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)

オーストラリア信号局 (ASD) は、組織がサイバーセキュリティの脅威のリスクを軽減するのに役立つ戦略を作成し、以前に策定しました。これらの戦略のうち 8 つが Essential Eight フレームワークを形成するために選択されました。オーストラリアの多くの公共部門および民間部門組織は、Essential Eight フレームワークの下で成熟する必要があります。

オーストラリアサイバーセキュリティセンター (ACSC) は、 Microsoftベースのインターネット接続ネットワークを保護するために Essential Eight フレームワークを作成しました。ただし、多くの組織は、オンプレミスとクラウドの両方のすべての環境で Essential Eight 成熟度に到達する必要があります。

Essential Eight フレームワークには、組織がプログレッシブイテレーションを通じてフレームワークを実装するのに役立つように設計された成熟度モデルも含まれています。このモデルは、成熟度レベル 0～3 の概要を示しています。成熟度レベル 3 は、高度なサイバーセキュリティ戦術や高度にターゲットを絞った攻撃に対する回復力を表します。このガイドでは、Essential Eight 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とコンプライアンス

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、機密性の高いトランザクションの処理、重要なサービスの構築を行います。

このガイドでは、Essential Eight フレームワークをクラウドに適応させる方法について説明しますが、 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役立つ以下の認定とモデル AWS も提供しています。

- 情報セキュリティ登録評価プログラム
- 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

情報セキュリティ登録評価プログラム

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、PROTECTED レベルで評価されています。独立系オーストラリア信号局 (ASD) 認定 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 AWS のサービス の の完全なリストについては、「対象AWS のサービス 範囲内の : 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

オーストラリアホスティング認定フレームワークは、政府のシステムとデータの安全な管理をサポート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、認定戦略的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ながら、急速なペースで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

AWS 責任共有モデルでは、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。

この共有モデルは、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、サービスが運用されている施設の物理的なセキュリティまで、多くのコンポーネントを AWS 運用、管理、制御するため、コンプライアンスと運用上の負担を軽減するのに役立ちます。お客様は、ゲストオペレーティングシステム (更新プログラムやセキュリティパッチを含む) およびその他の関連するアプリケーションソフトウェアを管理する責任があります。また、 が提供する AWS セキュリティグループファイアウォールを設定する責任も負います。

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、使用されるサービス、お客様の IT 環境へのそれらのサービスの統合、適用可能な法律および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向けに、安全で高性能、回復力、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。このフレームワークは、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、持続可能性の 6 つの柱を中心に構築されています。

AWS は、ワークロードを確認するための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。ワークロードの信頼性、安全性、効率性、コスト効率を向上させるための推奨事項を提供します。

クラウドの Essential Eight 戦略を再解釈する

以下は、 Microsoftベースのインターネット接続ネットワーク用に設計された元の Essential Eight 緩和戦略です。

- アプリケーションコントロール
- パッチアプリケーション
- Microsoft Office マクロ設定を構成する
- ユーザーアプリケーションの強化
- 管理者権限を制限する
- パッチオペレーティングシステム
- 多要素認証
- 定期的なバックアップ

Essential Eight フレームワークはクラウド環境向けに設計されていないことを繰り返し説明することが重要です。ただし、基本原則は適用可能で、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なさまざまなアプローチにより、セキュリティが向上し、コンプライアンスの負担が大幅に軽減されます。オンプレミス環境では、セキュリティのあらゆる側面を担当し、継承されたコントロールはありません。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。また、オートメーションとマネージドサービスを使用することで、コンプライアンスの負担を軽減することもできます。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。Amazon Simple Storage Service (Amazon S3) と Amazon DynamoDB は、 マネージドサービスの例です。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を使用する「」セクションを参照してください。

したがって、Essential Eight 戦略をワークロードに適したものにするには、いくつかの再解釈が必要です AWS。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を使用する
- テーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する
- テーマ 3: 自動化による変更可能なインフラストラクチャの管理
- テーマ 4: ID を管理する
- テーマ 5: データ境界を確立する
- テーマ 6: バックアップを自動化する
- テーマ 7: ログ記録とモニタリングを一元化する
- テーマ 8: 手動プロセスのメカニズムを実装する

各テーマには、トピックの概要、関連する AWS Well-Architected フレームワークのベストプラクティス、および 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。この手順では、手動の手順や、AWS Config ルールを使用したオートメーションの設定について説明します。手動ステップでは、検出結果に対処するためのメカニズムが必要です。詳細については、テーマ 8: 手動プロセスのメカニズムを実装する「. AWS Config rules requires similar oversight or automation in order to correctte noncompliant resources」を参照してください。これらのテーマに沿ったガイダンスに従うことで、クラウドのメリットを最大化するアプローチで Essential Eight の成熟度に到達できます。

クラウドの Essential Eight 戦略を再解釈する

Essential Eight フレームワークはクラウド環境向けに設計されていないため、各 Essential Eight 戦略の基本原則に対処する際には、クラウドネイティブなアプローチを取ることが不可欠です。このアプローチは、2 つの主要な質問によって異なります。

どのサービスを使用していますか？

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役立ちます。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、セキュリティの最適化 AWS を維持する責任を にシフトします。また、マネージドサービスでは、サービスを維持する運用上および管理上の負担がなくなり、イノベーションに集中する時間が増えます。

マネージドサービスには、Amazon API Gateway、、AWS LambdaDynamoDB などのサーバーレスサービスが含まれます。Amazon Relational Database Service (Amazon RDS) 上のデータベースは、Amazon Elastic Compute Cloud (Amazon EC2) 上のデータベースよりも運用上の責任が少なくて済みます。

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にパッチを適用する責任があります。Amazon RDS や Amazon Redshift などの他のサービスでは、メンテナンスウィンドウ中にパッチを管理する必要がある場合があります。

どのデプロイモデルを使用していますか？

組織は、変更可能またはイミュータブルなインフラストラクチャアプローチを使用していますか？

変更可能なインフラストラクチャモデルは、本番ワークロードの既存のインフラストラクチャを更新および変更します。 これは、サーバーインフラストラクチャの置き換えに非常にコストがかかり、時間がかかりすぎたため、本番環境にあるサーバーに変更を適用するという最も現実的なアプローチだったクラウド前の標準的なデプロイ方法です。クラウドにおける変更可能なアプローチの例としては、手動または AWS Systems Manager Run Command や などのソフトウェアデプロイサービスを使用して、実行中の EC2 インスタンスにアプリケーションの変更を直接デプロイすることが挙げられますAWS CodeDeploy。

イミュータブルなインフラストラクチャモデルは、既存のインフラストラクチャを更新、パッチ適用、または変更するのではなく、本番ワークロード用の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は、 AWS CloudFormationまたは でアプリケーションスタックを定義することですAWS Cloud Development Kit (AWS CDK)。これらのサービスを使用して、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、ローリングやブルー/グリーンなどのデプロイ方法を使用します。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」の「イミュータブルインフラストラクチャを使用したデプロイ」のベストプラクティスを参照してください。

例えば、パッチ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、パッチがデプロイモデルにどのように適用されるかを考慮する必要があります。変更可能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したり、自動化によって運用効率を向上させることができます。イミュータブルインフラストラクチャを使用している場合は、CI/CD パイプラインを使用して、オペレーティングシステムの最新バージョンで新しいインフラストラクチャをデプロイし

ます。実際、インフラストラクチャにはパッチを適用するのではなく置き換えられるため、パッチ適用という用語は、このモデルでは間違っています。

テーマ 1: マネージドサービスを使用する

対象となる Essential Eight 戦略
パッチアプリケーション、管理者権限の制限、パッチオペレーティングシステム

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、コンプライアンスの義務を軽減するのに役立ちます。

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 について と責任を共有します。これにより、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、サービスが運用されている施設の物理的なセキュリティまで、 がコンポーネントを 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、コード AWS Lambda やコンテナイメージの脆弱性のスキャンが含まれます。このガイドのすべてのテーマと同様に、モニタリングとコンプライアンスレポートの責任も保持します。Amazon Inspector を使用して、すべての の脆弱性をレポートできます AWS アカウント。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ること AWS Config を確認できます。

例えば、Amazon EC2 インスタンスを実行する場合、責任には以下が含まれます。

- アプリケーションコントロール
- パッチ適用アプリケーション
- Amazon EC2 コントロールプレーンとオペレーティングシステム (OS) への管理者権限の制限
- 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 関数を実行すると、責任が軽減され、以下が含まれます。

- アプリケーションコントロール
- ライブラリがup-to-dateであることを確認する

- Lambda コントロールプレーンへの管理者権限の制限
- 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 関数のコードと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

