AWS Well-Architected フレームワークを Amazon WorkSpaces アプリケーションに適用する

AWS 規範ガイダンス

AWS 規範ガイダンス: AWS Well-Architected フレームワークを Amazon WorkSpaces アプリケーションに適用する

Table of Contents

AWS Well-Architected フレームワークを Amazon WorkSpaces アプリケーションに適用する

モハメド・ウォリ、アマゾン ウェブ サービス

2025 年 7 月 (ドキュメント履歴)

このガイドでは、Amazon WorkSpaces アプリケーションを使用する際に AWS Well-Architected フレームワークを適用するためのベストプラクティスについて説明します。WorkSpaces Applications は、デスクトップアプリケーションを書き換えることなくユーザーにストリーミングできるフルマネージド型のアプリケーションストリーミングサービスです。

AWS Well-Architected フレームワークは、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、回復力があり、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。また、ユーザーと AWS パートナーがアーキテクチャを評価し、スケーラブルな設計を実装するための一貫したアプローチも提供します。

AWS Well-Architected フレームワークは、次の 6 つの柱を中心に構築されています。

- オペレーショナルエクセレンス
- 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

このガイドでは、これらの柱とベストプラクティスが WorkSpaces アプリケーションの使用にどのように適用されるかについて説明します。

対象者

このガイドは以下を対象としています。

- WorkSpaces アプリケーションソリューションを設計および実装し、アーキテクチャが AWS Well-Architected Framework の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るクラウドアーキテクトとエンジニア。

- WorkSpaces アプリケーション環境を管理および保守し、フリートの管理、スケーリング、モニタリングを処理し、コストとパフォーマンスを最適化する必要がある IT 運用チーム。
- WorkSpaces アプリケーションを検討中または既に使用しており、デスクトップアプリケーションをユーザーにストリーミングしたいと考えている組織や企業は、安全で高性能、回復力があり、効率的なインフラストラクチャを構築する必要があります。

目的

このガイドのベストプラクティスに従うと、次のことに役立ちます。

- でデスクトップアプリケーションをストリーミングするための、安全で高性能、回復力があり効率的なインフラストラクチャを構築します AWS クラウド。
- WorkSpaces アプリケーションアーキテクチャを評価し、スケーラブルな設計を実装するときは、一貫したアプローチを適用します。

運用上の優秀性の柱

運用上の優秀性 (OE) は、ユーザーの期待に一貫して応え、それを超える高品質のソフトウェアソリューションを作成することに専念しています。 AWS Well-Architected フレームワークの運用上の優秀性の柱には、効果的なチーム組織、堅牢なワークロード設計、効率的な大規模な運用、時間の経過とともに変化する要件へのシームレスな適応のための実証済みの戦略が含まれています。これらの原則に従うことで、組織はシステムの耐障害性、パフォーマンス、進化するビジネスニーズとの整合性を維持できます。

この柱を WorkSpaces アプリケーションストリーミング環境に適用するための主な重点領域:

- モニタリングとオブザーバビリティ
- 自動化と DevOps
- 運用手順とドキュメント
- サポートとインシデント管理

ビジネス成果に関するチームを編成する

ビジネス目標と主要業績評価指標 (KPIs) が最適化された人材、プロセス、テクノロジーを通じて組織の変革を促進する、強力なリーダーシップコミットメントを持つクラウドに沿った運用モデルを作成します。

- チーム構造。アプリケーションストリーミングの結果に沿った専用チームを確立します。例:
  - イメージ管理チームは、アプリケーションのパッケージングとイメージの最適化を担当します。
  - フリート運用チームは、容量、パフォーマンス、スケーリングを管理します。
  - ユーザーエクスペリエンスチームは、エンドユーザーのサポートと満足度を処理します。
- KPIsとメトリクス。次のようなビジネスに沿ったメトリクスを定義して追跡します。
  - アプリケーションの可用性レート
  - 新しいアプリケーションをデプロイする時間
  - アプリケーションストリーミング時間あたりのコスト
- 運用モデル。以下の明確なプロセスを作成します。
  - アプリケーションのオンボーディングと更新
  - フリート容量管理
  - ユーザーアクセスのプロビジョニング

- インシデント対応と解決

実用的なインサイトのためのオブザーバビリティを実装する

包括的なモニタリングとオブザーバビリティを実装して、KPIs。この原則により、データ駆動型の意思決定が可能になり、パフォーマンス、信頼性、コストにわたってプロアクティブな改善が可能になります。

- パフォーマンスモニタリングを実装します。Amazon CloudWatch を次のように設定します。
  - 需要を満たすのに十分な容量を確保します。たとえば、次のメトリクスを使用できます。
    - `AvailableCapacity` 利用可能なストリーミングインスタンスをモニタリングするには
    - `InUseCapacity` 現在使用されているインスタンスを追跡するには
    - `CapacityUtilization` フリート使用率をモニタリングする
  - ユーザーエクスペリエンスとパフォーマンスをモニタリングします。
  - サービスの問題を迅速に特定して対処します。
- WorkSpaces アプリケーション使用状況レポートを追跡および分析します。
- アプリケーションログをキャプチャして分析します。詳細については、 AWS ブログ記事「Using Kinesis Agent for Linux to stream application logs in WorkSpaces Applications」および「Using Kinesis Agent for Microsoft Windows to store WorkSpaces Applications Windows event logs」を参照してください。
- チャット通知を使用して WorkSpaces アプリケーションのメトリクスとイベントをモニタリングします。詳細については、 AWS ブログ記事「Chatbot による AWSAWS エンドユーザーコンピューティング (EUC) のモニタリングと自動化」を参照してください。
- ビジュアルキューを使用してプロアクティブセッション管理を有効にします。詳細については、AWS ブログ記事Amazon WorkSpaces アプリケーションでセッションの有効期限とカウントダウンタイマーを表示する」を参照してください。
- 使用パターンと傾向の視覚化を作成します。詳細については、 AWS ブログ記事「Amazon OpenSearch Service での Amazon WorkSpaces アプリケーション使用状況レポートの取り込みと視覚化」を参照してください。
- EUC ツールキットを使用して、アクティブなセッションのモニタリング、フリートインベントリの追跡、セッションレポートの生成 (CSV エクスポート) を行います。詳細については、AWS ブログ記事「EUC Toolkit を使用して Amazon WorkSpaces アプリケーションと Amazon WorkSpaces を管理する」を参照してください。

可能な限り安全に自動化する

Infrastructure as Code (IaC) の原則を適用して、ワークロードオペレーションのあらゆる側面を自動化します。ガードレールを使用すると、手動による介入を減らしながら、安全で一貫した実行を確保できます。

