
サンドボックス環境や信頼できる場所から実行される、または信頼できる発行元のデジタル署名がある Microsoft Office マクロのみ、実行を許可しています。			
Microsoft Office マクロに悪意のあるコードがないことを検証する責任を持つ特権ユーザーにのみ、信頼できる場所内のコンテンツへの書き込みまたは変更を許可しています。			
信頼できないパブリッシャーがデジタル署名した Microsoft Office マクロは、メッセージバーまたはバックステージビューを介して有効			

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
化できないようにしています。			
Microsoft Office の信頼できるパブリッシャーのリストは、年 1 回以上の頻度で検証しています。			
インターネットを送信元とするファイルの Microsoft Office マクロは、ブロックされるようにしています。			
Microsoft Office マクロへのウイルス対策スキャンを有効にしています。			
Microsoft Office マクロによる Win32 API コールの実行は、ブロックされるようにしています。			
Microsoft Office マクロのセキュリティ設定は、ユーザーが変更できないようにしています。			

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
許可またはブロックされている Microsoft Office マクロの実行を一元的にログに記録し、それらのログを不正な変更や削除から保護しています。また、侵害の兆候がないかをモニタリングし、サイバーセキュリティイベントが検出されたらそれらに対処します。			

ユーザーアプリケーションの強化

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
インターネットから取得された Java をウェブブラウザで処理できないようにしています。	「Technical example: User application hardening」(ACSC ウェブサイト) を参照してください。	該当しない	該当しない
インターネットから取得されたウェブ広告をウェブブラウザで処理できないようにしています。			

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
Internet Explorer 11 は、無効になっているか削除済みです。			
Microsoft Office では、子プロセスの作成がブロックされるようにしています。			
Microsoft Office では、実行可能コンテンツの作成がブロックされるようにしています。			
Microsoft Office では、他のプロセスへのコード挿入がブロックされるようにしています。			
Microsoft Office は、OLE パッケージをアクティブ化できないように設定済みです。			
PDF ソフトウェアでは、子プロセスの作成がブロックされるようにしています。			

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
ウェブブラウザ、Microsoft Office、PDF ソフトウェアの ACSC またはベンダー強化ガイダンスを実装済みです。			
ウェブブラウザ、Microsoft Office、PDF ソフトウェアのセキュリティ設定をユーザーが変更できないようにしています。			
.NET Framework 3.5 (2.0 .NET と 3.0 を含む) は、無効になっているか削除済みです。			
Windows PowerShell 2.0 は、無効になっているか削除済みです。			
PowerShell は、制約言語モードを使用するように設定済みです。			

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
ブロックされている PowerShell スクリプトの実行を一元的にログに記録し、それらのログを不正な変更や削除から保護しています。また、侵害の兆候がないかをモニタリングし、サイバーセキュリティイベントが検出されたらそれらに対処します。			

管理者権限の制限

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
システムおよびアプリケーションに対する特権アクセスのリクエストは、最初のリクエスト時に検証しています。	テーマ 4: ID の管理: ID フェデレーションの実装	一時的な認証情報を使用して AWS にアクセスする人間のユーザーには、ID プロバイダーとのフェデレーションを求めている	SEC02-BP04 一元化された ID プロバイダーを利用する SEC03-BP01 アクセス要件を定義する
システムおよびアプリケーションへの特権アクセスは、それらが再検証されない限り、12 か月後に自	テーマ 4: ID の管理: ID フェデレーションの実装	一時的な認証情報を使用して AWS にアクセスする人間のユーザーには、ID プロバイダーとのフェデ	SEC02-BP04 一元化された ID プロバイダーを利用する

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
動的に無効化しています。		レーションを求めている	
	テーマ 4: ID の管理: 認証情報のローテーション	ワークロードが IAM ロールを使用して にアクセスするよう要求する AWS 使用されていない IAM ロールの削除を自動化する 長期認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションする AWS Summit ANZ 2023: クラウドでの一時的な認証情報へのジャーニー (YouTube 動画)	SEC02-BP05 定期的に認証情報を監査およびローテーションする