パッチ適用を有効にする

- Amazon RDS 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に注意してください

脆弱性のスキャン

- Amazon Inspector を使用して Amazon Elastic Container Registry (Amazon ECR) コンテナイメージをスキャンする
- Amazon Inspector で Lambda 関数をスキャンする

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を評価する

- 複数のアカウントを管理する

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を通じてイミュータブルなインフラストラクチャを管理する

対象となる Essential Eight 戦略

アプリケーションコントロール、パッチアプリケーション、パッチオペレーティングシステム

イミュータブルなインフラストラクチャでは、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 著名エンジニアの Colm MacC árthaigh は、2022 AWS re:Invent カンファレンスの「ゼロ特権運用: データにアクセスできないサービスの実行YouTube」(ビデオ) 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、すべてのリソースを承認、保護、および自動化されたパイプラインを通じてデプロイまたは変更することを要求できます。通常、デプロイパイプラインをホストするアカウントにのみユーザーがアクセスできるようにする AWS Identity and Access Management (IAM) ポリシーを作成します。また、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 ポリシーを設定します。手動による変更を防ぐには、セキュリティグループを使用して、サーバーへの SSH およびWindowsリモートデスクトッププロトコル (RDP) 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 (AMIs) とコンテナイメージは、安全かつ繰り返し構築する必要があります。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が組み込まれた AMIs を構築できます。アプリケーションコントロールの詳細については、ACSC ウェブサイトの「アプリケーションコントロールの実装」を参照してください。Image Builder を使用してコンテナイメージを構築し、Amazon Elastic Container Registry (Amazon ECR) を使用してアカウント間でイメージを共有することもできます。中央セキュリティチームは、これらの AMIsとコンテナイメージを構築する自動プロセスを承認し、結果の AMI またはコンテナイメージがアプリケーションチームによって使用を承認されるようにできます。

アプリケーションは、 AWS CloudFormationや 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AWS Cloud Development Kit (AWS CDK)。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。

と同様にテーマ 1: マネージドサービスを使用する、Amazon Inspector は 全体の脆弱性をレポートできます AWS アカウント。一元化されたクラウドチームとセキュリティチームは、この情報を使用して、アプリケーションチームがセキュリティとコンプライアンスの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て報告するには、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 のみが使用されていること、および 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 とコンテナビルドパイプラインを実装する

- EC2 Image Builder を使用して、AMIs に以下を構築します。
  - 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
  - ログ記録に使用される Amazon CloudWatch エージェント
- すべての EC2 インスタンスについて、Systems Manager がインスタンスにアクセスするため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`ポリシーを含めます。
- 組織全体と AMIs 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が最新の AMIs を参照し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

セキュアなアプリケーションビルドパイプラインを実装する

- EC2 Image Builder や AWS CodePipeline (AWS ブログ記事) を使用するなど、IaC のビルドパイプラインを実装する
- CI/CD パイプラインで AWS CloudFormation Guard、cfn-nag (GitHub)、または 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
  - パスワードリテラル
- 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 で を使用して (AWS ブログ記事) 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
- パイプラインを使用する、またはパイプラインに変更を加えるための最小特権アクセスのための IAM ポリシーとサービスコントロールポリシーを設定する

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、AMIsの AMI をスキャンします。

  - EC2 Image Builder (GitHub) で AMIs のライフサイクルを管理する GitHub
- 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
- セキュリティ検出結果の優先順位付けと修正を行う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 とログを継続的にモニタリングする

- IAM ポリシーを定期的に見直して、以下を確認してください。
  - 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
  - 承認されたサービスのみがデータに直接アクセスできる
  - ユーザーがリソースや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、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
- のルートユーザー認証情報 AWS アカウント が使用された場合に通知す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: 自動化による変更可能なインフラストラクチャの管理

対象となる Essential Eight 戦略

アプリケーションコントロール、パッチアプリケーション、パッチオペレーティングシステム

イミュータブルインフラストラクチャと同様に、変更可能なインフラストラクチャを IaC として管理し、自動化プロセスを通じてこのインフラストラクチャを変更または更新します。イミュータブルインフラストラクチャの実装手順の多くは、ミュータブルインフラストラクチャにも適用されます。ただし、変更可能なインフラストラクチャでは、手動でコントロールを実装して、変更されたワークロードが引き続きベストプラクティスに従っていることを確認する必要があります。

変更可能なインフラストラクチャでは、の一機能であるパッチマネージャーを使用してパッチ管理を自動化できます AWS Systems Manager。 AWS 組織内のすべてのアカウントで Patch Manager を有効にします。

SSH および RDP への直接アクセスを禁止し、ユーザーが Session Manager または Run Command を使用することを要求します。これは Systems Manager の機能でもあります。SSH や RDP とは異なり、これらの機能はシステムアクセスと変更を記録できます。

コンプライアンスをモニタリングして報告するには、パッチコンプライアンスを継続的に確認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

パッチ適用の自動化

- 「組織内のすべてのアカウントでパッチマネージャーを有効にする」の手順を実装します。 AWS
- すべての EC2 インスタンスについて、Systems Manager がインスタンスにアクセスするために使用するインスタンスプロファイルまたは IAM ロールAmazonSSMManagedInstanceCoreに CloudWatchAgentServerPolicyと を含めます。

手動プロセスではなくオートメーションを使用する

- 「 での AMI とコンテナビルドパイプラインの実装」のガイダンスを実装する テーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する
- 直接 SSH または RDP アクセスの代わりに Session Manager または Run Command を使用する https://docs.aws.amazon.com/systems-manager/latest/userguide/run-command.html

オートメーションを使用して EC2 インスタンスに以下をインストールする

- AWS Systems Manager エージェント (SSM Agent)。インスタンスの検出と管理に使用されます。
- 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
- ログ記録に使用される Amazon CloudWatch エージェント

リリース前にピアレビューを使用して、変更がベストプラクティスを満たしていることを確認する

- ワイルドカードを使用するポリシーなど、許可が広すぎる 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
- パスワードリテラル
- 安全な IAM ポリシー

ID レベルのコントロールを使用する

- ユーザーが自動プロセスを通じてリソースを変更し、手動設定を防ぐように要求するには、ユーザーが引き受けることができるロールに読み取り専用アクセス許可を付与します。
- Systems Manager が使用するロールなど、サービスロールにのみリソースを変更する許可を付与する

脆弱性スキャンの実装

- 「 で脆弱性スキャンを実装する」のガイダンスを実装する テーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する
- 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

パッチコンプライアンスを継続的にモニタリングする

- オートメーションとダッシュボードを使用してパッチコンプライアンスを報告する
- パッチコンプライアンスのダッシュボードを確認するメカニズムを実装する

IAM とログを継続的にモニタリングする

- IAM ポリシーを定期的に見直して、以下を確認してください。
  - 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
  - 承認されたサービスのみがデータに直接アクセスできる
  - ユーザーがリソースや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、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。
- AWS Identity and Access Management Access Analyzer 検出結果を定期的に確認する
- のルートユーザー認証情報 AWS アカウント が使用された場合に通知す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 を管理する

対象となる Essential Eight 戦略

管理者権限の制限、多要素認証

アイデンティティとアクセス許可の堅牢な管理は、クラウドでセキュリティを管理する上で重要な側面です。強力なアイデンティティプラクティスは、必要なアクセスと最小特権のバランスを取ります。これにより、開発チームはセキュリティを損なうことなく迅速に作業できます。

ID フェデレーションを使用して、ID の管理を一元化します。これにより、1 つの場所からアクセスを管理できるため、複数のアプリケーションやサービス間のアクセスを簡単に管理できます。これにより、一時的なアクセス許可と多要素認証 (MFA) を実装することもできます。

タスクの実行に必要なアクセス許可のみをユーザーに付与します。 AWS Identity and Access Management Access Analyzer はポリシーを検証し、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できます。 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 Identity Center 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。

どのような種類の認証を行う場合も、一時的な認証情報を使用して、認証情報が誤って開示、共有、盗難されるなどのリスクを軽減または排除することをお勧めします。IAM ユーザーの代わりに IAM ロールを使用します。

MFA などの強力なサインインメカニズムを使用して、サインイン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します。ルートユーザーに MFA を要求し、フェデレーションレベルで要求することもできます。IAM ユーザーの使用が避けられない場合は、MFA を適用します。

コンプライアンスをモニタリングして報告するには、継続的にアクセス許可の削減、IAM Access Analyzer からの検出結果のモニタリング、未使用の IAM リソースの削除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強制され、認証情報が有効期間が短く、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 フェデレーションを実装する

- 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 プロバイダーとフェデレーションするよう要求する
- 環境への一時的な昇格アクセスを実装する AWS

