セキュリティの柱

AWS Well-Architected フレームワーク

2020 年 7 月

This paper has been archived.

The latest version is now available at:

https://docs.aws.amazon.com/ja_jp/wellarchitected/latest/security-pillar/welcome.html

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) AWS の現行製品と慣行ついて説明していますが、予告なしに変更されることがあり、(c) AWS およびその関連会社、サプライヤーまたはライセンサーからの契約上の義務や保証をもたらすものではありません。AWS の製品やサービスは、明示または暗示を問わず、一切の保証、表明、条件なしに「現状のまま」提供されます。お客様に対する AWS の責任は、AWS 契約により規定されます。本書は、AWS とお客様の間で締結される一切の契約の一部ではなく、その内容を修正することもありません。

目次

要約

このホワイトペーパーは、Well-Architected フレームワークのセキュリティの柱に焦点を当てています。お客様が安全な AWS ワークロードの設計、配信、メンテナンスにベストプラクティスと最新の推奨事項を適用するうえで役立つガイダンスを提供します。

はじめに

AWS Well-Architected フレームワークは、AWS でワークロードを構築するための意思決定におけるトレードオフの理解に役立ちます。効率が良く、費用対効果が高く、安全で信頼のおけるクラウド対応ワークロードを設計して運用するために、最新のアーキテクチャに関するベストプラクティスをフレームワークに従って学ぶことができます。このフレームワークにより、ワークロードをベストプラクティスに照らし合わせて一貫的に測定し、改善点を特定することが可能となります。ワークロードを適切に設計することで、ビジネスが成功する可能性が大いに高まると弊社は確信しています。

フレームワークは次の 5 つの柱に基づいています。

- 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

このホワイトペーパーは、セキュリティの柱に焦点を当てています。これを理解して最新の AWS の推奨事項に従うことで、ビジネス要件および規制要件を満たすことができます。この内容は、最高技術責任者 (CTO)、最高情報セキュリティ責任者 (CSO/CISO)、設計者、開発者、オペレーションチームメンバーなどの技術担当者を対象にまとめられています。

このホワイトペーパーを読むことで、セキュリティを念頭に置いてクラウドアーキテクチャを設計するための AWS の最新の推奨事項と戦略を理解できます。ここでは、実装の詳細やアーキテクチャのパターンについては説明していませんが、そのような情報に該当するリソースへの参照が記載されています。このホワイトペーパーにある手法を採用すれば、データとシステムを保護し、アクセスをコントロールし、セキュリティイベントに自動的に応答するアーキテクチャを構築できます。

セキュリティ

セキュリティの柱では、クラウドテクノロジーを活用して、データ、システム、資産の保護とセキュリティ体制の向上をはかる方法を説明します。このホワイトペーパーでは、AWS で安全なワークロードを設計するための詳細なベストプラクティスガイダンスを提供します。

設計の原則

クラウドには、次のようなワークロードのセキュリティ強化に役立つ原則があります。

- **強力なアイデンティティ基盤を実装する:** 最小権限の原則の適用と、職務分掌の徹底によって、適切な認証を用いた各 AWS リソースとの通信を実現します。ID 管理を一元化し、長期間にわたって一つの認証情報を使用し続けないようにします。
- **トレーサビリティを実現する:** 環境に対するアクションおよび変更をリアルタイムでモニタリング、警告、監査します。ログとメトリクスの収集をシステムに統合して、自動的に調査しアクションを実行します。
- **すべてのレイヤーでセキュリティを適用する:** 複数のセキュリティコントロールを使用して深層防御アプローチを適用します。ネットワークのエッジ、VPC、ロードバランシング、すべてのインスタンスとコンピューティングサービス、オペレーティングシステム、アプリケーション、コードなど、すべてのレイヤーに適用します。
- **セキュリティのベストプラクティスを自動化する:** ソフトウェアベースのセキュリティメカニズムの自動化によって、より迅速かつコスト効率の良い方法で、安全にスケールできるようになります。バージョン管理されているテンプレートにおいてコードとして定義および管理されるコントロールを実装するなど、セキュアなアーキテクチャを作成します。
- **転送中および保管中のデータを保護する:** 機密度レベルでデータを分類し、必要に応じて暗号化、トークン分割、アクセスコントロールなどのメカニズムを使用します。