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
システムおよびアプリケーションへの特権アクセスが 45 日間アクティブでない場合は、それらを自動的に無効化しています。	テーマ 4: ID の管理: ID フェデレーションの実装 テーマ 4: ID の管理: 認証情報のローテーション	人間のユーザーに、一時的な認証情報 AWS を使用して にアクセスすることを ID プロバイダーと連携させる ワークロードが IAM ロールを使用して にアクセスするよう要求する AWS 使用されていない IAM ロールの削除を自動化する 長期認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションする AWS Summit ANZ 2023: クラウドでの一時的な認証情報へのジャーニー (YouTube 動画)	SEC02-BP04 一元化された ID プロバイダーを利用する SEC02-BP05 定期的に認証情報を監査およびローテーションする

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
システムおよびアプリケーションへの特権アクセスは、業務上ユーザーおよびサービスに必要なものにのみ制限しています。	テーマ 4: ID の管理: 最小特権アクセス許可の適用	ルートユーザーの認証情報を保護し、日常的なタスクには使用しない IAM Access Analyzer を使用して、アクセスアクティビティに基づいて最小特権ポリシーを生成する IAM Access Analyzer を使用してリソースへのパブリックアクセスとクロスアカウントアクセスを検証する IAM Access Analyzer を使用して IAM ポリシーを検証し、安全で機能的なアクセス許可を行う 複数のアカウントでアクセス許可ガードレールを確立する アクセス許可の境界を使用して、ID ベースのポリシーで付与可能な権限の上限を設定する	SEC01-BP02 安全なアカウントのルートユーザーとプロパティ SEC03-BP02 最小特権のアクセスを付与する

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
		IAM ポリシーの条件を使用してアクセスをさらに制限する 未使用のユーザー、ロール、アクセス許可、ポリシー、認証情報を定期的に確認して削除する AWS 管理ポリシーを開始し、最小特権のアクセス許可に移行する IAM Identity Center のアクセス許可セット機能を使用する	
特権アカウントでは、インターネット、E メール、ウェブサービスにアクセスできないようにしています。	「Technical example: Restrict administrative privileges」(ACSC ウェブサイト) を参照してください。	インターネットへのアクセス機能を持たない VPC がその状態を維持するための SCP の実装を検討する	該当しない

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
特権ユーザーが特権運用環境と非特権運用環境を分けて使用するようにしています。	テーマ 5: データ境界を確立する	データ境界を確立する。OFFICIAL: SENSITIVE または PROTECTED といった異なるデータ分類の環境間、あるいは、開発、テスト、本番といった異なるリスクレベルの環境間にデータ境界を実装することを検討してください。	SEC06-BP03 手動管理とインタラクティブアクセスを削減する
特権運用環境を非特権運用環境内で仮想化しないようにしています。			
特権のないアカウントでは、特権アカウントの運用環境にログオンできないようにしています。			
特権アカウント (ローカル管理者アカウントを除く) では、権限のない運用環境にログオンできないようにしています。			
システムとアプリケーションの管理に、ジャストインタイム管理を使用しています。	テーマ 4: ID の管理: ID フェデレーションの実装	人間のユーザーに、一時的な認証情報 AWS を使用して にアクセスすることを ID プロバイダーと連携させる 環境への一時的な昇格アクセスを実装する AWS (AWS ブログ記事)	SEC02-BP04 一元化された ID プロバイダーを利用する

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
管理アクティビティは、ジャンプサーバーを介して行っています。	テーマ 1: マネージドサービスの使用 テーマ 3: 自動化によるミュータブルなインフラストラクチャの管理: 手動ではなく、自動プロセスを使用する	SSH または RDP による直接アクセスではなく、Session Manager または Run Command を使用する	SEC01-BP05 セキュリティ管理の範囲を縮小する SEC06-BP03 手動管理とインタラクティブアクセスを削減する
ローカル管理者アカウントとサービスアカウントの認証情報は、一意で、予測不可能であり、管理対象となっています。	「Technical example: Restrict administrative privileges」(ACSC ウェブサイト) を参照してください。	該当しない	該当しない
Windows Defender Credential Guard と Windows Defender Remote Credential Guard を有効にしています。			