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取得する
- 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、アイデンティティベースのポリシーが付与できるアクセス許可の上限を設定する
- 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 IAM ロールを使用して にアクセスするように要求する AWS
- 未使用の 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 を強制する

- ルートユーザーに MFA を要求する
- IAM アイデンティティセンターを通じて 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

最小特権アクセスをモニタリングする

- IAM Access Analyzer の検出結果を に送信する AWS Security Hub
- 重要な IAM Identity Center の検出結果の通知の設定を検討する
- 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

対象となる Essential Eight 戦略

管理者権限を制限する

データ境界は、環境 AWS 内の一連の予防ガードレールであり、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していることを確認できます。これらのガードレールは、幅広い AWS アカウント およびリソースの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。これらの組織全体のガードレールは、既存のきめ細かなアクセスコントロールを置き換えるものではありません。代わりに、すべての AWS Identity and Access Management (IAM) ユーザー、ロール、およびリソースが、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。

通常、 で作成される組織の境界の外部からのアクセスを防止するポリシーを使用して、データ境界を確立できます AWS Organizations。データ境界を確立するために使用される 3 つの主要な境界認可条件は次のとおりです。

- 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 動作するプリンシパル。
- 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 のネットワーク。

`OFFICIAL:SENSITIVE` や などの異なるデータ分類や、開発、テスト`PROTECTED`、本番稼働などの異なるリスクレベルの環境間で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 データの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 のみに リソースへのアクセスを許可する – 条件キー aws:PrincipalOrgIDおよびでリソースベースのポリシーを使用しますaws:PrincipalIsAWSService。これにより、 AWS 組織のプリンシパルと 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。
- ネットワークからのみ信頼できる ID を許可する – 条件キー aws:PrincipalOrgIDおよび で VPC エンドポイントポリシーを使用しますaws:PrincipalIsAWSService。これにより、 AWS 組織のプリンシパルと のプリンシパルのみが VPC AWS エンドポイントを介して のサービスにアクセスできます。

リソースコントロールの実装

- ID が信頼できるリソースにのみアクセスできるようにする – 条件キー でサービスコントロールポリシー (SCPs) を使用しますaws:ResourceOrgID。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。
- ネットワークからのみ信頼されたリソースへのアクセスを許可する – 条件キー で VPC エンドポイントポリシーを使用しますaws:ResourceOrgID。これにより、ID は組織 AWS の一部である 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の実装

- ID が予想されるネットワークからのみリソースにアクセスできるようにする – 条件キー aws:SourceIp、aws:SourceVpc、、aws:SourceVpceおよび で SCPs を使用しますaws:ViaAWSService。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のみ、および を介してリソースにアクセスできます AWS のサービス。
- 予想されるネットワークから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および でリソースベースのポリシーを使用しますaws:PrincipalIsAWSService。これにより、リソースへのアクセスが許可されるのは、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、または呼び出し元の ID が AWS のサービスである場合のみです 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

- SCPs、VPC エンドポイントポリシーを確認するメカニズムを実装する

次の AWS Config ルールを実装する

- `SERVICE_VPC_ENDPOINT_ENABLED`

テーマ 6: バックアップを自動化する

対象となる Essential Eight 戦略

定期的なバックアップ

「失敗は与えられたものであり、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 TCP パケットを破損するメモリユニット、一時的な エラーから永続的な障害まで、最終的にはすべてが時間の経過とともに失敗します。これは、最高品質のハードウェアを使用しているか、低コストのコンポーネントを使用しているかにかかわらず、与えられたものです。」 —Werner Vogels、CTO、Amazon、All Things Distributed

データのバックアップと復旧は、システムの信頼性の重要な部分です。 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、バックアップされたデータの復旧を確実に行うように設計されています。

AWS Backup は、データのバックアップを一元化および自動化するフルマネージドサービスです AWS のサービス。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アクティブで、猶予期間が終わると、ユーザー、アカウント、データ所有者、または がボールト設定を変更または削除することはできません AWS。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 99.999999999% の年間耐久性を提供 AWS Backup できます。これは 11 ナインとも呼ばれます。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とリカバリを自動化する

- でのデータバックアップの実装 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)

結果全体にガバナンスを実装する AWS Backup

- 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上位 10 件 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
- Audit Manager AWS Backup 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

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: ログ記録とモニタリングを一元化する

対象となる Essential Eight 戦略

アプリケーションコントロール、アプリケーションへのパッチ適用、管理者権限の制限、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介して行われた AWS API コールなど AWS Management Console、アカウントの API コールの履歴を作成することで、 AWS デプロイをモニタリングするのに役立ちます。 AWS SDKs CloudTrail をサポートするサービスの場合、サービスの API を呼び出したユーザーとアカウント、呼び出し元のソース IP アドレス、および呼び出しの発生日時を特定することもできます。
- Amazon CloudWatch は、 AWS リソースと AWS で実行しているアプリケーションのメトリクスを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、ログを分析して処理し、 AWS 環境内の予期しないアクティビティや不正なアクティビティの可能性を特定する継続的なセキュリティモニタリングサービスです。GuardDuty は Amazon EventBridge と統合して、自動応答を開始したり、人間に通知したりします。
- AWS Security Hub 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環境をチェックするのに役立ちます。

これらのツールと機能は、可視性を高め、環境に悪影響を及ぼす前に問題に対処するのに役立つように設計されています。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改善し、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 の検出結果のアラートを設定する
- CloudTrail で組織の証跡を作成する

ログ記録セキュリティ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

ログを一元化する

- 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 をアカウントに集中させる (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 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

メカニズムを実装する

- ログの検出結果を確認するメカニズムを確立する

- Security Hub の検出結果を確認するメカニズムを確立する
- GuardDuty の検出結果に対応するメカニズムを確立する

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

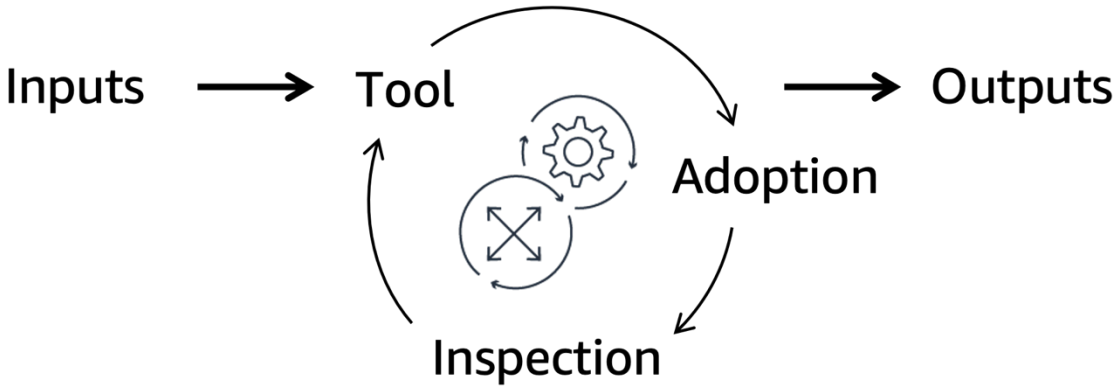
テーマ 8: 手動プロセスのメカニズムを実装する

対象となる Essential Eight 戦略

アプリケーションコントロール、パッチアプリケーション

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) 。つまり、望ましい成果を達成するためには、ベストエフォートを自動化された反復可能なスケーラブルなプロセスとツールに置き換える必要があります。

次の図に示すように、メカニズムは、ツールを作成し、ツールの導入を推進し、調整のために結果を検査する完全なプロセスです。これは、動作中に自身を強化して改善するサイクルです。制御可能な入力を受け取り、継続的な出力に変換して、繰り返されるビジネス上の課題に対処します。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

- コンプライアンスギャップを確認して対処するメカニズムを確立する
- セキュリティポリシーを更新するメカニズムを確立する
- サポートされていないアプリケーションを削除し、ルール拒否リストに追加する AWS Config
- でアクセスポリシーを検証する AWS Identity and Access Management Access Analyzer
- 脆弱性登録を自動的にup-to-date状態に保つ Amazon Inspector を有効にする
- 少なくとも、アプリケーションコントロールルールセットを毎年確認する
- 手動プロセスの負担を軽減するために、AWS Config ルールなどの自動化の実装を検討する
- AWS Systems Manager インベントリを使用して、ソフトウェアポリシーで必要なソフトウェアを実行しているインスタンスを可視化することを検討する