- Image Assistant CLI を使用して、WorkSpaces アプリケーションイメージの作成と設定を自動化します。詳細については、WorkSpaces Applications ドキュメントの「Image Assistant CLI オペレーションを使用してプログラムで Amazon WorkSpaces アプリケーションイメージを作成する」を参照してください。 WorkSpaces
  - アプリケーションのインストール: Image Assistant CLI を使用して、イメージの作成中にアプリケーションのインストールを自動化します。
  - イメージの作成: Image Assistant CLI コマンドを使用して、プログラムで WorkSpaces アプリケーションイメージを作成します。
  - 設定管理: デフォルトのアプリケーション設定と起動パラメータの設定を自動化します。
- WorkSpaces Applications イメージのカスタマイズを自動化します。詳細については、 AWS ブログ記事「カスタマイズされた WorkSpaces Applications Windows イメージを自動的に作成する」を参照してください。
- IaC を適用して、WorkSpaces アプリケーションのインフラストラクチャとアプリケーションコンポーネントをデプロイします。詳細については、 AWS ブログ記事「Automation of infrastructure and application deployment for Amazon WorkSpaces Applications with Terraform」を参照してください。
- 以下を含むフリート管理の自動プロセスを実装します。
  - 需要に基づくフリートスケーリング。自動スケーリングポリシーを設定して、使用率メトリクスに基づいてフリート容量を自動的に調整します。詳細については、 AWS ブログ記事「Use AWS Lambda to adjust scaling steps and thresholds for Amazon WorkSpaces Applications」を参照してください。
  - ベースイメージの更新。が提供する WorkSpaces Applications ベースイメージの自動更新を利用できます AWS。
  - 容量の最適化。自動スケーリングしきい値を設定して、需要パターンに基づいてリソース使用量を最適化します。
- 安全制御を自動化するようにガードレールを設定します。
  - 最大フリートサイズ制限。過剰プロビジョニングを防ぐために、フリート容量の上限を設定します。

  - スケーリングポリシーの設定。適切なしきい値を使用して、ステップスケーリングまたはターゲット追跡スケーリングポリシーを実装します。
  - サービスクォータ。 AWS 過剰なリソース割り当てを防ぐために、サービスクォータを組み込みの制限として使用します。
  - スケールイン保護。スケールイン保護を設定して、スケーリングイベント中にアクティブなインスタンスが削除されないようにします。
- Image Builder、フリート、統合テストなど、テストと検証を実行します。
  - Image Builder のテスト:
    - Image Builder インターフェイスで直接アプリケーションをテストします。
    - アプリケーションの起動と機能を確認します。
    - ユーザー設定と設定をテストします。
    - アプリケーションの互換性を検証します。
  - フリートテスト:
    - さまざまなクライアントデバイスからストリーミングセッションをテストします。
    - ユーザーの使用権限とアクセスを確認します。
    - アプリケーションのパフォーマンスを検証します。
    - クリップボード、ファイル転送、印刷などの要素とオペレーションのユーザーエクスペリエンスをテストします。
  - 統合テスト:
    - Active Directory または SAML 2.0 ベースの認証をテストします。
    - ホームフォルダと永続的ストレージをテストします。
    - アプリケーションのエンタイトルメントをテストします。
    - USB デバイスリダイレクトをテストします (設定されている場合) 。
- WorkSpaces アプリケーションマネージャーを使用して、アプリケーションのパッケージ化とデプロイを自動化します。詳細については、 AWS ブログ記事Amazon WorkSpaces Applications のアプリケーションマネージャーによるアプリケーションのオンボーディングを効率化する」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) パイプラインを使用して、新しいアプリケーションバージョンのデプロイを自動化します。詳細については、 AWS ブログ記事「Eagle: Optimize CI/CD and end user experience in Amazon WorkSpaces Applications」を参照してください。

頻繁で小さく、可逆的な変更を行う

最小限のリスクと簡単なロールバック機能で、頻繁で小規模な自動デプロイを可能にする、疎結合でスケーラブルなワークロードを構築します。

- イメージの更新には、バージョニングされたイメージの作成と増分更新を使用します。
  - バージョニングされたイメージの作成:
    - Image Builder を使用して、変更のセットごとに新しいイメージを作成します。
    - ロールバックシナリオをサポートするために、複数のイメージバージョンを維持します。
    - AWS タグ付け戦略を使用して、イメージのバージョンと属性を追跡します。
  - 増分更新:
    - アプリケーションまたは設定に小さな増分変更を加えます。
    - 新しいイメージを作成する前に、Image Builder で更新を徹底的にテストします。
    - 各新しいイメージバージョンで行ったすべての変更を文書化します。
- コントロールフリートの更新の場合:
  - テスト用に更新されたイメージを使用して新しいフリートを作成します。
  - アクティブなセッションを中断することなく、既存のフリート属性を変更します。
- ドキュメント、テストプロトコル、承認ワークフロー、モニタリングプロセスに関する変更管理手順を確立します。
  - ドキュメント:
    - すべてのイメージとフリートの更新の詳細な変更ログを維持します。
    - 各変更のテスト手順と結果を文書化します。
    - AWS CloudTrail を使用して、設定の変更を追跡および監査します。
  - プロトコルのテスト:
    - すべての変更に対して包括的なテストプロセスを確立します。
    - アプリケーションの機能、パフォーマンス、ユーザーエクスペリエンステストを含めます。
    - 新しいイメージを作成する前に、Image Builder でテストを実行します。
    - フルデプロイの前に、非本番稼働フリートで追加のテストを実行します。
  - 承認ワークフロー:
    - 本番環境への変更の承認プロセスを実装します。
    - 承認と標準の更新を必要とする変更の基準を定義します。
    - 変更承認の役割と責任を確立します。

- モニタリングと検証:
  - Amazon CloudWatch を使用して、変更後のフリートとアプリケーションのパフォーマンスをモニタリングします。
  - 主要メトリクスのアラートを設定して、更新後に問題をすばやく特定します。
  - 実装後のレビューを実施して、変更の成功を検証し、学習情報を収集します。

オペレーション手順を頻繁に絞り込む

定期的なレビュー、更新、チームエンゲージメントを通じて運用手順を継続的に改善し、すべてのステークホルダーに最新情報を提供し、ベストプラクティスに合致させます。

- ドキュメント管理。WorkSpaces Applications の手順に関するバージョン管理された最新のドキュメントを一元管理し、運用上の一貫性とチーム間の知識共有を確保します。
  - 必要なドキュメント: イメージの作成と管理、フリートオペレーション、トラブルシューティングのための重要な WorkSpaces アプリケーションオペレーションに関するup-to-dateドキュメントを維持します。
  - 運用レビュー: パフォーマンスメトリクスやインシデント管理など、主要な運用面を監視およびレビューします。
- 継続的な改善。 AWS のサービス 更新、運用メトリクス、学習したベストプラクティスを標準手順に組み込むことで、WorkSpaces アプリケーションの運用を体系的に強化します。
  - サービスの更新: WorkSpaces アプリケーションのリリースノートをモニタリングして、新機能、サービスの改善、セキュリティの更新、リージョンの可用性を確認します。
  - ベストプラクティス: AWS Well-Architected Framework の更新、WorkSpaces アプリケーションのベストプラクティス、 AWS リファレンスアーキテクチャ、および AWS セキュリティの推奨事項を確認して組み込みます。
  - ナレッジ管理: 標準運用手順、ランブック、トラブルシューティングガイド、およびユーザーサポートドキュメントを維持および更新します。