- **データに人を近づけない:** データに対する直接的なアクセスや手動処理の必要性を低減または排除するためのメカニズムとツールを使用します。これにより、機密性の高いデータを扱う際の誤処理、改変、ヒューマンエラーのリスクを軽減します。
- **セキュリティイベントに対して備える:** 組織の要件に合わせたインシデント管理および調査のポリシーとプロセスを導入し、インシデントに備えます。インシデント対応シミュレーションを実行し、ツールとオートメーションにより、検出、調査、復旧のスピードを上げます。

定義

クラウドのセキュリティには、次の 5 つの領域があります。

1. アイデンティティ管理とアクセス管理
2. 検出
3. インフラストラクチャの保護
4. データ保護
5. インシデント対応

セキュリティとコンプライアンスは、AWS とお客様の共有責任です。この責任共有モデルは、お客様の運用上の負担を軽減するのに役立ちます。お客様の責任範囲は、使用するサービス、IT 環境へのサービス統合、適用可能な法律および規制に応じて異なります。したがって、お客様は選択するサービスを注意深く検討する必要があります。この共有責任の性質により、デプロイが可能となる柔軟性と制御がもたらされます。

ワークロードを安全に運用する

ワークロードを安全に運用するには、すべてのセキュリティ領域に包括的なベストプラクティスを適用する必要があります。組織レベルおよびワークロードレベルにおいて、「運用上の優秀性」で定義した要件とプロセスを抽出し、それらをすべての領域に適用します。AWS や業

界のレコメンデーションおよび脅威インテリジェンスを最新に保つことで、脅威モデルと管理の目標を進化させることができます。セキュリティプロセス、テスト、検証を自動化することで、セキュリティオペレーションを拡張できます。

脅威モデルを使用してリスクを特定し、優先順位を付ける: 脅威モデルを使用して潜在的な脅威を特定し、その登録を最新の状態に維持します。脅威に優先順位を付け、セキュリティコントロールを調整して防止、検出、対応を行います。進化するセキュリティ環境の状況に応じてセキュリティコントロールを再確認および維持します。

管理目標の特定と検証: 脅威モデルから特定されたコンプライアンス要件とリスクに基づいて、ワークロードに適用する必要がある管理目標および管理を導き出し、検証します。管理目標と制御を継続的に検証することは、リスク軽減の効果測定に役立ちます。

セキュリティの脅威情報を常に最新に保つ: 最新のセキュリティ脅威を常に把握して攻撃ベクトルを認識し、適切な管理を定義して実装できるようにします。

セキュリティに関する推奨事項を最新に保つ: AWS と業界の両方のセキュリティ推奨事項を常に最新に保ち、ワークロードのセキュリティ体制を進化させます。

新しいセキュリティサービスと機能を定期的に評価および実装する: ワークロードのセキュリティ体制を進化させることができる AWS、および APN パートナーのセキュリティサービスと機能を評価および実装します。

パイプラインのセキュリティコントロールのテストと検証を自動化する: ビルド、パイプライン、プロセスの一環としてテストおよび検証されるセキュリティメカニズムの安全なベースラインとテンプレートを確立します。ツールとオートメーションを使用して、すべてのセキュリティコントロールの継続的なテストと検証を実施します。たとえば、マシンイメージやインフラストラクチャなどの項目をコードテンプレートとしてスキャンして、セキュリティの脆弱性、不規則性、ドリフトを各ステージで確立されたベースラインから確認します。

本番環境に取り込まれるセキュリティの誤設定の数を減らすことが非常に重要です。ビルドプロセスでより適切な品質管理をより多く実行し、欠陥の数を減らすことができれば、より優れたものになります。継続的インテグレーションおよび継続的デプロイ (CI/CD) のパイプライ

ンは、可能な限りセキュリティの問題をテストできるように設計する必要があります。CI/CD パイプラインは、ビルドと配信の各段階でセキュリティを強化する機会を提供します。CI/CD セキュリティツールも更新して、進化する脅威を軽減する必要があります。