このテーマのモニタリング

- コンプライアンス、ギャップの検査、メカニズムの評価など、目標に向けた進捗状況を追跡できるのエグゼクティブスポンサーを監督します。

で Essential Eight 成熟度に到達するための示唆的なケーススタディ AWS

この章では、Essential Eight の成熟度を対象とする政府機関のケーススタディを示します AWS。

この章のセクション：

シナリオとアーキテクチャの概要

政府機関には、次の 3 つのワークロードがあります AWS クラウド。

- 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
- Amazon Elastic Container Service (Amazon ECS) で実行され、Amazon Relational Database Service (Amazon RDS) のデータベースを使用するコンテナ化されたウェブサービス
- Amazon EC2 で実行されている市販off-the-shelf (COTS) ソフトウェア

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 環境のコアサービスを提供します。各ワークロードは、開発者チームまたはデリバリーチームとも呼ばれる個別のアプリケーションチームによって所有されます。

コアアーキテクチャ

クラウドチームは、 で次の機能を既に確立しています AWS クラウド。

- ID Microsoft フェデレーション AWS IAM Identity Center は、Entra ID (以前の Azure Active Directory) インスタンスにリンクします。フェデレーションは、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を強制します。

- 一元化された AMI パイプラインは、EC2 Image Builder で OSsとコアアプリケーションにパッチを適用するために使用されます。
- Amazon Inspector は脆弱性を識別でき、すべてのセキュリティ検出結果が Amazon GuardDuty に送信され、一元管理されます。
- 確立されたメカニズムは、アプリケーションコントロールルールの更新、サイバーセキュリティイベントへの対応、コンプライアンスギャップの確認に使用されます。
- AWS CloudTrail はログ記録とモニタリングに使用されます。
- ルートユーザーのログインなどのセキュリティイベントは、アラートを開始します。
- SCPsと VPC エンドポイントポリシーは、環境のデータ境界を確立します AWS 。
- SCPs、アプリケーションチームが CloudTrail や などのセキュリティサービスとログ記録サービスを無効にできないようにします AWS Config。
- AWS Config の結果は、 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 の例ですテーマ 1: マネージドサービスを使用する。

データレイクは、ストレージに Amazon S3 を使用し、ETL AWS Lambda に 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。このパイプラインはアプリケーションチームに制限されています。アプリケーションチームがコードリポジトリのプルリクエストを行うと、2 人称ルールが使用されます。

このワークロードでは、アプリケーションチームは Essential Eight 戦略に対応するために以下のアクションを実行します。

アプリケーションコントロール

- アプリケーションチームは、GuardDuty で Lambda Protection を有効にし、Amazon Inspector で Lambda スキャンを有効にします。
- アプリケーションチームは、Amazon Inspector 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。

パッチアプリケーション

- アプリケーションチームは、Amazon Inspector で Lambda スキャンを有効にし、廃止されたライブラリまたは脆弱なライブラリの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。

管理者権限を制限する

- コアアーキテクチャ「」セクションで説明されているように、アプリケーションチームは、デプロイパイプラインの承認ルールを通じて、本番環境のデプロイへのアクセスを既に制限しています。
- アプリケーションチームは、コアアーキテクチャ「」セクションで説明されている一元化された ID フェデレーションと一元化されたログ記録ソリューションに依存しています。
- アプリケーションチームは、 AWS CloudTrail 証跡と 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 のデプロイと AWS CloudFormation スタックの削除に関する Amazon Simple Notification Service (Amazon SNS) アラートを設定します。

パッチオペレーティングシステム

- アプリケーションチームは、Amazon Inspector で Lambda スキャンを有効にし、廃止されたライブラリまたは脆弱なライブラリのアラートを設定します。

多要素認証

- アプリケーションチームは、コアアーキテクチャ「」セクションで説明されている一元化された ID フェデレーションソリューションに依存しています。このソリューションは、疑わしい MFA イベントに対して MFA を適用し、認証とアラートをログに記録し、自動的に応答します。

定期的なバックアップ

- 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。
- アプリケーションチームは、バージョニングと Amazon S3 オブジェクトロックを有効にして、オブジェクトの削除や変更を防止します。

- アプリケーションチームは、データセット全体を別のデータセットにレプリケートするのではなく、組み込みの Amazon S3 の耐久性に依存しています AWS リージョン。
- 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 の例ですテーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する。

ウェブサービスは Amazon ECS で実行され、Amazon RDS のデータベースを使用します。アプリケーションチームは AWS 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 で作成され、Amazon ECR に保存されます。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。このパイプラインはアプリケーションチームに制限されています。アプリケーションチームがコードリポジトリのプルリクエストを行うと、2 人称ルールが使用されます。

このワークロードでは、アプリケーションチームは Essential Eight 戦略に対応するために以下のアクションを実行します。

アプリケーションコントロール

- アプリケーションチームは、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。
- アプリケーションチームは、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「アプリケーションコントロールの実装」を参照してください。
- アプリケーションチームは、出力を Amazon CloudWatch Logs に記録するように Amazon ECS タスク定義を設定します。
- アプリケーションチームは、Amazon Inspector 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。

パッチアプリケーション

- アプリケーションチームは、Amazon Inspector で Amazon ECR コンテナイメージのスキャンを有効にし、廃止されたライブラリまたは脆弱なライブラリのアラートを設定します。
- アプリケーションチームは、Amazon Inspector の検出結果への応答を自動化します。新しい検出結果は Amazon EventBridge トリガーを介してデプロイパイプラインを開始し、CodePipeline がターゲットです。
- 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。

管理者権限を制限する

- アプリケーションチームは、デプロイパイプラインの承認ルールを通じて、本番環境のデプロイへのアクセスを既に制限しています。
- アプリケーションチームは、認証情報のローテーションと一元的なログ記録のために、一元化されたクラウドチームの ID フェデレーションに依存しています。
- アプリケーションチームは CloudTrail 証跡と 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 CloudFormation スタック削除の Amazon SNS アラートを設定します。

パッチオペレーティングシステム

- アプリケーションチームは、Amazon Inspector で Amazon ECR コンテナイメージのスキャンを有効にし、OS パッチ更新のアラートを設定します。
- アプリケーションチームは、Amazon Inspector の検出結果への応答を自動化します。新しい検出結果は EventBridge トリガーを介してデプロイパイプラインを開始し、CodePipeline がターゲットです。
- アプリケーションチームは Amazon RDS イベント通知をサブスクライブして、更新について通知されるようにします。これらの更新を手動で適用するか、Amazon RDS に自動的に適用するかについて、ビジネス所有者とリスクベースの決定を行います。
- アプリケーションチームは、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

- アプリケーションチームは、コアアーキテクチャ「」セクションで説明されている一元化された ID フェデレーションソリューションに依存しています。このソリューションは、疑わしい MFA イベントに対して MFA を適用し、認証とアラートをログに記録し、自動的に応答します。

定期的なバックアップ

- 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。
- アプリケーションチームは 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) 。自動テストが実行されると、パイプラインはスタックを破棄します。このパイプラインは 1 か月に 1 回自動的に実行され、復旧手順の有効性を検証します。

ワークロードの例: Amazon EC2 の COTS ソフトウェア

このワークロードは の例ですテーマ 3: 自動化による変更可能なインフラストラクチャの管理。

Amazon EC2 で実行されているワークロードは、 を使用して手動で作成されました AWS Management Console。デベロッパーは、EC2 インスタンスにログインし、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ます。

このワークロードでは、クラウドチームとアプリケーションチームは、Essential Eight 戦略に対応するために以下のアクションを実行します。

アプリケーションコントロール

- 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。結果の AMI は、組織内のすべてのアカウントで共有されます。
- 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 OpenSearch Service で実行され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 Amazon CloudWatch Logs 出力を送信します。OpenSearch

- 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キャッチするための独自のメカニズムを実装します。検出結果に対処するための脆弱性管理プログラムの作成に関する詳細なガイダンスについては、「 でのスケーラブルな脆弱性管理プログラム AWSの構築」を参照してください。

パッチアプリケーション

- 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。
- クラウドチームは基本 AMI にパッチを適用し、その AMI が変更されると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、ワークロードが必要とするポートでのみトラフィックを許可するようにセキュリティグループルールを設定することで、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する代わりに、Patch Manager を使用してインスタンスにパッチを適用します。
- EC2 インスタンスの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する必要がある場合は、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。

管理者権限を制限する

- 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を許可す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が制限され、ユーザーは Session Manager を介して EC2 インスタンスにアクセスする必要があります。
- アプリケーションチームは、認証情報のローテーションと一元的なログ記録のために、一元化されたクラウドチームの ID フェデレーションに依存しています。
- アプリケーションチームは CloudTrail 証跡と 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 CloudFormation スタック削除の Amazon SNS アラートを設定します。

パッチオペレーティングシステム

- クラウドチームは基本 AMI にパッチを適用し、その AMI が変更されると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 の一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 Patch Manager を使用して、個々のインスタンスにログインするインスタンスにパッチを適用します。
- EC2 インスタンスの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する必要がある場合は、Session Manager を使用します。