失敗を予測する

障害シナリオテストを定期的に実施して、リスクを理解し、対応手順を検証し、実際のインシデントを処理するためのチームの準備状況を向上させます。

- 失敗テスト。フリート容量の枯渇、アプリケーションの起動障害、ネットワーク接続の問題などの障害を定期的にシミュレートしてテストします。
  - フリート容量の枯渇:
    - 容量制限に近づいたときのフリートスケーリング動作をモニタリングしてテストします。
    - `CapacityUtilization` および `AvailableCapacity`メトリクスの CloudWatch アラームを設定します。
    - ピーク時の使用時に容量の制約を処理する手順を実装します。
  - アプリケーション起動の失敗:
    - ストリーミングインスタンスでアプリケーションの起動動作をテストします。
    - さまざまなフリート設定でアプリケーションアクセスとパフォーマンスを検証します。
  - ネットワーク接続の問題:
    - さまざまなネットワーク条件でストリーミングセッションのパフォーマンスをテストします。
    - 接続品質の問題`StreamingSessionLatency`を監視します。
    - VPC 設定とセキュリティグループが適切に設定されていることを確認します。
- 復旧手順。以下の手順を開発してテストします。
  - 間のフリートフェイルオーバー AWS アベイラビリティーゾーン。さらに、フリート容量のスケーリング、フリートの更新の管理、インスタンスのヘルス問題への対応の手順を文書化します。
  - ユーザーデータ管理:
    - Windows フリートの場合は Amazon Simple Storage Service (Amazon S3) のホームフォルダのアプリケーション設定の永続性とストレージソリューションを設定し、Linux フリートの場合は Amazon Elastic File System (Amazon EFS) の共有ファイルシステムを設定します。
    - セッション間のデータ同期を検証します。
  - サービスの継続性。新しいフリートインスタンスの作成、イメージの更新の管理、セッションの切断の処理の手順を維持します。
- リスク管理。以下を特定して軽減します。
  - 適切なフリート最小容量の設定、需要パターンに基づく自動スケーリングポリシーの設定、`CapacityUtilization`、、 などの CloudWatch メトリクスを使用したフリート使用率の傾向のモニタリングによる容量の制約`InUse`、`Capacity`、`Available`、`Capacity`。
  - などの主要なメトリクスを追跡`StreamingSessionLatency`し、適切な CloudWatch アラームを設定することで、パフォーマンスのボトルネックが発生します。

すべての運用イベントとメトリクスから学ぶ

運用上のイベントや失敗から学んだ教訓を組織全体で共有することで、継続的な改善の文化を育みます。ビジネス成果への影響を強調します。

- イベント分析。サービスの中断、パフォーマンスの低下、ユーザーの苦情、容量の問題を文書化して分析します。
- メトリクスのレビュー。使用状況パターン、パフォーマンス傾向、コストメトリクス、ユーザー満足度データを定期的に分析します。
- ナレッジ共有。チーム学習セッション、ベストプラクティスドキュメント、チーム間の知識移転、インシデントの遡及のためのプロセスを確立します。

マネージドサービスを使用する

AWS マネージドサービスを使用し、その周囲に標準化された手順を構築することで、運用上のオーバーヘッドを最小限に抑えます。を次の AWS マネージドサービスと統合します。

- AWS Systems Manager オートメーション用
- モニタリング用の Amazon CloudWatch
- アクセスコントロールの AWS Identity and Access Management (IAM)
- Windows フリートのユーザーストレージ用 Amazon S3
- Linux フリートのユーザーストレージ用の Amazon EFS
- AWS Directory Service ユーザー認証用

セキュリティの柱

AWS Well-Architected フレームワークのセキュリティの柱は、クラウド機能を活用して、情報、インフラストラクチャ、リソースの堅牢な保護メカニズムを確立することに焦点を当てています。これらの原則は、イノベーションを可能にしながら、全体的なセキュリティ体制を強化するのに役立ちます。

この柱を WorkSpaces Applications ストリーミング環境に適用するための主な重点領域:

- データの整合性と機密性
- ユーザーアクセス許可の管理
- セキュリティイベントを検出するためのコントロールの確立

強力な ID 基盤を実装する

ID 管理を一元化し、長期的な認証情報を回避しながら AWS 、リソースにアクセスするために必要な最小限のアクセス許可を使用します。

- WorkSpaces アプリケーションリソースに最小特権のアクセス許可を付与します。
  - 最小限のアクセス許可で WorkSpaces アプリケーションフリートの特定の IAM ロールを作成します。
  - Image Builder の制限された IAM アクセス許可を設定します。
  - WorkSpaces アプリケーション管理機能への管理アクセスを制限します。
  - スタックとフリート管理の詳細なアクセス許可を定義します。
- 適切なユーザー認証メカニズムを実装します。
  - エンタープライズ ID プロバイダーの統合用に SAML 2.0 フェデレーションを設定します。
  - ユーザー管理AWS IAM Identity Center用に をセットアップします。
  - 特定の認証シナリオで必要な場合のみ、カスタム ID ブローカーを使用します。
  - サポートされている場合は、多要素認証 (MFA) を実装します。
- アプリケーションへのユーザーアクセスを制御します。
  - アプリケーションの使用権限を設定して、特定のアプリケーションへのアクセスを制限します。
  - ユーザーロールに基づいてアプリケーション割り当てグループを作成します。
  - スタックのアクセス許可を使用してアプリケーションアクセスを管理します。

    - アプリケーションの動作を制御するセッションポリシーを実装します。
- 適切なコントロールを使用してユーザーセッションを保護します。
    - セッションタイムアウトポリシーを設定します。
    - 切断タイムアウトアクションを設定します。
    - セッション永続性要件を実装します。
    - ファイルシステムのリダイレクト許可を制御します。
- WorkSpaces アプリケーションの証明書ベースの認証を設定します。詳細については、 AWS ブログ記事「Simplify certificate-based authentication for WorkSpaces Applications and WorkSpaces with AWS Private CA Connector for Active Directory」を参照してください。
- セッションタグを使用して、きめ細かなアクセスコントロールを実装します。詳細については、AWS ブログ記事「セッションタグを使用して WorkSpaces アプリケーションのアクセス許可を簡素化する」を参照してください。

トレーサビリティを維持する

すべての環境の変更とアクティビティに対して、リアルタイムモニタリングと自動応答システムを実装します。

- アプリケーションログの CloudWatch ログ記録を設定して、アプリケーションの起動、クラッシュ、エラーなど、アプリケーション固有のイベントをモニタリングします。セッションの開始、停止、ユーザー接続イベントなど、ストリーミングセッション情報を追跡するようにセッションログを設定します。
- CloudTrail をアクティブ化してすべての WorkSpaces Applications API コールをログに記録し、フリートの作成と変更、Image Builder オペレーション、スタック設定、ユーザー管理アクティビティなどの管理イベントを追跡します。
- WorkSpaces Applications インスタンスのアクティビティをモニタリングします。
    - システムレベルのイベントをキャプチャするようにインスタンスのログ記録を設定します。
    - アプリケーションの起動と失敗を追跡します。
    - システムリソースの使用状況とパフォーマンスをモニタリングします。
- ユーザーアクティビティを追跡します。
    - ユーザー認証の試行と失敗をモニタリングします。CloudWatch メトリクスと CloudWatch Logs を使用して、ユーザーのログイン試行、セッションの開始時刻と終了時刻、セッション切断イベントを追跡します。