リソース

ワークロードの安全な運用方法の詳細については、以下のリソースを参照してください。

動画

- Well-Architected の手法によるセキュリティのベストプラクティス
- 自動化とガバナンスにより AWS の大規模な採用を可能にする
- AWS Security Hub: セキュリティアラートの管理とコンプライアンスの自動化
- AWS のセキュリティを自動化する

ドキュメント

- セキュリティプロセスの概要
- セキュリティ速報
- セキュリティブログ
- AWS の最新情報
- AWS セキュリティ監査ガイドライン
- AWS で CI/CD パイプラインを設定する

AWS アカウントの管理と分離

AWS では、社内のレポート構造を流用せずに、個別アカウントごとにワークロードを整理し、機能、コンプライアンス要件、共通のコントロールセットに基づいてアカウントをグループ化することを推奨しています。AWS において、アカウントは確固とした境界線を持つ、リソー

スを守るゼロトラストのコンテナです。たとえば、開発およびテストのワークロードと本番ワークロードを切り離すために、アカウントレベルの分離を強く推奨しています。

アカウントを使用してワークロードを分離する: セキュリティとインフラストラクチャを念頭に置いて、ワークロードが増大するにつれて組織が共通のガードレールを設定できるようにします。このアプローチによって、ワークロード間の境界と制御が確立します。開発環境およびテスト環境から本番環境を分離する場合、または外部コンプライアンス要件 (PCI-DSS や HIPAA など) で定義されている機密レベルの異なるデータを処理するワークロードとそうでないワークロードとの間に強力な論理的境界を設ける場合は、アカウントレベルの分離を強くお勧めします。

AWS アカウントを保護する: AWS アカウントの保護には、ルートユーザーの保護および使用の回避、連絡先情報を最新の状態に保つなど、さまざまな側面があります。AWS Organizationsを使用すれば、ワークロードの拡大やスケーリングに合わせて、アカウントを一元管理できます。AWS Organizations は、アカウント全体の管理、制御の設定、サービスの構築に役立ちます。

アカウントを一元管理する: AWS Organizations は、AWS アカウントの作成と管理、およびアカウント作成後の制御を自動化します。AWS Organizations を使用してアカウントを作成する場合、使用する E メールアドレスの検討が重要です。これがパスワードリセットを許可するルートユーザーとなるためです。組織は、要件とワークロードの目的に応じた異なる環境である組織単位 (OU) でアカウントをグループ化できます。

制御を一括設定する: 特定のサービス、リージョン、サービスアクションのみを適切なレベルで許可することによって、AWS アカウントが実行できる操作を制御します。AWS Organizations では、サービスコントロールポリシー (SCP) を使用して、組織レベル、組織単位、アカウントレベルでアクセス許可ガードレールを適用できます。これは、すべての AWS Identity and Access Management (IAM) ユーザーおよびロールに適用されます。たとえば、SCP を適用して、明示的に許可されていないリージョンにいるユーザーがリソースを起動することを制限できます。AWS Control Tower では、複数アカウントの効率的な設定と管理が可能です。このサービスを使うと、AWS Organization のアカウント設定の自動化、プロビジョニングの自動化、ガードレール (予防や検出など) の適用、ダッシュボードによる可視化を実現できます。

サービスとリソースを一括設定する: AWS Organizations では、すべてのアカウントに適用する AWS のサービスを設定できます。たとえば、組織全体で実行されるすべてのアクションの集中ログ記録を AWS CloudTrail で構築し、メンバーアカウントがログ記録を無効化しないように設定できます。他にも AWS Config では、定義したルールを元にデータを一元的に集約することもできます。これによってワークロードのコンプライアンス監査と、変更への迅速な対応が可能となります。AWS CloudFormation StackSets を使用すると、組織内の複数のアカウントと OU にまたがる AWS CloudFormation スタックを集中管理できます。これによって、新しいアカウントを自動的にプロビジョニングしてセキュリティ要件を満たすことができます。