多要素認証

- アプリケーションチームは、コアアーキテクチャ「」セクションで説明されている一元化された ID フェデレーションソリューションに依存しています。このソリューションは、疑わしい MFA イベントに対して MFA を適用し、認証とアラートをログに記録し、自動的に応答します。

定期的なバックアップ

- 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、毎月バックアップ復元を手動で実行するメカニズムを実装します。

リソース

AWS ドキュメント

- AWS セキュリティリファレンスアーキテクチャ (AWS SRA)
- AWS セキュリティドキュメント
- AWS Well-Architected フレームワークのセキュリティの柱

その他の AWS リソース

- AWS クラウドセキュリティ
- AWS クラウド導入フレームワーク (セキュリティの観点から)

オーストラリアサイバーセキュリティセンターのリソース

- Essential Eight Explained
- Essential Eight Maturity Model
- Essential Eight 評価プロセスガイド

寄稿者

本ドキュメントの寄稿者は次のとおりです。

- AWS ソリューションアーキテクチャ、シニアソリューションアーキテクト、James™smill
- AWS ソリューションアーキテクチャ、シニアソリューションアーキテクト、Chris Harding
- Jess Modini、Advisory Solutions Architect、 AWS Solutions Architecture
- Justin Bowden、セキュリティ保証プリンシパル、 AWS セキュリティ保証
- AWS ソリューションアーキテクチャ、シニアソリューションアーキテクト、Rob Powell
- Tony Mihaljevic、シニアクラウドアーキテクト、 AWS Professional Services
- AWS グローバルサービスセキュリティ、プリンシパルセキュリティアドバイザー、Volker Rath

付録: Essential Eight コントロールマトリックス

次の表は、Essential Eight 戦略を AWS Well-Architected フレームワークの AWS 実装ガイダンスと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 コントロールについては AWS クラウド、表に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 :

- アプリケーションコントロール
- パッチアプリケーション
- Microsoft Office マクロ設定を構成する
- ユーザーアプリケーションの強化
- 管理者権限を制限する
- パッチオペレーティングシステム
- 多要素認証
- 定期的なバックアップ

アプリケーションコントロール

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
アプリケーションコントロールは、ワークステーションとサーバーに実装され、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 HTML、HTML アプリケーション、コントロールパネルのアプ	テーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する: AMI とコンテナビルドパイプラインを実装する	EC2 Image Builder を使用して以下を組み込みます。 • 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 強化された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
レット、ドライバーの実行を、組織が承認したセットに制限します。		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AMIs を共有する アプリケーションチームが最新の AMIs を参照し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」が実装されています。 Microsoftの「推奨ドライバーブロックルール」が実装されています。	「アプリケーションコントロールの実装」(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
アプリケーションコントロールルールセットは、年単位またはより頻繁に検証されます。	テーマ 8: 手動プロセスのメカニズムを実装する: セキュリティポリシーを更新するメカニズムを実装する	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た実行とブロックされた実行は一元的に記録され、不正な変更や削除から保護され、侵害の兆候がないか監視され、サイバーセキュリティイベントが検出されたときに対処されます。	テーマ 7: ログ記録とモニタリングを一元化する: ログ記録を有効にする	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 の検出結果の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 Amazon S3 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: ログ記録とモニタリングを一元化する: ログ記録のセキュリティ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: ログ記録とモニタリングを一元化する: ログを一元化する	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 アカウント内の CloudWatch Logs を一元化して監査と分析を行う (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 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: 手動プロセスのメカニズムを実装する: コンプライアンスギャップを確認して対処するメカニズムを実装する	手動プロセスの負担を軽減するために、AWS Config ルールなどの自動化の実装を検討する	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

パッチアプリケーション

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
アセット自動検出方法は、少なくとも 2 か月ごとに使用され、その後の脆弱性スキャンアクティビティのアセットの検出をサポートします。	テーマ 1: マネージドサービスを使用する: 脆弱性のスキャン テーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する: 脆弱性スキャンを実装する テーマ 3: 自動化による変更可能なインフラストラクチャの管	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 セキュリティ検出結果の優先順位付けと修正を行う脆弱性管	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
	理: 脆弱性スキャンを実装する	理プログラムを構築する	
	テーマ 7: ログ記録とモニタリングを一元化する: ログを一元化する	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 アカウント内の CloudWatch Logs を一元化して監査と分析を行う (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ブログ記事) AWS Security Hub 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
up-to-date脆弱性データベースを持つ脆弱性スキャナーは、脆弱性スキャンアクティビティに使用されます。	テーマ 1: マネージドサービスを使用する: 脆弱性のスキャン テーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する: 脆弱性スキャンを実装する テーマ 3: 自動化による変更可能なインフラストラクチャの管理: 脆弱性スキャンを実装する	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 セキュリティ検出結果の優先順位付けと修正を行う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
脆弱性スキャナーは、インターネット向けサービスのセキュリティ脆弱性に関する欠落しているパッチや更新プログラムを特定するために、少なくとも毎日使用されます。			
脆弱性スキャナーは、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関する欠落しているパッチや更新プログラムを特定するために、少なくとも毎週使用されます。	「技術例: パッチアプリケーション」(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
脆弱性スキャナーは、他のアプリケーションのセキュリティ脆弱性に関する欠落しているパッチや更新プログラムを特定するために、少なくとも 2 週間使用されます。	テーマ 1: マネージドサービスを使用する: 脆弱性のスキャン テーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する: 脆弱性スキャンを実装する テーマ 3: 自動化による変更可能なインフラストラクチャの管理: 脆弱性スキャンを実装する	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 セキュリティ検出結果の優先順位付けと修正を行う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するパッチ、更新、またはベンダー緩和は、リリースから 2 週間以内、または悪用が存在する場合は 48 時間以内に適用されます。	テーマ 1: マネージドサービスを使用する: 脆弱性のスキャン テーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する: 脆弱性スキャンを実装する テーマ 3: 自動化による変更可能なインフラストラクチャの管理: 脆弱性スキャンを実装する	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 セキュリティ検出結果の優先順位付けと修復を行う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: 自動化による変更可能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	組織内のすべてのアカウントで Patch Manager AWS 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するパッチ、更新、またはベンダー緩和は、リリースから 2 週間以内、または悪用が存在する場合は 48 時間以内に適用されます。	「技術例: パッチアプリケーション」(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するパッチ、更新、またはベンダーの緩和策は、リリースから 1 か月以内に適用されます。	テーマ 1: マネージドサービスを使用する: 脆弱性のスキャン テーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する: 脆弱性スキャンを実装する テーマ 3: 自動化による変更可能なインフラストラクチャの管理: 脆弱性スキャンを実装する	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 セキュリティ検出結果の優先順位付けと修復を行う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: 自動化による変更可能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	組織内のすべてのアカウントで Patch Manager AWS 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
ベンダーでサポートされなくなったアプリケーションは削除されます。	テーマ 8: 手動プロセスのメカニズムを実装する: コンプライアンスギャップを確認して対処するメカニズムを実装する	AWS Systems Manager インベントリを使用して、ソフトウェアポリシーで必要なソフトウェアを実行しているインスタンスを可視化することを検討する	SEC06-BP02 強化された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 マクロ設定を構成する

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
Microsoft Office マクロは、ビジネス要件が実証されていないユーザーに対して無効になります。	「技術例: マクロ設定の構成」(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
サンドボックス化された環境、信頼されたロケーション内、または信頼されたパブリッシャーによってデジタル署名されたMicrosoft Officeマクロのみを実行できます。			
Trusted Locations 内のコンテンツを書き込んだり変更したりできるのは、Microsoft Officeマクロに悪意のあるコードがないことを検証する権限を持つユーザーのみです。			
Microsoft Office 信頼できないパブリッシャーによってデジタル署名された マクロは、メッセージバーまたはバックス			

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
テージビューを介して有効にすることはできません。			
Microsoft Officeの信頼できるパブリッシャーのリストは、年単位またはより頻繁に検証されます。			
Microsoft Office インターネットから送信されるファイルの マクロはブロックされます。			
Microsoft Office マクロウイルス対策スキャンが有効になっています。			
Microsoft Office マクロは Win32 API コールの実行をブロックされます。			
Microsoft Office マクロセキュリティ設定は、ユーザーが変更することはできません。			

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
許可されたマクロ実行とブロックされた Microsoft Officeマクロ実行は一元的にログに記録され、不正な変更や削除から保護され、侵害の兆候がないか監視され、サイバーセキュリティイベントが検出されたときに対処されます。			

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
ウェブブラウザはインターネットJavaから処理しません。	「技術例: ユーザーアプリケーションの強化」(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
ウェブブラウザは、インターネットからのウェブ広告を処理しません。			
Internet Explorer 11 は無効または削除されています。			