  - アプリケーションの使用パターンを追跡します。WorkSpaces アプリケーション使用状況レポートを有効にして、セッション期間、開始時刻と終了時刻、使用されるインスタンスタイプ、アクセスされたアプリケーションなどの情報を取得します。
  - 有効なホームフォルダを使用してファイルシステムのアクティビティを記録します。
  - データ損失防止の目標を達成するために、クリップボード設定と印刷オペレーションを設定します。
- ユーザー認証の失敗、異常なセッションパターン、リソースアクセス違反などのセキュリティ関連のメトリクスに対して CloudWatch アラームを設定します。
- EUC ツールキットを使用して、アクティブなセッションと状態を追跡し、使用中のアクティブなセッションの IP アドレスをモニタリングし、監査のためにセッションデータをエクスポートします。詳細については、 AWS ブログ記事「Using the EUC toolkit to manage Amazon WorkSpaces Applications and Amazon WorkSpaces」を参照してください。

すべてのレイヤーにセキュリティを適用する

ネットワークエッジからアプリケーションコードまで、インフラストラクチャのすべてのコンポーネントに複数のセキュリティコントロールレイヤーを実装します。

- ネットワークレイヤーのセキュリティを設定します。
  - 厳格なセキュリティグループルールを実装します。
  - WorkSpaces Applications フリートインスタンスを、直接インターネットにアクセスできないプライベートサブネットに配置します。NAT デバイスによるインターネットアクセスを制御します。
  - 仮想プライベートクラウド (VPC) エンドポイントを使用して、サポートされている AWS のサービス (Amazon S3 など) にアクセスします。
  - 追加のネットワークセキュリティレイヤーとしてネットワークアクセスコントロールリスト (ACLs) を実装します。
  - ストリーミングポート (HTTPS および WebSocket Secure の場合は TCP 8443) アクセスを特定の IP 範囲に制限します。
- アクセスレイヤーのセキュリティを設定します。
  - セッションタイムアウトポリシーを実装して、非アクティブなユーザーを自動的に切断します。
  - セッションタグを使用して属性ベースのアクセスコントロールを実装します。詳細については、AWS ブログ記事「セッションタグを使用して WorkSpaces アプリケーションのアクセス許可を簡素化する」を参照してください。

- アプリケーションレイヤーのセキュリティを設定します。
  - アプリケーションの使用権限を設定して、特定のアプリケーションにアクセスできるユーザーを制御します。
  - ファイルシステムのリダイレクトコントロールを有効にして、ローカルドライブへのアクセスを制限します。
  - セキュリティ要件に基づいて、クリップボード、ファイル転送、印刷のアクセス許可を設定します。
  - セキュリティポリシーに従って USB デバイスアクセスコントロールを設定します。
- イメージレイヤーのセキュリティを設定します。
  - セキュリティ要件を満たす強化ベースイメージを作成して維持します。
  - ベースイメージを最新のセキュリティパッチで更新します。
  - ベースイメージで Windows セキュリティ設定を構成します。
  - ベースイメージの不要な Windows サービスと機能を無効にします。

セキュリティのベストプラクティスを自動化する

バージョン管理されたテンプレートでコード定義の自動セキュリティコントロールを使用して、安全でスケーラブルなインフラストラクチャのデプロイを可能にします。

- などのサービスを使用してInfrastructure as Code (IaC) を使用し AWS CloudFormation 、すべてのフリートデプロイに一貫したセキュリティ設定を実装します。詳細については、 AWS ブログ記事「Amazon Amazon WorkSpaces アプリケーションと Amazon WorkSpaces に追加のセキュリティグループを自動的にアタッチする」を参照してください。
- Image Assistant CLI を使用して、イメージ作成のセキュリティプロセスを自動化します。
- Amazon CloudWatch アラーム、Amazon EventBridge ルール、自動応答用の AWS Lambda 関数を使用して、容量使用率のしきい値を超えた自動応答、不正アクセスの試行、セキュリティグループの変更を設定します。

データから遠ざける

データ処理プロセスを自動化して、直接の人間によるアクセスを最小限に抑え、エラーや誤処理のリスクを軽減します。

- アプリケーションの使用権限を設定して、特定のアプリケーションにアクセスできるユーザーを制御します。
- 動的アプリケーションフレームワークを使用して動的アプリケーションプロバイダーを構築し、ユーザー属性に基づいてアプリケーションを動的に利用できるようにします。
- ファイルシステムのリダイレクトを設定して、ユーザーがアクセスできるローカルドライブを制御し、特定のフォルダへのアクセスを制限し、ローカルセッションとストリーミングセッション間のファイル転送アクセス許可を管理します。
- クリップボードの制限を実装して、ローカルセッションとストリーミングセッション間のクリップボード共有を無効にし、必要に応じて一方向クリップボードフローを有効にして、不正なデータコピーを防止します。
- アプリケーション設定の永続性を設定して、アプリケーション設定を自動的に保存および復元し、手動設定の必要性を排除し、一貫したユーザーエクスペリエンスを維持します。

セキュリティイベントに備える

自動ツールを使用してインシデント対応計画を策定して実践し、セキュリティイベントを迅速に検出、調査、復旧できるようにします。

- 認証試行の失敗、フリートセキュリティグループの変更、イメージ設定の変更、異常なストリーミングセッションパターンに対して CloudWatch アラームを設定します。
- 次のような一般的な WorkSpaces アプリケーションセキュリティシナリオの応答手順を文書化します。
  - 許可されていないアクセスの試行
    - 検出: 認証の失敗をモニタリングします。
    - レスポンス: ユーザー権限の取り消し、セッションログの確認、アクセスポリシーの更新を行います。
  - 侵害されたストリーミングインスタンス
    - 検出: インスタンスの動作をモニタリングします。
    - レスポンス: 影響を受けるセッションを終了し、フリートインスタンスを置き換え、セキュリティグループ設定を確認します。
  - データ流出の試行
    - 検出: ファイル転送アクティビティをモニタリングします。
    - レスポンス: クリップボードとファイル転送ログの確認、ファイル転送アクセス許可の調整、データ保護ポリシーの更新を行います。

- フリートインスタンスの交換、セキュリティグループの復元、ユーザーアクセスの再設定、アプリケーション設定の復旧のための自動復旧プロセスを実装します。
- セキュリティの検出 AWS のサービス 結果には 、脅威の検出 AWS Security Hub CSPM には Amazon GuardDuty などのセキュリティ管理に使用します。

信頼性の柱

AWS Well-Architected フレームワークの信頼性の柱は、システムが意図した機能とパフォーマンスレベルを、その存続期間を通じて期待される運用期間中にどの程度維持しているかに対処します。ワークロードライフサイクルのすべての段階でテストと検証の戦略など AWS、 で信頼性の高いシステムを構築および維持するための包括的なガイドラインを提供します。

この柱を WorkSpaces アプリケーションストリーミング環境に適用するための主な重点領域:

- フリートの管理とスケーリング
- セッションの信頼性
- アプリケーションの可用性
- 復旧手順

障害から自動的に復旧する

ビジネス価値KPIs をモニタリングして、障害がオペレーションに影響を与える前に障害を予測、防止、または復旧できる自動応答をトリガーします。

- IP サブネットを割り当てる際には、拡張性と可用性を考慮するようにします。
- 重要な CloudWatch メトリクスをモニタリングして、 や などのフリート容量メトリクス`AvailableCapacity`、 などのストリーミング品質メトリクスなど`InUseCapacity`、サービスの可用性とパフォーマンスを確保します`StreamingSessionLatency`。
- キャパシティのしきい値、セッションヘルスメトリクス、パフォーマンスの低下、フリートのヘルスステータスの変更に関するアラートを設定します。
- 組み込み WorkSpaces アプリケーションの自動スケーリング機能を使用して、次のことを行います。
  - 最小および最大フリート容量を設定します。
  - 容量使用率に基づいてスケーリングポリシーを設定します。
  - 技術的なメトリクスだけでなく、ユーザーエクスペリエンスメトリクスとビジネス要件に基づいてスケールアウトとスケールインのしきい値を定義します。
- WorkSpaces アプリケーション環境のディザスタリカバリ環境を構築します。詳細については、AWS ブログ記事Amazon WorkSpaces アプリケーションのディザスタリカバリに関する考慮事項」を参照してください。

復旧手順をテストする

クラウド環境により、障害シナリオと復旧手順の自動テストが可能になります。これらの機能は、実際の障害が発生する前に脆弱性を特定して修正するのに役立ちます。

- フリート復旧テスト。複数のシナリオにわたって包括的なフリート復旧テストを実装します。
  - インスタンスの終了をシミュレートして、自動スケーリングレスポンスを検証します。
  - フリートの最小容量のメンテナンスを検証します。
  - インスタンスの置き換えタイミングとユーザーリダイレクトをテストします。
  - スケーリングポリシーの有効性を検証します。
  - フリートの容量制限とオーバーフロー処理をテストします。
- セッション復旧テスト。セッション復旧の検証手順を実装します。
  - 切断と再接続のシナリオをテストします。
  - アプリケーションの状態の保存を確認します。
  - さまざまなネットワーク中断シナリオをテストします。
  - セッションタイムアウトの動作を検証します。
  - ユーザー認証の永続性を検証します。
  - 一時的なストレージ処理を確認します。

ワークロードの全体的な可用性を高めるために水平方向にスケールする

ワークロードを複数の小規模なリソースに分散して、個々の障害の影響を最小限に抑え、単一障害点を排除します。

- 複数のアベイラビリティーゾーンにフリートインスタンスをデプロイします。
- 適切な最小フリート容量を設定します。
- フリートの自動スケーリングを設定し、適切なスケーリングしきい値を設定します。
- フリート全体のキャパシティ使用率をモニタリングします。
- WorkSpaces アプリケーションスタックを複数のリージョンにデプロイします。詳細については、AWS ブログ記事Amazon WorkSpaces アプリケーションのレイテンシーベースのルーティングによるユーザーエクスペリエンスの最適化」を参照してください。

容量の推測を停止する

クラウドの自動スケーリング機能を使用して、需要に基づいてリソースを動的に調整します。これにより、最適な効率を維持しながら、リソースの飽和を防ぐことができます。

- CapacityUtilization、、 などの主要なメトリクスをモニタリングInUseCapacityしてAvailableCapacity、容量のニーズを把握します。
- さまざまな期間におけるフリート使用率の傾向を追跡します。毎日のパターン、毎週のバリエーション、毎月の傾向、季節的なピークをモニタリングします。
- スケーリングポリシーを設定し、スケーリングしきい値を設定します。
- フェイルオーバーを行える十分な最大使用量に配慮して、現在のクォータが設定されていることを確認します。
- アーキテクチャ全体で、固定のサービスクォータおよび制約に対応します。

自動化による変更の管理

自動化コード自体のバージョン管理された変更を含む、自動化によるインフラストラクチャの変更を実装します。

- フリート設定には IaC を使用します。
- 一貫したスケーリングポリシーを実装します。
- Image Assistant CLI を使用して、一貫したイメージを作成します。

パフォーマンス効率の柱

AWS Well-Architected フレームワークのパフォーマンス効率の柱は、需要の変動や新たなテクノロジーへの適応性を確保しながら、パフォーマンス目標を達成または上回るようにクラウドリソースの使用を最適化することに重点を置いています。動的なクラウド環境でピーク効率を維持するために、システムを継続的に微調整することの重要性を強調しています。

この柱を WorkSpaces アプリケーションストリーミング環境に適用するための主な重点領域:

- インスタンスタイプの選択と最適化
- ストリーミングパフォーマンスの最適化
- フリート容量管理

高度なテクノロジーの民主化

複雑なテクノロジーのクラウドベンダーマネージドサービスを活用して、チームがインフラストラクチャ管理ではなく製品開発に集中できるようにします。

- アプリケーションの要件に基づいて適切なインスタンスタイプを設定します。
  - グラフィックを多用するアプリケーションには GPU 対応インスタンスを選択します。
  - アプリケーションのニーズに基づいて、適切な GPU ファミリー (Graphics G4dn や Graphics G5 など) を選択します。
- 次のいずれかの認証方法を選択して設定します。
  - SAML 2.0 ベースの ID プロバイダーとの統合を設定します。
  - ユーザープールの設定を行います。
  - を と統合しますAWS Directory Service。
- ユーザーのニーズに基づいてストレージオプションを有効にして設定します。
  - Amazon S3 for Windows ベースのフリートにホームフォルダを設定します。
  - Linux ベースのフリート用に Amazon EFS で共有ファイルシステムを設定します。
  - 永続的ストレージのアクセス許可を設定します。
  - アプリケーション設定の永続化を有効にします。

数分でグローバル化

マルチリージョンデプロイを使用すると、レイテンシーが短縮され、グローバルユーザーエクスペリエンスが向上します。

- ユーザーに最も近いリージョンにフリートをデプロイし、リージョンごとに個別のスタックを作成 AWS リージョン することで、複数の にフリートを設定します。
- クロスリージョンリダイレクトを実装して、WorkSpaces アプリケーションユーザーを現在の場所に最も近い AppStream スタックに自動的にリダイレクトします。
- アプリケーション設定の永続性、ホームフォルダ、エラスティックフリートなど、WorkSpaces アプリケーションのオプション機能を使用している場合は、Windows ベースのフリートのユーザーデータの Amazon S3 クロスリージョンレプリケーションと Linux ベースのフリートのクロスリージョンレプリケーションを設定する必要があります。
- リージョン間でイメージをレプリケートします。詳細については、 AWS ドキュメントのAWS リージョン Amazon WorkSpaces アプリケーションの別の に所有しているイメージをコピーする」を参照してください。
- ドメインに参加しているフリートの場合、Active Directory フェデレーションサービス (AD FS) を含む Active Directory インフラストラクチャ (代替として SAML 2.0 と Amazon Cognito を使用している場合を除く) が他のリージョンで適切に設定されていること、およびマルチリージョンレプリケーション機能AWS Directory Service for Microsoft Active Directoryに使用することを確認してください。
- レイテンシーが最も低い WorkSpaces アプリケーションエンドポイントにユーザーを誘導します。詳細については、 AWS ブログ記事Amazon WorkSpaces アプリケーションのレイテンシーベースのルーティングによるユーザーエクスペリエンスの最適化」を参照してください。