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
特権アクセスの使用を一元的にログに記録し、それらのログを不正な変更や削除から保護しています。また、侵害の兆候がないかをモニタリングし、サイバーセキュリティイベントが検出されたらそれらに対処します。	テーマ 7: ログ記録およびモニタリングの一元化: ログ記録の有効化 テーマ 7: ログ記録およびモニタリングの一元化: ログを一元管理する	CloudWatch エージェントを使用して、OS レベルのログを CloudWatch Logs に発行する 組織の CloudTrail を有効にする 監査と分析のために アカウントに CloudWatch Logs を一元化する (AWS ブログ記事) Amazon Inspector の管理を一元化する Security Hub CSPM の管理を一元化する AWS Configで組織全体のアグリゲータを作成する (AWS ブログ記事) GuardDuty の管理を一元化する Amazon Security Lake の使用を検討する 複数のアカウントから CloudTrail ログを受信する	SEC04-BP01 サービスとアプリケーションのログ記録を設定する SEC04-BP02 標準化された場所でログ、検出結果、メトリクスをキャプチャする
特権アカウントおよびグループへの変更を一元的にログに記録し、それらのログを不正な変更や削除から保護しています。また、侵害の兆候がないかをモニタリングし、サイバーセキュリティイベントが検出されたらそれらに対処します。			

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
		ログアーカイブアカウントにログを送信する	

オペレーティングシステムへのパッチ適用

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
インターネット向けサービスが稼働するオペレーションシステムのセキュリティ脆弱性に対処するパッチ、更新プログラム、ベンダーによる緩和を、リリースから 2 週間以内に、エクスプロイトが存在する場合は 48 時間以内に適用しています。	テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理: AMI およびコンテナビルドパイプラインの実装	EC2 Image Builder を使用して、以下を構築する • AWS Systems Manager エージェント (SSM エージェント) • Security Enhanced Linux (SELinux) (GitHub)、ファイルアクセスポリシーデーモン (fapolicyd) (GitHub)、OpenSCAP などのアプリケーション制御用のセキュリティツール • Amazon CloudWatch エージェント 組織全体と AMI を共有する	SEC01-BP05 セキュリティ管理の範囲を縮小する SEC06-BP01 脆弱性管理を実行する SEC06-BP03 手動管理とインタラクティブアクセスを削減する

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
		アプリケーションチームで、最新の AMI が参照されていることを確認する パッチ管理に AMI パイプラインを使用する	
	テーマ 1: マネージドサービスの使用: パッチ適用の有効化 テーマ 3: 自動化によるミュータブルなインフラストラクチャの管理: パッチ適用を自動化する	AWS 組織内のすべてのアカウントで Patch Manager を有効にする	SEC06-BP01 脆弱性管理を実行する SEC06-BP05 コンピューティング保護を自動化する

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
ワークステーション、サーバー、ネットワークデバイスが稼働するオペレーティングシステムのセキュリティ脆弱性に対処するパッチ、更新プログラム、ベンダーによる緩和を、リリースから 2 週間以内に、エクスプロイトが存在する場合は 48 時間以内に適用しています。	テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理: AMI およびコンテナビルドパイプラインの実装	EC2 Image Builder を使用して、以下を構築する • AWS Systems Manager エージェント (SSM エージェント) • Security Enhanced Linux (SELinux) (GitHub)、ファイルアクセスポリシーデーモン (fapolicyd) (GitHub)、OpenSCAP などのアプリケーション制御用のセキュリティツール • Amazon CloudWatch エージェント 組織全体と AMI を共有する アプリケーションチームで、最新の AMI が参照されていることを確認する パッチ管理に AMI パイプラインを使用する	SEC01-BP05 セキュリティ管理の範囲を縮小する SEC06-BP01 脆弱性管理を実行する SEC06-BP02 強化イメージからコンピューティングをプロビジョニングする