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
Microsoft Office は子プロセスの作成をブロックされます。			
Microsoft Office は、実行可能コンテンツの作成をブロックされます。			
Microsoft Office は、他のプロセスへのコードの挿入をブロックされます。			
Microsoft Office は、OLE パッケージのアクティブ化を防ぐように設定されています。			
PDF ソフトウェアは子プロセスの作成をブロックされます。			
ウェブブラウザ用の ACSC またはベンダー強化ガイダンスMicrosoft OfficeとPDF ソフトウェアが実装されています。			

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と PDF ソフトウェアのセキュリティ設定は、ユーザーが変更することはできません。			
.NET Framework 3.5 (2.NET.0 と 3.0 を含む) は無効または削除されます。			
Windows PowerShell 2.0 は無効または削除されています。			
PowerShell は、制約言語モードを使用するように設定されています。			
ブロックされたPowerShellスクリプトの実行は一元的に記録され、不正な変更や削除から保護され、侵害の兆候がないか監視され、サイバーセキュリティイベントが検出されたときに対処されます。			

管理者権限を制限する

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のリクエストは、最初にリクエストされたときに検証されます。	テーマ 4: ID を管理する: ID フェデレーションを実装する	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 プロバイダーとフェデレーションするよう要求する	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、再検証されない限り、12 か月後に自動的に無効になります。	テーマ 4: ID を管理する: ID フェデレーションを実装する	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 プロバイダーとフェデレーションするよう要求する	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 を管理する: 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に要求する AWS 未使用の 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 AWS Summit ANZ 2023: クラウドでの一時的な認証情報への	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
		ジャーニー (YouTube ビデオ)	
システムおよびアプリケーションへの特権アクセスは、45 日間非アクティブ状態になると自動的に無効になります。	テーマ 4: ID を管理する: ID フェデレーションを実装する テーマ 4: ID を管理する: 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 プロバイダーとフェデレーションするよう要求する 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に要求する AWS 未使用の 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 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 ビデオ)	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、ユーザーおよびサービスが職務を果たすために必要なもののみに制限されます。	テーマ 4: ID を管理する: 最小特権の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取得する 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、アイデンティティベースのポリシーが付与できるアクセス許可の上限を設定する	SEC01-BP02 アカウントのルートユーザーとプロパティを保護する 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	
特権アカウントは、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ません。	「技術例: 管理者権限の制限」(ACSC ウェブサイト) を参照してください。	インターネットにアクセスできない VPC が取得できないようにする SCP の実装を検討してください。	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
特権ユーザーは、特権運用環境と非特権運用環境を別々に使用します。	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 や などの異なるデータ分類や、開発、テストPROTECTED 、本番稼働などの異なるリスクレベルの環境間で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
特権運用環境は、特権のない運用環境内では仮想化されません。			
権限のないアカウントは、権限のある運用環境にログオンできません。			
特権アカウント (ローカル管理者アカウントを除く) は、特権のない運用環境にログオンできません。			
Just-in-time管理は、システムとアプリケーションの管理に使用されます。	テーマ 4: ID を管理する: ID フェデレーションを実装する	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 プロバイダーとフェデレーションするよう要求する 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通じて実施されます。	テーマ 1: マネージドサービスを使用する テーマ 3: 自動化による変更可能なインフラストラクチャの管理: 手動プロセスではなくオートメーションを使用する	直接 SSH または RDP アクセスの代わりに Session Manager または Run Command を使用する https://docs.aws.amazon.com/systems-manager/latest/userguide/run-command.html	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、管理されています。	「技術例: 管理者権限の制限」(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が有効になっています。			

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は一元的に記録され、不正な変更や削除から保護され、侵害の兆候がないか監視され、サイバーセキュリティイベントが検出されたときに対処されます。	テーマ 7: ログ記録とモニタリングを一元化する: ログ記録を有効にする テーマ 7: ログ記録とモニタリングを一元化する: ログを一元化する	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 アカウント内の CloudWatch Logs を一元化して監査と分析を行う (AWS ブログ記事) Amazon Inspector の管理を一元化する Security Hub 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ブログ記事) AWS 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
特権アカウントとグループへの変更は一元的に記録され、不正な変更や削除から保護され、侵害の兆候がないか監視され、サイバーセキュリティイベントが検出されたときに対処されます。			

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	

パッチオペレーティングシステム

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のオペレーティングシステムのセキュリティ脆弱性に対するパッチ、更新、またはベンダーの緩和策は、リリースから 2 週間以内、または悪用が存在する場合は 48 時間以内に適用されます。	テーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する: AMI とコンテナビルドパイプラインを実装する	EC2 Image Builder を使用して以下を組み込みます。 • 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AMIs 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が最新の AMIs を参照し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を使用する: パッチ適用を有効にする テーマ 3: 自動化による変更可能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	組織内のすべてのアカウントで Patch Manager AWS 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のオペレーティングシステムのセキュリティ脆弱性に対するパッチ、更新、またはベンダーの緩和策は、リリースから 2 週間以内、または悪用が存在する場合は 48 時間以内に適用されます。	テーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する: AMI とコンテナビルドパイプラインを実装する	EC2 Image Builder を使用して以下を組み込みます。 • 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AMIs を共有する アプリケーションチームが最新の AMIs を参照し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 強化された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を使用する: パッチ適用を有効にする テーマ 3: 自動化による変更可能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	組織内のすべてのアカウントで Patch Manager AWS 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
脆弱性スキャナーは、インターネット向けサービスのオペレーティングシステムでセキュリティの脆弱性に関する欠落しているパッチや更新プログラムを特定するために、少なくとも毎日使用されます。	テーマ 1: マネージドサービスを使用する: 脆弱性のスキャン テーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する: 脆弱性スキャンを実装する テーマ 3: 自動化による変更可能なインフラストラクチャの管理: 脆弱性スキャンを実装する	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 セキュリティ検出結果の優先順位付けと修復のための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 強化された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
脆弱性スキャナーは、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でセキュリティの脆弱性に関する欠落しているパッチや更新プログラムを特定するために、少なくとも週 1 回使用されます。			

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
オペレーティングシステムの最新リリースまたは以前のリリースは、ワークステーション、サーバー、ネットワークデバイスに使用されます。	テーマ 2: 安全なパイプラインを通じてイミュータブルなインフラストラクチャを管理する: 脆弱性スキャンを実装する	EC2 Image Builder を使用して以下を組み込みます。 • 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AMIs を共有する アプリケーションチームが最新の AMIs を参照し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 強化された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
ベンダーでサポートされなくなったオペレーティングシステムは置き換えられます。			

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
多要素認証は、組織のユーザーが組織のインターネット向けサービスに対して認証する場合に使用されます。	テーマ 4: ID を管理する: ID フェデレーションを実装する	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 プロバイダーとフェデレーションするよう要求する 環境への一時的な昇格アクセスを実装する AWS	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 を管理する: MFA を強制する	ルートユーザーに MFA を要求する を通じて MFA を要求する AWS IAM Identity Center 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
多要素認証は、組織のユーザーが組織の機密データを処理、保存、または通信するサードパーティーのインターネット向けサービスに対して	「多要素認証の実装」(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
認証する場合に使用されます。			
多要素認証 (利用可能な場合) は、組織のユーザーが、組織の機密データを処理、保存、または通信するサードパーティーのインターネット向けサービスに対して認証する場合に使用されます。			
多要素認証は、組織のインターネット向けサービスに対して認証する場合、組織以外のユーザー (オプトアウトを選択できます) に対してデフォルトで有効になります。			
多要素認証は、システムの特権ユーザーを認証するために使用されます。	テーマ 4: ID を管理する: ID フェデレーションを実装する	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 プロバイダーとフェデレーションするよう要求する 環境への一時的な昇格アクセスを実装する AWS	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 を管理する: MFA を強制する	ルートユーザーに MFA を要求する IAM アイデンティティセンターを通じて 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
多要素認証は、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ために使用されます。	テーマ 4: ID を管理する: MFA を強制する	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
多要素認証は、検証用のなりすましに対する耐性があり、ユーザーが持っているものとユーザーが知っているもの、またはユーザーが知っているか知っているものによって口ック解除されているもののいずれかを使用します。	「多要素認証の実装」(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は一元的に記録され、不正な変更や削除から保護され、侵害の兆候がないか監視され、サイバーセキュリティイベントが検出されたときに対処されます。	テーマ 7: ログ記録とモニタリングを一元化する: ログ記録を有効にする テーマ 7: ログ記録とモニタリングを一元化する: ログを一元化する	アカウント内の CloudWatch Logs を一元化して監査と分析を行う (AWS ブログ記事) Amazon Inspector の管理を一元化する Security Hub 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ブログ記事) AWS 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