リソース

複数の AWS アカウントのデプロイと管理に関する AWS の推奨事項の詳細については、以下のリソースを参照してください。

動画

- AWS Organizations を使用したマルチアカウント AWS 環境の管理と統制
- AXA: グローバルなランディングゾーンによる導入のスケーリング
- AWS Control Tower を使用したマルチアカウント AWS 環境の統制

ドキュメント

- ベストプラクティスに基づく AWS 環境の確立
- AWS Organizations
- AWS Control Tower
- AWS CloudFormation StackSets の使用
- サービスコントロールポリシーを使用して、AWS Organization のアカウント間に許可ガードレールを設定する方法

ハンズオン

- ラボ: AWS アカウントおよびルートユーザー

アイデンティティ管理とアクセス管理

AWS のサービスを使用するには、ユーザーとアプリケーションに AWS アカウントのリソースへのアクセス権限を与える必要があります。AWS で実行するワークロードの増加に伴い、適切なユーザーが適切な条件で適切なリソースにアクセスできるようにするためには、強固な ID 管理とアクセス許可が必要です。AWS は、幅広い機能の選択肢を提供することによって、ユーザーとマシンの ID および権限の管理を支援しています。これらの機能のベストプラクティスは、次の 2 つの領域に大きく分類されます。

- ID 管理
- 権限管理

ID 管理

AWS ワークロードを安全に運用するには、2 種類の ID を管理する必要があります。

ユーザー ID: 管理者、開発者、オペレーター、アプリケーションのエンドユーザーは、AWS 環境とアプリケーションにアクセスできる ID が必要です。これらのユーザーは、あなたの組織のメンバー、または共同作業を行う外部ユーザーで、ウェブブラウザ、クライアントアプリケーション、モバイルアプリ、インタラクティブなコマンドラインツールを介して AWS リソースを操作します。

マシン ID: ワークロードアプリケーション、運用ツール、コンポーネントには、データ読み取りなどのため、AWS のサービスにリクエストを送信できる ID が必要です。このような ID には、Amazon EC2 インスタンスや AWS Lambda 関数など、AWS 環境で実行されているマシンが含まれます。また、アクセスを必要とする外部関係者のマシン ID を管理することもできます。さらに、AWS 環境にアクセスする必要があるマシンが AWS 外にある場合もあります。

一元化された ID プロバイダーを利用する

ユーザー ID の場合、ID を一元管理できる ID プロバイダーを利用します。一つの場所から権限の作成、管理、取り消しを行うため、複数のアプリケーションおよびサービスに影響する権限を効率的に管理できます。たとえば誰かが組織を離れる場合、すべてのアプリケーションとサービス (AWS を含む) へのアクセスを一つの場所で取り消すことができます。これにより、複数の認証情報を用意する必要性がなくなり、既存の人事 (HR) プロセスと統合できる可能性が生まれます。

AWS の個別アカウントのフェデレーションでは、AWS IAM を使った SAML 2.0 ベースのプロバイダーに AWS の一元化された ID を使用できます。これには SAML 2.0 プロトコルと互換性のあるプロバイダーであればいずれも使用できます。AWS でホストされているかどうか、AWS 外部にあるかどうか、AWS パートナーネットワーク (APN) から提供されているかどうかは問いません。AWS アカウントと選択したプロバイダーのフェデレーションを使用して、SAML アサーションで一時的なセキュリティ認証情報を取得すれば、ユーザーまたはアプリケーションに AWS API オペレーションを呼び出すアクセス権限を付与できます。ウェブベースのシングルサインオンもサポートされており、ユーザーはサインインポータルから AWS マネジメントコンソールにサインインできます。

AWS Organization の複数のアカウントに対するフェデレーションでは、AWS Single Sign-On (AWS SSO) でアイデンティティソースを設定し、ユーザーとグループの保存先を指定できます。設定が完了すると、ID プロバイダーが真のソースとなり、クロスドメイン ID 管理システム (SCIM) の v2.0 プロトコルを使用して情報を同期できます。その後ユーザーまたはグループを検索し、AWS アカウントやクラウドアプリケーションへのシングルサインオンアクセスを付与できます。