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
	テーマ 1: マネージドサービスの使用: パッチ適用の有効化 テーマ 3: 自動化によるミュータブルなインフラストラクチャの管理: パッチ適用を自動化する	AWS 組織内のすべてのアカウントで Patch Manager を有効にする	SEC06-BP01 脆弱性管理を実行する SEC06-BP05 コンピューティング保護を自動化する
脆弱性スキャナーを少なくとも毎日使用して、インターネット向けサービスが稼働するオペレーションシステムのセキュリティ脆弱性対処に不足しているパッチまたは更新プログラムを特定しています。	テーマ 1: マネージドサービスの使用: 脆弱性のスキャン テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理: 脆弱性スキャンの実装 テーマ 3: 自動化によるミュータブルなインフラストラクチャの管理: 脆弱性スキャンの実装	組織内のすべてのアカウントで Amazon Inspector を有効にする Amazon Inspector を使用して、Amazon ECR リポジトリの拡張スキャンを設定する セキュリティ関連の検出結果をトリアージして修正する脆弱性管理プログラムを構築する	SEC01-BP05 セキュリティ管理の範囲を縮小する SEC06-BP01 脆弱性管理を実行する SEC06-BP02 強化イメージからコンピューティングをプロビジョニングする
脆弱性スキャナーを少なくとも毎週使用して、ワークステーション、サーバー、ネットワークデバイスが稼働するオペレーティングシステムのセキュリティ脆弱性対処に不足しているパッチまたは更新プログラムを特定しています。			

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
ワークステーション、サーバー、ネットワークデバイスのオペレーティングシステムには、最新または以前のリリースを使用しています。	テーマ 2: 安全なパイプラインによる、イミュータブルなインフラストラクチャの管理: 脆弱性スキャンの実装	EC2 Image Builder を使用して、以下を構築する • AWS Systems Manager エージェント (SSM エージェント) • Security Enhanced Linux (SELinux) (GitHub)、ファイルアクセスポリシーデーモン (fapolicyd) (GitHub)、OpenSCAP などのアプリケーション制御用のセキュリティツール • Amazon CloudWatch エージェント 組織全体と AMI を共有する アプリケーションチームで、最新の AMI が参照されていることを確認する パッチ管理に AMI パイプラインを使用する	SEC01-BP05 セキュリティ管理の範囲を縮小する SEC06-BP01 脆弱性管理を実行する SEC06-BP02 強化イメージからコンピューティングをプロビジョニングする
ベンダーがサポートしなくなったオペレーティングシステムは、置き換え済みです。			

多要素認証

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
組織がインターネットで公開しているサービスで組織ユーザーを認証する場合は、多要素認証を適用しています。	テーマ 4: ID の管理: ID フェデレーションの実装	人間のユーザーに、一時的な認証情報 AWS を使用して にアクセスすることを ID プロバイダーと連携させる AWS 環境への一時的な昇格アクセスを実装する	SEC02-BP04 一元化された ID プロバイダーを利用する
	テーマ 4: ID の管理: MFA の強制	ルートユーザーに MFA を求める を通じて MFA を要求する AWS IAM アイデンティティセンター サービス固有の API アクションに MFA を要求することを検討する	SEC02-BP01 強力なサインインメカニズムを使用する
サードパーティーがインターネットで公開しており、組織の機密データの処理、保存、通信を行うサービスで組織ユーザーを認証する場合は、多要素認証を適用しています。	「Implementing Multi-Factor Authentication」(ACSC ウェブサイト) を参照してください。	該当しない	該当しない

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
サードパーティーがインターネットで公開しており、組織の非機密データの処理、保存、通信を行うサービスで組織ユーザーを認証する場合は、多要素認証を適用しています (利用可能な場合)。			
組織がインターネットで公開しているサービスで組織以外のユーザーを認証する場合は、多要素認証をデフォルトで有効にしています (ただし、ユーザーはオプトアウトを選択可能)。			
システムの特権ユーザーを認証する場合は、多要素認証を適用しています。	テーマ 4: ID の管理: ID フェデレーションの実装	人間のユーザーに、一時的な認証情報 AWS を使用して にアクセスすることを ID プロバイダーと連携させる AWS 環境への一時的な昇格アクセスを実装する	SEC02-BP04 一元化された ID プロバイダーを利用する