定期的なバックアップ

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、および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調整され、回復力のある方法で実行および保持されます。	テーマ 6: バックアップを自動化する: データのバックアップとリカバリを自動化する	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
システム、ソフトウェア、およびバックアップからの重要なデータの復元は、ディザスタリカバリの演習の一環として、調整された方法でテストされます。	テーマ 6: バックアップを自動化する: データのバックアップとリカバリを自動化する テーマ 6: バックアップを自動化する: AWS Backup 結果全体にガバナンスを実装する	(ブログ記事) でデータ復旧の検証を自動化する AWS BackupAWS 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
非特権アカウント、および特権アカウント (バックアップ管理者を除く) は、バックアップにアクセスできません。	テーマ 6: バックアップを自動化する: 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上位 10 件 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
権限のないアカウント、および権限のあるアカウント (バックアップブレークグラスアカウントを除く) は、バックアップを変更または削除することはできません。		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	

注意

お客様は、この文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ており、(b) 通知なしに変更される可能性がある現在の AWS 製品提供および慣行を表し、(c) AWS およびその関連会社、サプライヤー、または許諾者からのいかなる約束または保証も作成しません。 AWS 製品またはサービスは、明示または黙示を問わず、いかなる種類の保証、表明、条件もなしに「現状のまま」提供されます。顧客 AWS に対する の責任は契約によって AWS 管理され、本書は AWS と顧客との間の契約の一部でも変更もされません。

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します。
- 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: でオンプレミスの Oracle データベースを Oracle 用 Amazon Relational Database Service (Amazon RDS) に移行します AWS クラウド。
- 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: カスタマーリレーションシップ管理 (CRM) システムを Salesforce.com に移行します。
- 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: オンプレミスの Oracle データベースを の EC2 インスタンス上の Oracle に移行します AWS クラウド。
- 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。

- 使用停止 — お客様のソース環境で不要になったアプリケーションを停止または削除します。

A

ABAC

「属性ベースのアクセスコントロール」を参照してください。

抽象化されたサービス

「 マネージドサービス」を参照してください。

ACID

「アトミック性」、「整合性」、「分離」、「耐久性」を参照してください

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。より柔軟ですが、アクティブ/パッシブ移行よりも多くの作業が必要です。

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の方法。移行中、ターゲットデータベースはトランザクションを受け付けません。

集計関数

行のグループで動作し、グループの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 などがありますMAX。

AI

「人工知能」を参照してください。

AIOps

「人工知能オペレーション」を参照してください。

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。

アプリケーションコントロール

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 の需要要素であり、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?」を参照してください。

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

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)

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は、組織がクラウドに成功するための効率的で効果的な計画を立て AWS るのに役立ちます。 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ます。ビジネス、人材、ガバナンスの観点では、ビジネススキルとプロセスに重点を置き、プラットフォーム、セキュリティ、オペレーションの視点は技術的なスキルとプロセスに焦点を当てています。例えば、人材の観点では、人事 (HR)、人材派遣機能、および人材管理を扱うステークホルダーを対象としています。この観点から、 AWS CAF は、クラウド導入を成功させるための組織の準備に役立つ人材開発、トレーニング、コミュニケーションに関するガイダンスを提供します。詳細については、AWS CAF ウェブサイト と AWS CAF のホワイトペーパー を参照してください。

AWS ワークロード認定フレームワーク (AWS WQF)

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

B

不正なボット

個人または組織に損害を与えることを目的としたボット。

BCP

「事業継続計画」を参照してください。

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Data in a behavior graph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。エンディアン性も参照してください。

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

2 つの異なる同一の環境を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 (青) 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 (緑) 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、ボットハーダーまたはボットオペレーターとして知られる単一関係者の管理下にあるボットのネットワーク。ボットは、ボットとその影響をスケールするための最もよく知られているメカニズムです。

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なります。詳細については、 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 AWS 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、ホワイトペーパー AWSでのコンテナ化されたマイクロサービスの実行 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。

Canary デプロイ

エンドユーザーへのバージョンのスローリリースと増分リリース。確信が持てば、新しいバージョンをデプロイし、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

「データキャプチャの変更」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します。AWS Fault Injection Service （ AWS FIS ） を使用して、 AWS ワークロードにストレスを与え、その応答を評価する実験を実行できます。

CI/CD

継続的インテグレーションと継続的デリバリーを参照してください。

分類

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

クライアント側の暗号化

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

Cloud Center of Excellence (CCoE)

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

クラウドコンピューティング

リモートデータストレージと IoT デバイス管理に通常使用されるクラウドテクノロジー。クラウドコンピューティングは、一般的にエッジコンピューティングテクノロジー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 に移行するときに通常実行する 4 つのフェーズ AWS クラウド :

- 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。AWS 移行戦略との関連性については、「移行準備ガイド」を参照してください。

CMDB

「設定管理データベース」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または が含まれますBitbucket Cloud。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などのビジュアル形式から情報を分析および抽出する AI の分野。例えば、Amazon SageMaker AI は CV 用の画像処理アルゴリズムを提供します。

設定ドリフト

ワークロードの場合、設定は想定状態から変化します。これにより、ワークロードが非準拠になる可能性があり、通常は段階的かつ意図的ではありません。

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

一元管理とガバナンスを備えた分散型の分散データ所有権を提供するアーキテクチャフレームワーク。

データ最小化

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

データ境界

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

データの前処理

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

データ出所

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

データ件名

データを収集、処理している個人。

データウェアハウス

分析などのビジネスインテリジェンスをサポートするデータ管理システム。データウェアハウスには、通常、大量の履歴データが含まれており、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る。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。

ディープラーニング

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。

デプロイ

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、Implementing security controls on AWSのDetective controls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

スタースキーマでは、ファクトテーブル内の量的データに関するデータ属性を含む小さなテーブル。ディメンションテーブル属性は通常、テキストフィールドまたはテキストのように動作する

離散数値です。これらの属性は、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一般的に使用されます。

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)

災害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に使用する戦略とプロセス。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。例えば、 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

「開発値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、応答時間を短縮できます。

電子データ交換 (EDI)

組織間の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プロセス。

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

星スキーマの中央テーブル。事業運営に関する量的データを保存します。通常、ファクトテーブルには、メジャーを含む列とディメンションテーブルへの外部キーを含む列の 2 つのタイプの列が含まれます。

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で段階的なテストを使用する哲学。これはアジャイルアプローチの重要な部分です。

障害分離の境界

では AWS クラウド、アベイラビリティーゾーン AWS リージョン、コントロールプレーン、データプレーンなどの境界で、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

同様のタスクの実行を求める前に、タスクと必要な出力を示す少数の例を LLM に提供します。この手法は、プロンプトに埋め込まれた例 (ショット) からモデルが学習するコンテキスト内学習のアプリケーションです。少数ショットプロンプトは、特定のフォーマット、推論、またはドメインの知識を必要とする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コントロール」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

段階的なアプローチを使用する代わりに、変更データキャプチャによる継続的なデータレプリケーションを使用して、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)

一般化データとラベル付けされていないデータの大規模なデータセットでトレーニングされている大規模な深層学習ニューラルネットワーク。FMs は、言語の理解、テキストと画像の生成、自然言語の会話など、さまざまな一般的なタスクを実行できます。詳細については、「基盤モデルとは」を参照してください。

G

生成 AI

大量のデータでトレーニングされ、シンプルなテキストプロンプトを使用してイメージ、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る AI モデルのサブセット。詳細については、「生成 AI とは」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 ワークフローはレガシーと見なされ、トランクベースのワークフローはモダンで推奨されるアプローチです。

ゴールデンイメージ

そのシステムまたはソフトウェアの新しいインスタンスをデプロイするためのテンプレートとして使用されるシステムまたはソフトウェアのスナップショット。例えば、製造では、ゴールデンイメージを使用して複数のデバイスにソフトウェアをプロビジョニングし、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、、 AWS Trusted Advisor Amazon Inspector、およびカスタム AWS Lambda チェックを使用して実装されます。

H

HA

「高可用性」を参照してください。

異種混在データベースの移行

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

ハイアベイラビリティ (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に使用されるデータセットから保留される、ラベル付きの履歴データの一部。モデル予測をホールドアウトデータと比較することで、ホールドアウトデータを使用してモデルのパフォーマンスを評価できます。