AWS SSO は AWS Organizations と統合されているため、ID プロバイダーを一度設定するだけで、組織が管理する既存および新規アカウントへのアクセス権を付与できます。
AWS SSO には、ユーザーとグループを管理できるデフォルトストアがあります。AWS SSO ストアを使用する場合は、ユーザーとグループを作成してから、最小権限のベストプラクティスに基づきそのアクセスレベルを必要な AWS アカウントとアプリケーションに割り当てます。ま

たは、SAML 2.0 を使用して外部の ID プロバイダーに接続するか、AWS Directory Service を使用して Microsoft AD ディレクトリに接続することもできます。設定が完了したら、一元化された ID プロバイダーで認証すれば、AWS マネジメントコンソール、コマンドラインインターフェイス、AWS モバイルアプリにサインインできるようになります。

モバイルアプリなどのワークロードのエンドユーザー管理には、Amazon Cognitoを使用できます。このサービスには、ウェブおよびモバイルアプリケーションの認証、承認、ユーザー管理の機能があります。ユーザーは、ユーザー名とパスワードを使用して直接サインインするか、Amazon、Apple、Facebook、Google などのサードパーティーを通じてサインインできます。

ユーザーグループと属性を活用する

管理対象のユーザー数が増えるにつれて、大規模な管理ができるユーザー管理方法が必要となります。一般的なセキュリティ要件を持つユーザーを ID プロバイダーで定義したグループに分け、アクセスコントロールに使用される可能性のあるユーザー属性 (部署や場所など) を最新で正確な状態に保つメカニズムを導入します。アクセス制御には、個々のユーザーではなくこのグループと属性を使用します。これにより、ユーザーのアクセスニーズが変化したときに複数のポリシーをそれぞれ更新する必要がなくなります。アクセス許可セットでユーザーのグループメンバーシップや属性を一度変更するだけで、アクセスを一元管理できます。ユーザーグループと属性の管理には、AWS SSO を使用します。AWS SSO は、一般的に使用されている属性に対応しています。ユーザー作成時の手動入力も、クロスドメイン ID 管理システム (SCIM) 仕様などで定義された同期エンジンを使用した自動プロビジョニングも可能です。

強力なサインインメカニズムを使用する

パスワードに最小の長さを設定し、よくあるパスワードや再利用を避けるようにユーザーを教育します。ソフトウェアまたはハードウェアのメカニズムを使用した Multi-Factor Authentication (MFA) を強制して、検証を追加します。たとえば AWS SSO をアイデンティティソースとして使用する場合、MFA の「コンテキストアウェア」または「常時オン」設定でユーザーに独自の MFA デバイスの登録を許可すると、すばやく使用開始できます。外部の ID プロバイダー (IdP) を使用する場合は、IdP を MFA 用に設定します。

一時的な認証情報を使用する

ID を使って一時的な認証情報を動的に取得します。ユーザー ID の場合、AWS SSO または IAM のフェデレーションを使用して AWS アカウントにアクセスします。EC2 インスタンスや Lambda 関数などのマシン ID の場合、長期的なアクセスキーを持つ IAM ユーザーではなく IAM ロールを使用する必要があります。

AWS マネジメントコンソールを使用するユーザー ID の場合、ユーザーは一時的な認証情報の取得と AWS へのフェデレーションが必要です。これは、AWS SSO ユーザーポータルを使用するか、IAM でフェデレーションを設定して実行します。CLI アクセスが必要なユーザーの場合、AWS Single Sign-On (AWS SSO) との直接統合に対応する AWS CLI v2を使用するようにしてください。ユーザーは、AWS SSO アカウントおよびロールにリンクされた CLI プロファイルを作成できます。CLI は AWS の認証情報を AWS SSO から自動的に取得し、ユーザーに代わってその情報を更新します。これにより、AWS SSO コンソールから一時的な AWS の認証情報をコピーして貼り付ける作業を省略できます。SDK については、ユーザーは AWS STS を使用して、一時的な認証情報を取得するロールを引き受ける必要があります。場合によって、一時的な認証情報が使用できないこともあります。アクセスキーを保存するリスクに注意しながら頻繁に更新し、可能なら MFA を条件として設定する必要があります。