サーバーレスアーキテクチャを使用する

サーバーレスアーキテクチャは、コンピューティング機能にクラウドマネージドサービスを使用することで、サーバー管理のオーバーヘッドを排除し、コストを削減します。

次のような AWS サーバーレスサービスを使用します。

- AWS Lambda イベント駆動型関数を使用してタスクを自動化し、カスタムロジックを統合するには
- WorkSpaces アプリケーションのユーザーデータ、アプリケーションファイル、セッションアーティファクトにスケーラブルなストレージを提供する Amazon S3

- WorkSpaces アプリケーションのパフォーマンスと使用状況メトリクスのモニタリング、ログ記録、アラートを提供する Amazon CloudWatch
- WorkSpaces アプリケーションアプリケーションのユーザー認証とアクセスコントロールを容易にする Amazon Cognito
- WorkSpaces アプリケーションと他の サービスまたはカスタムアプリケーション間のインターフェイスとなる RESTful API を作成する Amazon API Gateway APIs WorkSpaces

より頻繁に実験する

クラウドインフラストラクチャを使用すると、さまざまなリソース設定を迅速にテストして、パフォーマンスとコストを最適化できます。

- パフォーマンスとコストを最適化するために、さまざまなインスタンスタイプをテストします。
  - 異なるインスタンスファミリー間でストリームパフォーマンスを比較します。
  - グラフィックアプリケーションの GPU インスタンスと非 GPU インスタンスを評価します。
  - メモリを大量に消費するアプリケーションのメモリ最適化インスタンスをテストします。
- Image Builder を使用してアプリケーション設定をテストします。
  - さまざまなアプリケーション設定でテストイメージを作成します。
  - デプロイ前にアプリケーションのパフォーマンスを検証します。
  - さまざまなインスタンスタイプとのアプリケーションの互換性をテストします。
- 最小容量と最大容量、スケーリングポリシー、最大セッション期間などのセッション設定、切断タイムアウト設定などのフリート容量設定を使用して、フリート設定をテストします。

機械的な交感神経を検討する

ワークロード固有の要件と使用パターンに基づいてクラウドサービスを選択し、最適なパフォーマンスと効率を確保します。

- グラフィックを多用するアプリケーション、DirectX、OpenGL、OpenCL、または 3D 視覚化ソフトウェアを必要とするアプリケーションには、グラフィックス G5 インスタンスを選択します。
- ビジネスアプリケーション、ウェブブラウザ、ライトグラフィックスアプリケーションの`stream.standard`インスタンスを選択する
- などの CloudWatch メトリクスに基づいてストリーミングプロトコルをモニタリングおよび調整します`StreamingSessionLatency`。

- ユーザーに最も近い VPCs で WorkSpaces アプリケーションを設定し、アプリケーションの要件に基づいて適切なネットワーク帯域幅を使用します。
- アプリケーションの動作に基づいて、適切なフリートタイプを選択します。たとえば、専用リソースを必要とするアプリケーションにはシングルセッションフリートを選択し、リソースを効率的に共有できるアプリケーションにはマルチセッションフリートを選択します。
- マルチセッション環境とのアプリケーションの互換性を検討してください。
- ファイルシステムのリダイレクト機能を使用して、リモートアプリケーションとローカルアプリケーション間のインタラクションを処理します。詳細については、 AWS ブログ記事「Launching local applications from an Amazon WorkSpaces Applications streaming session」を参照してください。

コスト最適化の柱

AWS Well-Architected フレームワークのコスト最適化の柱は、支出を最小限に抑えながらビジネス価値を最大化することに重点を置いています。クラウドリソースに費やすすべてのドルが、組織の目標の達成に効果的に貢献するのに役立ちます。

この柱を WorkSpaces Applications ストリーミング環境に適用するための主な重点領域:

- フリート容量管理とインスタンスタイプの選択
- スケーリングとスケジューリングの最適化
- 使用パターンのモニタリングと分析
- コスト配分と追跡

クラウド財務管理はどのように実装しますか?

構造化されたプログラムとプロセスを通じて、クラウド財務管理とコスト最適化に専念する組織能力を構築し、クラウドの価値と効率を最大化します。

- AWS Cost Explorer および 使用状況レポートを使用して WorkSpaces アプリケーションのコストをモニタリングし、ストリーミング時間の使用状況を追跡し、フリートインスタンスのコストを分析し、リージョンのコスト配分をモニタリングします。
- AWS Budgets を使用して、WorkSpaces Applications のサービスコスト全体のアラートを設定し、サービスの予算しきい値を作成し、予算額に対する実際の支出を監視することで、コストコントロールを計画および設定します。詳細については、 AWS ブログ記事「オートメーションを使用して Amazon WorkSpaces アプリケーションのコストを最適化および制御する方法」を参照してください。

消費モデルを追加する

実際の使用パターンに基づいてコンピューティングリソースとコストをスケールします。たとえば、営業時間外に非本番環境をシャットダウンして、支出を最適化できます。

- 適切な料金モデルを選択します。例えば、常時オンフリートを一貫した使用に使用し、オンデマンドフリートを可変ワークロードに使用します。

- 最適なインスタンスタイプを選択します。たとえば、一般的なアプリケーションには `stream.standard` インスタンスを使用し、必要な場合にのみグラフィックスインスタンス (G4dn) を使用します。

全体的な効率を測定する

ビジネス出力のcost-per-unitを計算して追跡し、効率の向上を定量化し、最適化の取り組みをガイドします。

- セッション効率を追跡します。
- 次の CloudWatch メトリクスを使用してフリートの使用率をモニタリングします。
  - `AvailableCapacity` 未使用の容量を追跡する
  - `InUseCapacity` 実際の使用量を測定するには
- ストリーミング時間あたりのコスト、ユーザーあたりのコスト、アプリケーションあたりのコストなど、セッションあたりのコストを計算して追跡します。
- WorkSpaces アプリケーションのコストオプティマイザーを実装して、ビルダーをモニタリングします。
- フリートタイプ間でコストを比較します。たとえば、以下を比較します。
  - 単一セッションとマルチセッションのライセンスコスト
  - リソース使用率
  - インスタンスあたりのユーザー密度
- プロセス追跡データを使用して、使用率の低いアプリケーションや不要なアプリケーションを特定します。詳細については、 AWS ブログ記事 Track user processes in Amazon WorkSpaces Applications sessions を参照してください。

差別化されていない重い作業にお金を費やすのをやめる

AWS はインフラストラクチャの運用を管理し、マネージドサービスを提供するため、組織は IT メンテナンスではなくビジネス目標に集中できます。

- Image Builder を使用してアプリケーションイメージを作成および維持し、アプリケーションのパッケージ化、アプリケーション設定の構成、アプリケーションの互換性のテストを行います。
- 適切なインスタンスタイプを選択し、スケーリングしきい値を定義し、希望する容量制限を設定して、フリートの仕様を設定します。