同種データベースの移行

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

ホットデータ

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

ホットフィックス

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

ハイパーケア期間

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

I

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

「産業用モノのインターネット」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

既存のインフラストラクチャを更新、パッチ適用、または変更する代わりに、本番環境のワークロード用に新しいインフラストラクチャをデプロイするモデル。イミュータブルインフラストラクチャは、本質的に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、予測性が高くなります。詳細については、 AWS 「 Well-Architected フレームワーク」の「イミュータブルインフラストラクチャを使用したデプロイ」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 によって導入された用語で、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ゼーションを指します。

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。

産業分野における 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、「Building an industrial Internet of Things (IIoT) digital transformation strategy」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 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

「モノのインターネット」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)

大量のデータに対して事前トレーニングされた深層学習 AI モデル。LLM は、質問への回答、ドキュメントの要約、テキストの他の言語への翻訳、文の完了など、複数のタスクを実行できます。詳細については、LLMs」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

「ラベルベースのアクセスコントロール」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。エンディアン性も参照してください。

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または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、不正アクセスにつながる可能性があります。マルウェアの例としては、ウイルス、ワーム、ランサムウェア、トロイの木馬、スパイウェア、キーロガーなどがあります。

マネージドサービス

AWS のサービス はインフラストラクチャレイヤー、オペレーティングシステム、プラットフォーム AWS を運用し、エンドポイントにアクセスしてデータを保存および取得します。Amazon Simple Storage Service (Amazon S3) と Amazon DynamoDB は、マネージドサービスの例です。これら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、制御するためのソフトウェアシステムで、原材料を工場の完成製品に変換します。

MAP

「移行促進プログラム」を参照してください。

メカニズム

ツールを作成し、ツールの導入を推進し、調整を行うために結果を検査する完全なプロセス。メカニズムは、動作中にそれ自体を強化して改善するサイクルです。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。

メッセージキューイングテレメトリトランスポート (MQTT)

リソースに制約のある IoT デバイス用の、パブリッシュ/サブスクライブパターンに基づく軽量 machine-to-machine (M2M) 通信プロトコル。

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)

に移行するためのビジネスケースを検証するための情報を提供するオンラインツール AWS クラウド。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 に移行するために使用するアプローチ AWS クラウド。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、「」の「アプリケーションをモダナイズするための戦略 AWS クラウド」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、『』の「アプリケーションのモダナイゼーション準備状況の評価 AWS クラウド」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

「移行ポートフォリオ評価」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。

ミュータブルインフラストラクチャ

本番ワークロードの既存のインフラストラクチャを更新および変更するモデル。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

「オリジンアクセスコントロール」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

「 オペレーションの統合」を参照してください。

OLA

「運用レベルの契約」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。

オープンプロセス通信 - 統合アーキテクチャ (OPC-UA)

産業用オートメーション用のmachine-to-machine (M2M) 通信プロトコル。OPC-UA は、データの暗号化、認証、認可スキームを備えた相互運用性標準を提供します。

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)

インシデントや潜在的な障害の理解、評価、防止、または範囲の縮小に役立つ質問とそれに関連するベストプラクティスのチェックリスト。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するハードウェアおよびソフトウェアシステム。製造では、OT と情報技術 (IT) システムの統合が、Industry 4.0 変換の主な焦点です。

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のため、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。

オリジンアクセスコントロール (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コントロールが可能です。

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

「プログラム可能な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

アクセス許可を定義 (アイデンティティベースのポリシーを参照)、アクセス条件を指定 (リソースベースのポリシーを参照)、または の組織内のすべてのアカウントに対する最大アクセス許可を定義 AWS Organizations （ サービスコントロールポリシーを参照) できるオブジェクト。

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。詳細については、マイクロサービスでのデータ永続性の有効化 を参照してください。

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、「移行準備状況ガイド」を参照してください。

述語

`true` または を返すクエリ条件。一般的に`false`は `WHERE`句にあります。

述語プッシュダウン

転送前にクエリ内のデータをフィルタリングするデータベースクエリ最適化手法。これにより、リレーショナルデータベースから取得して処理する必要があるデータの量が減少し、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、Implementing security controls on AWSのPreventative controls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内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じてプライバシーを考慮するシステムエンジニアリングアプローチ。

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を防ぐ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、プロビジョニング前にリソースをスキャンします。リソースがコントロールに準拠していない場合、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)

設計、開発、発売から成長と成熟まで、製品のデータとプロセスのライフサイクル全体にわたる管理。

本番環境

「環境」を参照してください。

プログラム可能なロジックコントローラー (PLC)

製造では、マシンをモニタリングし、製造プロセスを自動化する、信頼性の高い適応可能なコンピュータです。

プロンプトの連鎖

1 つの LLM プロンプトの出力を次のプロンプトの入力として使用して、より良いレスポンスを生成します。この手法は、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善または拡張したりするために使用されます。これにより、モデルのレスポンスの精度と関連性が向上し、より詳細でパーソナライズされた結果が得られます。

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。

パブリッシュ/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。スケーラビリティと応答性を向上させます。たとえば、マイクロサービスベースの MES では、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できるチャネルにイベントメッセージを発行できます。システムは、公開サービスを変更せずに新しいマイクロサービスを追加できます。

Q

クエリプラン

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る手順などの一連のステップ。

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

責任、説明責任、相談、通知 (RACI) を参照してください。

RAG

「取得拡張生成」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

責任、説明責任、相談、情報 (RACI) を参照してください。

RCAC

「行と列のアクセスコントロール」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。

再設計

「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、AWS リージョン「アカウントで使用できる を指定する」を参照してください。

回帰

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。

プラットフォーム変更

「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

中断に抵抗または回復するアプリケーションの機能。高可用性とディザスタリカバリは、 で回復性を計画する際の一般的な考慮事項です AWS クラウド。詳細については、AWS クラウド「レジリエンス」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。サポートを含めると、そのマトリックスは RASCI マトリックスと呼ばれ、サポートを除外すると 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、Implementing security controls on AWSのResponsive controls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。

取得拡張生成 (RAG)

LLM がレスポンスを生成する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を参照する生成 AI テクノロジー。たとえば、RAG モデルは、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する場合があります。詳細については、「RAG とは」を参照してください。

ローテーション

攻撃者が認証情報にアクセスすることをより困難にするために、シークレットを定期的に更新するプロセス。

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、 AWS Management Console にログインしたり AWS 、 API オペレーションを呼び出すことができます。SAML 2.0 ベースのフェデレーションの詳細については、IAM ドキュメントのSAML 2.0 ベースのフェデレーションについてを参照してください。

SCADA

「監視コントロールとデータ取得」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。シークレット値は、バイナリ、1 つの文字列、または複数の文字列にすることができます。詳細については、Secrets Manager ドキュメントの「Secrets Manager シークレットの内容」を参照してください。

設計によるセキュリティ

開発プロセス全体でセキュリティを考慮するシステムエンジニアリングアプローチ。

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、予防的、検出的、応答的、プロアクティブの 4 つの主なタイプがあります。

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に自動的に応答または修復するように設計された、事前定義された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。自動応答アクションの例としては、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

送信先にあるデータの、それ AWS のサービス を受け取る による暗号化。

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、スループットなど、サービスのパフォーマンス側面の測定。

サービスレベルの目標 (SLO)

サービスレベルのインジケータによって測定される、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。

SIEM

セキュリティ情報とイベント管理システムを参照してください。

単一障害点 (SPOF)

システムを中断する可能性のあるアプリケーションの 1 つの重要なコンポーネントの障害。

SLA

「サービスレベルの契約」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、『』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 AWS クラウド」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

1 つの大きなファクトテーブルを使用してトランザクションデータまたは測定データを保存し、1 つ以上の小さなディメンションテーブルを使用してデータ属性を保存するデータベース組織構造。この構造は、データウェアハウスまたはビジネスインテリジェンスの目的で使用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。

監視制御とデータ収集 (SCADA)

製造では、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稼働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潜在的な問題を検出したり、パフォーマンスをモニタリングしたりする方法でシステムをテストします。Amazon CloudWatch Synthetics を使用して、これらのテストを作成できます。

システムプロンプト

LLM にコンテキスト、指示、またはガイドラインを提供して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、ユーザーとのやり取りのルールを確立するのに役立ちます。

T

tags

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。

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 枚のピザで養う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。詳細については、深層学習システムにおける不確実性の定量化 ガイドを参照してください。

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する行のグループに対して計算を実行する SQL 関数。ウィンドウ関数は、移動平均の計算や、現在の行の相対位置に基づく行の値へのアクセスなどの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 「ワークロード認定フレームワーク」を参照してください。

Write Once, Read Many (WORM)

データを 1 回書き込み、データの削除や変更を防ぐストレージモデル。承認されたユーザーは、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

ゼロデイ脆弱性を利用する攻撃、通常はマルウェア。

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 にタスクを実行する手順を提供しますが、タスクのガイドに役立つ例 (ショット) はありません。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。