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
	テーマ 4: ID の管理: MFA の強制	ルートユーザーに MFA を求める IAM Identity Center を通じて MFA を要求する サービス固有の API アクションに MFA を要求することを検討する	SEC02-BP01 強力なサインインメカニズムを使用する
重要なデータリポジトリにアクセスするユーザーを認証する場合は、多要素認証を適用しています。	テーマ 4: ID の管理: MFA の強制	サービス固有の API アクションに MFA を要求することを検討する	SEC02-BP01 強力なサインインメカニズムを使用する
認証側としてのなりすましを防ぐ仕組みとして多要素認証を取り入れ、次のいずれかを使用しています: ユーザーが所有している要素および知っている要素、もしくは、ユーザーがロックを解除できる要素 (既知の情報または自分自身の特徴)	「Implementing Multi-Factor Authentication」(ACSC ウェブサイト) を参照してください。	該当しない	該当しない

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
成功した多要素認証と失敗した多要素認証を一元的にログに記録し、それらのログを不正な変更や削除から保護しています。また、侵害の兆候がないかをモニタリングし、サイバーセキュリティイベントが検出されたらそれらに対処します。	テーマ 7: ログ記録およびモニタリングの一元化: ログ記録の有効化 テーマ 7: ログ記録およびモニタリングの一元化: ログを一元管理する	監査と分析のために アカウントに CloudWatch Logs を一元化する (AWS ブログ記事) Amazon Inspector の管理を一元化する Security Hub CSPM の管理を一元化する AWS Configで組織全体のアグリゲータを作成する (AWS ブログ記事) GuardDuty の管理を一元化する Security Lake の使用を検討する 複数のアカウントから CloudTrail ログを受信する ログアーカイブアカウントにログを送信する	SEC04-BP01 サービスとアプリケーションのログ記録を設定する SEC04-BP02 標準化された場所でログ、検出結果、メトリクスをキャプチャする

定期バックアップ

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
重要なデータ、ソフトウェア、構成設定のバックアップは、ビジネス継続性の要件に従って、組織的な調整とレジリエンス確保を行い実行および保持しています。	テーマ 6: バックアップの自動化: データのバックアップおよび復旧を自動化する	にデータバックアップを実装する AWS 大規模なデータバックアップの自動化 (AWS ブログ記事)	REL09-BP01 バックアップが必要なすべてのデータを特定し、バックアップする、またはソースからデータを再現する REL09-BP02 バックアップを保護し、暗号化する REL09-BP03 データバックアップを自動的に実行する
ディザスタリカバリ演習の一環として、組織的な調整を行った上で、バックアップからのシステム、ソフトウェア、重要なデータの復元をテストしています。	テーマ 6: バックアップの自動化: データのバックアップおよび復旧を自動化する テーマ 6: バックアップの自動化: AWS Backup の成果物全体にガバナンスを実装する	Automate data recovery validation with AWS Backup (AWS ブログ記事) AWS Backup Audit Manager を使用してポリシーのコンプライアンスを監査する AWS Backup	REL09-BP04 データの定期的な復旧を行ってバックアップの完全性とプロセスを確認する
特権のないアカウントと特権アカウント (バックアップ管理者を除く) では、バックアップにアクセスで	テーマ 6: バックアップの自動化: AWS Backup 結果全体にガバナンスを実装する	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 ブログ記事)	SEC08-BP04 アクセスコントロールを適用する

Essential Eight コントロール	実装のガイダンス	AWS リソース	AWS Well-Architected ガイダンス
きないようにしています。		AWS Backup ボールトロックを使用してバックアップボールトのセキュリティを向上させる AWS Backup Audit Manager を使用してポリシーのコンプライアンスを監査する AWS Backup	
特権のないアカウントと特権アカウント (バックアップブレークグラスアカウントを除く) では、バックアップの変更または削除を行えないようにしています。			