- Windows ベースのフリートの場合は Amazon S3 のホームフォルダを設定し、Linux ベースのフリートの場合は Amazon EFS の共有ファイルシステムを設定して、永続的ストレージオプションを設定します。ストレージのアクセス許可を設定し、保持ポリシーを定義します。

支出の分析と属性付け

クラウドにより、ワークロードあたりのリソース使用量とコストを正確に追跡できるため、投資収益率 (ROI) の正確な測定とターゲット最適化の機会が可能になります。

- コスト配分用のフリート、アセット追跡用のイメージ、環境指定用の Image Builder、組織グループ化用のスタックの包括的なタグ付け戦略を実装します。
- AWS コストと使用状況レポート (AWS CUR) を使用して、タグ付けされたリソース別に WorkSpaces アプリケーションのコストを分類し、フリート、スタック、イメージあたりのコストを分析します。
- AWS Cost Explorer を使用して WorkSpaces アプリケーションの支出傾向を視覚化し、リージョンやインスタンスタイプなどのさまざまなディメンションのコストを比較します。
- フリート使用率、インスタンスタイプの効率、ストリーミング時間をアプリケーション別にモニタリングおよび分析します。
- 未使用のリザーブドキャパシティ、使用率の低いフリートまたはスタック、フリート使用状況のアイドル期間を追跡します。
- 各アプリケーションのユーザーあたりのコスト、アプリケーションあたりのストリーミング時間、ストリーミングアプリケーションのユーザー導入率を計算して追跡します。
- WorkSpaces アプリケーションの使用状況レポートを設定し、Amazon Athena を使用して使用状況データをクエリし、Amazon Quick でビジュアライゼーションを作成してコストと使用状況に関するインサイトを取得することで、詳細な使用状況分析を設定します。
- Windows Server ライセンス、アプリケーションライセンスモデル、ユーザーごとのライセンスなどの総コストの考慮事項を、デバイスごとのライセンスと比較して評価します。
- Amazon Athena を使用して、ホームフォルダのストレージコストとユーザー別の使用パターンをクエリおよび分析します。詳細については、 AWS ブログ記事「How to report Amazon WorkSpaces Applications home folder use with Amazon Athena」を参照してください。

持続可能性の柱

AWS Well-Architected フレームワークの持続可能性の柱は、環境フットプリントを最小限に抑え、エネルギー使用量と効率を最適化することを強調しています。アーキテクトがシステム設計とリソース配分戦略で環境意識に基づいた意思決定を行うようにガイドします。

この柱を WorkSpaces アプリケーションストリーミング環境に適用するための主な重点領域:

- 実際の需要に合わせてリソース割り当てを理解して最適化し、ストリーミング環境の無駄を最小限に抑える
- アプリケーション配信とストリーミングセッションの効率を向上させるためのユーザー消費パターンの分析と適応
- 適切なハードウェア設定を選択して使用し、パフォーマンス要件を満たしながらエネルギー効率を最大化する
- AWS マネージドサービス機能を使用して、これらのサービスが提供するスケールメリットと組み込み効率機能を活用する

影響を把握する

出力単位あたりのリソース効率と排出量を測定することで、ワークロードの環境への影響をモニタリングおよび最適化します。このデータを使用して KPIs。

- フリート使用率パターンをモニタリングします。
- ユーザーあたりのストリーミング時間を追跡します。
- フリート容量の使用状況の傾向を分析します。

持続可能性の目標を設定する

組織の目標に沿った各ワークロードについて、測定可能な持続可能性目標を設定します。スケーリングするトランザクションあたりのリソース強度を減らすことに焦点を当てます。

- フリート使用率、インスタンスタイプの効率、ストリーミング時間の最適化のターゲットを設定します。
- 実際の使用パターンに基づいて容量を計画します。

使用率を最大化する

リソースのサイズを適正化し、使用率を最大化することで、ワークロードの効率を最適化します。アイドル容量を減らしてエネルギー消費を最小限に抑え、持続可能性を向上させます。

- 実際の需要に合わせて自動スケーリングを設定します。
- 使用パターンに基づいて適切なサイズのフリート容量。
- 適切な最小容量制限と最大容量制限を実装します。
- ワークロードに適したインスタンスタイプを選択します。
- ストリーミングセッションの密度をモニタリングして最適化します。
- オフピーク時のアイドル容量を減らします。

より効率的な新しいハードウェアおよびソフトウェアサービスを予測して採用する

パートナーやサプライヤーからの新しい効率的なテクノロジーを常に把握し、迅速に導入して、ワークロードの環境への影響を継続的に改善します。

- 現行世代のインスタンスタイプを使用します。
- 利用可能な場合は、新しいインスタンスタイプにアップグレードします。
- アプリケーションストリーミング設定を最適化します。
- 適切なストリーミングプロトコルを設定します。
- 最新の WorkSpaces アプリケーション機能を更新します。

使用済みマネージドサービス

共有クラウドサービスとマネージドソリューションを活用して、リソース使用率を最大化し、自動スケーリングとライフサイクル管理による環境への影響を最小限に抑えます。

- Windows ベースのフリートのユーザーストレージには Amazon S3 を使用し、Linux ベースのフリートの共有ファイルシステムには Amazon EFS を使用します。
- モニタリング用の CloudWatch を実装します。
- アクセス管理用に IAM を設定します。

クラウドワークロードのダウンストリームへの影響を軽減する

クライアント側のリソース要件を最小限に抑え、エネルギー消費を削減し、ユーザーのデバイス寿命を延長するサービスを設計します。

- 不要なリソースの消費を防ぐため、最大セッション時間を調整します。
- 適切なセッションタイムアウトを設定します。
- 切断タイムアウトポリシーを設定します。
- 必要に応じてセッション永続性ポリシーを実装します。

リソース

AWS ドキュメント

- AWS Well-Architected フレームワーク
- Amazon WorkSpaces アプリケーション管理ガイド
- Amazon CloudWatch ユーザーガイド
- Amazon EFSS ユーザーガイド
- Amazon S3 ユーザーガイド
- IAM ユーザーガイド

AWS ブログ投稿

- Active Directory グループのメンバーシップベースの WorkSpaces アプリケーションターゲティング
- Azure AD を使用してすべての Amazon WorkSpaces アプリケーションスタックに単一の ID プロバイダーを作成する
- Amazon WorkSpaces および Amazon WorkSpaces アプリケーションの Windows リモートアシスタンスの設定
- AS2TrustedDomains DNS TXT レコードを作成して Amazon WorkSpaces アプリケーションのネイティブクライアントをサードパーティーの ID プロバイダーにリダイレクトする
- Amazon WorkSpaces アプリケーションでのカスタムログ記録と Amazon CloudWatch アラートの作成
- Geo Targetly および Amazon WorkSpaces アプリケーションによるクロスリージョンリダイレクト
- クロスアカウントリソースと Amazon WorkSpaces アプリケーション
- バイオキー PortalGuard および Amazon WorkSpaces アプリケーションでフェデレーションを有効にする
- SimpleSAMLphp および Amazon WorkSpaces アプリケーションでフェデレーションを有効にする
- Duo Single Sign-On および Amazon WorkSpaces アプリケーションで ID フェデレーションを有効にする
- Shibboleth および Amazon WorkSpaces アプリケーションで ID フェデレーションを有効にする