注意

お客様は、本書に記載されている情報を独自に評価する責任を負うものとします。本書は、(a) 情報提供のみを目的とし、(b) AWS の現行製品と慣行について説明しており、これらは予告なしに変更されることがあり、(c) AWS およびその関連会社、サプライヤー、またはライセンサーからの契約上の義務や保証をもたらすものではありません。AWS の製品やサービスは、明示または黙示を問わず、一切の保証、表明、条件なしに「現状のまま」提供されます。お客様に対する AWS の責任は AWS 契約によって規定されます。本書は、AWS とお客様との間で締結されるいかなる契約の一部でもなく、その内容を修正するものでもありません。

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
ベストプラクティスの更新	AWS Well-Architected フレームワークのセキュリティの柱における最新のベストプラクティスを反映するために、このガイドを更新しました。	2024 年 11 月 6 日
初版発行	—	2023 年 10 月 20 日

AWS 規範ガイダンスの用語集

以下は、 AWS 規範ガイダンス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、およびそれらを移行す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

A2A (Agent-to-Agent)

タスクの委任と状態転送をサポートするagent-to-agentコラボレーション用のステートフルプロトコル。

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。

[エージェント]

目標を達成するためのツールを使用して、自律的に推論、計画、アクションを実行できる AI システム。

エージェントオペレーション

AI エージェントを本番環境で大規模に構築、テスト、デプロイ、実行するための運用プラクティス。

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。

アベイラビリティーゾーン (AZ)

他のアベイラビリティーゾーンの障害から 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ー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 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の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 1 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。

ブレークグラスアクセス

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 ガイダンスの「ブレークグラス手順の実装」インジケータを参照してください。

ブラウンフィールド戦略

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ン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。

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。

シチズンデベロッパー

専門的な技術スキルを持たないノーコード/ローコードプラットフォームを使用して AI アプリケーションを作成するビジネスユーザー。

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が、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。信頼できる ID のみが、期待されるネットワークから信頼できる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。この戦略を採用するときは AWS、 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て、リソースの安全性を確保します。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で使用できるサービス」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ュファクチャリング・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。

ディザスタ

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 (ボストン: 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

「開発バリューストリームマッピング」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使用できる、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、アベイラビリティーゾーン、コントロールプレーン AWS リージョン、データプレーンなどの境界で、障害の影響を制限し、ワークロードの耐障害性を向上させるのに役立ちます。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。

FM ゲートウェイ

基盤モデルへのアクセスを制御および正規化する一元化された仲介者。LLM ゲートウェイとも呼ばれます。

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、、Amazon GuardDuty AWS Security Hub CSPM、、 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。

ガードレール (AI)

エージェントの入力と出力をフィルタリング、検証、制約する安全メカニズムは、責任ある安全な AI の動作を確保するのに役立ちます。

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。

ヒューman-in-the-loop (HitL)

エージェントの実行が重要な決定時点で人間によるレビューと承認のために一時停止するワークフローパターン。

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

IaC

「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 が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、プラットフォームを AWS 運用し、ユーザーが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。

MCP

「モデルコンテキストプロトコル」を参照してください。

モデルコンテキストプロトコル (MCP)

エージェントとツール間の通信のためのステートレスプロトコル。

MCP サーバー

Model Context Protocol を通じて 1 つ以上のツールを公開するサービス。

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/サブスクライブ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 のすべてのイベント AWS CloudTrail をログに記録する 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 のエンティティ。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

読み取り専用に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)

サービスの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 とは独立しています。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。

レスポンシブコントロール

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「レスポンシブコントロール」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、フェデレーティ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、 にログイン AWS マネジメントコンソール 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、レスポンシブ、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

のエントリポイントの 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、お客様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。

シャドウ AI

組織内の管理対象チャネルの外部で構築または使用される認可されていない AI アプリケーション。

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ー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の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。

tool

エージェントが外部システムでオペレーションを実行するために呼び出すことができる関数または API。

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な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「 を他の AWS のサービス AWS Organizations で使用する 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

ゼロデイエクスプロイト

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。