- Amazon エンドユーザーコンピューティングを使用したオンプレミス VDI のフェイルオーバー戦略
- Amazon が Amazon WorkSpaces アプリケーションを使用してデータサイエンティストとアナリストに機密データへのアクセスを提供する方法
- Amazon WorkSpaces アプリケーションの証明書ベースの認証を設定する方法
- Amazon WorkSpaces アプリケーションのアプリケーションエンタイトルメントで Okta クレームを使用する方法
- オープンソースの仮想アプリケーション管理を使用した Amazon WorkSpaces アプリケーションのコンピュータラボの管理
- WorkSpaces アプリケーションのコストをビジネスユニットに割り当てる方法
- Amazon OpenSearch Service と Amazon Kinesis Data Firehose を使用した Amazon WorkSpaces アプリケーションのモニタリング
- Amazon WorkSpaces、Amazon WorkSpaces アプリケーション、Amazon WorkSpaces Amazon Macie を使用したネットワーク分離とデータサニタイズ
- Amazon WorkSpaces アプリケーションを使用した OneLogin SSO
- Amazon WorkSpaces アプリケーションで Amazon Connect コールオーディオパスを最適化する
- Amazon Elastic File System 上の Amazon WorkSpaces アプリケーション Linux フリートの永続的ストレージ
- Okta SAML アプリを Amazon WorkSpaces Applications ネイティブクライアントにリダイレクトする
- Application Masking を使用して Amazon WorkSpaces アプリケーションのイメージ管理を簡素化する
- Amazon WorkSpaces Applications Elastic フリートと Linux との互換性により、アプリケーションを低コストでストリーミングする
- WorkSpaces アプリケーションを使用した規制対象環境のインターフェイス VPC エンドポイントからのストリーミング
- Azure AD で Amazon WorkSpaces Applications アプリケーションの使用権限を使用する
- Amazon WorkSpaces アプリケーションのユーザー問題レポーター
- Google Workspace での Amazon WorkSpaces アプリケーションアプリケーションの使用権限の使用
- Amazon WorkSpaces アプリケーションでの Microsoft Active Directory での Auth0 の使用
- Microsoft AppLocker を使用した Amazon WorkSpaces アプリケーションのアプリケーションエクスペリエンスの管理

- Python を使用して WorkSpaces アプリケーション Linux イメージングアシスタント GUI を強化する
- WorkSpaces アプリケーションクライアントのウェブアプリケーションリダイレクトオプション

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
初版発行	—	2025 年 7 月 23 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

ABAC

「属性ベースのアクセス制御」をご覧ください。

抽象化されたサービス

「マネージドユーザー」をご覧ください。

ACID

「原子性、一貫性、分離性、耐久性 (ACID)」をご覧ください。

アクティブ/アクティブ移行

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

アクティブ/パッシブ移行

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

集計関数

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

AI

「人工知能」をご覧ください。

AIOps

「AI オペレーション」をご覧ください。

匿名化

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

アンチパターン

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

アプリケーション制御

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

アプリケーションポートフォリオ

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

人工知能 (AI)

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

AI オペレーション (AIOps)

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

非対称暗号化

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

原子性、一貫性、分離性、耐久性 (ACID)

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

属性ベースのアクセス制御 (ABAC)

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

信頼できるデータソース

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

アベイラビリティーゾーン (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、組織がクラウド導入を成功させるための準備を支援します。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。

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。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

「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、リソースの保護に役立つように、 AWS Organizations 構造の異なるレイヤーに複数のコントロールを追加します。たとえば、多層防御アプローチでは、多要素認証、ネットワークセグメンテーション、暗号化を組み合わせることができます。

委任管理者

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

Organizations ドキュメントの「AWS Organizationsで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 、ガバナンス要件のコンプライアンスに影響を与える可能性のあ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、障害の影響を制限し、ワークロードの耐障害性を高めるのに役立つアベイラビリティーゾーン 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。詳細については、「AWS 障害分離境界」を参照してください。

機能ブランチ

「ブランチ」を参照してください。

特徴量

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

特徴量重要度

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

機能変換

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

数ショットプロンプト

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

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

FGAC

「きめ細かなアクセス制御」を参照してください。

きめ細かなアクセス制御 (FGAC)

複数の条件を使用してアクセス要求を許可または拒否すること。

フラッシュカット移行

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

FM

「基盤モデル」を参照してください。

基盤モデル (FM)

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

G

生成 AI

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

ジオブロッキング

「地理的制限」を参照してください。

地理的制限 (ジオブロッキング)

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

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

Gitflow ワークフロー

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

ゴールデンイメージ

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

グリーンフィールド戦略

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

ガードレール

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

H

HA

「高可用性」を参照してください。

異種混在データベースの移行

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

高可用性 (HA)

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

ヒストリアンのモダナイゼーション

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

ホールドアウトデータ

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

同種データベースの移行

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

ホットデータ

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

ホットフィックス

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

ハイパーケア期間

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

I

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

製造実行システム (MES)

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

MAP

「Migration Acceleration Program」を参照してください。

メカニズム

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

メンバーアカウント

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

MES

「製造実行システム」を参照してください。

Message Queuing Telemetry Transport (MQTT)

発行/サブスク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 から分離され、独立しています。詳細については、「アカウントが使用できる AWS リージョン を指定する」を参照してください。

リグレッション

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

リホスト

「7 Rs」を参照してください。

リリース

デプロイプロセスで、変更を本番環境に昇格させること。

再配置

「7 Rs」を参照してください。

リプラットフォーム

「7 Rs」を参照してください。

再購入

「7 Rs」を参照してください。

回復性

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

リソースベースのポリシー

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

実行責任者、説明責任者、協業先、報告先 (RACI) に基づくマトリックス

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

レスポンシブコントロール

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

SCADA

「監視制御とデータ取得」を参照してください。

SCP

「サービスコントロールポリシー」を参照してください。

シークレット

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

セキュリティバイデザイン

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

セキュリティコントロール

脅威アクターによるセキュリティ脆弱性の悪用を防止、検出、軽減するための、技術上または管理上のガードレール。セキュリティコントロールには、主に 4 つの種類があります。4 つとは、予防、検出、レスポンシブ、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、 はクラウドのセキュリティを担当します。詳細については、「責任共有モデル」を参照してください。

SIEM

「Security Information and Event Management システム」を参照してください。

単一障害点 (SPOF)

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

SLA

「サービスレベルアグリーメント」を参照してください。

SLI

「サービスレベルインジケータ」を参照してください。

SLO

「サービスレベルの目標」を参照してください。

スプリットアンドシードモデル

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

SPOF

「単一障害点」を参照してください。

スタースキーマ

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

strangler fig パターン

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

サブネット

VPC 内の IP アドレスの範囲。サブネットは、1 つのアベイラビリティ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。

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「Using AWS Organizations with other AWS services AWS Organizations 」を参照してください。

チューニング

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

ツーピザチーム

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

U

不確実性

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