エンドユーザーに AWS リソースへのアクセスを許可する必要がある場合は、Amazon Cognito の ID プールで権限が制限された一時的な認証情報を割り当てます。各ユーザーの権限は、作成した IAM ロール で制御されます。ユーザーのロールを選択するルールは、ユーザーの ID トークンの登録に基づいて定義します。認証済みユーザーにはデフォルトのロールを定義します。認証されていないゲストユーザーには、制限付きのアクセス権限を持つ IAM ロールを個別に定義できます。

マシン ID に AWS へのアクセス許可を付与するには IAM ロールが必要です。EC2 インスタンスには、Amazon EC2 のロールを使用できます。IAM ロールを EC2 インスタンスにアタッチすると、Amazon EC2 で実行されているアプリケーションは、AWS が自動的に作成、配布、ローテーションする一時的なセキュリティ認証情報を使用できるようになります。キーまたはパスワードを使用して EC2 インスタンスにアクセスする場合、AWS Systems Manager を使用す

ると、保存されたシークレットなしで安全に、インストール済みエージェントを使用したインスタンスにアクセスおよび管理できます。さらに、AWS Lambda などの AWS のサービスでは、IAM サービスロールを設定して、一時的な認証情報でサービスに AWS アクションを実行する権限を与えることができます。

定期的に認証情報を監査およびローテーションする

正しい制御が実施されていることを確認するには、定期的な検証、できれば自動化されたツールによる検証が必要です。ユーザー ID の場合、ユーザーにはパスワードの定期的な変更と、一時的な認証情報を優先したアクセスキーの削除を要求する必要があります。また、ID プロバイダーの MFA 設定を継続的にモニタリングすることを推奨します。AWS Config Rules を設定すれば、これらの設定のモニタリングが可能です。マシン ID の場合、IAM ロールを使用した一時的な認証情報を使用する必要があります。それが不可能なときは、アクセスキーの監査および更新の頻度を高めることが重要です。

シークレットを安全に保存して使用する

IAM に関係がない認証情報 (データベースのログインなど) については、AWS Secrets Manager などのシークレット管理用に設計されたサービスを使用します。AWS Secrets Manager では、サポートされているサービスを使用して、暗号化されたシークレットの管理、ローテーション、安全な保管を簡単に行うことができます。シークレットにアクセスするための呼び出しは、監査用に CloudTrail に記録されます。IAM のアクセス許可を使用すれば、それに最小権限のアクセス許可を付与することが可能です。

リソース

AWS 認証情報の保護に関する AWS のベストプラクティスの詳細については、以下のリソースを参照してください。

動画

- すべての層でアイデンティティをマスターする

- AWS SSO を使用した大規模なユーザー権限の管理
- シークレットを大規模に管理、取得、変更するためのベストプラクティス

ドキュメント

- AWS アカウントのルートユーザー
- AWS アカウントのルートユーザーの認証情報とIAM ユーザーの認証情報
- IAM のベストプラクティス
- IAM ユーザー用のアカウントパスワードポリシーの設定
- AWS Secrets Manager の開始方法
- インスタンスプロファイルの使用
- 一時的なセキュリティ認証情報
- ID プロバイダーとフェデレーション

権限管理

権限管理によって、AWS とワークロードへのアクセスを必要とするユーザー ID やマシン ID へのアクセスをコントロールします。権限を分けることで、どのような条件で誰が何にアクセスできるかを制御します。特定のユーザー ID およびマシン ID にアクセス権限を設定し、必要とするリソースに対するサービスアクションへのアクセスのみを許可します。さらに、アクセスを取得するために満たすべき条件を指定します。たとえば、新しい Lambda 関数を作成する開発者の権限を特定のリージョンのみに指定できます。大規模な AWS 環境を管理する場合、以下のベストプラクティスに従って、それぞれのアイデンティティに必要なアクセスのみを許可し、必要以上に設定しないようにします。

組織のアクセス許可ガードレールを定義する

AWS で管理するワークロードの増加に伴い、アカウントを使用してワークロードを分離し、AWS Organizations を使用してそのアカウントを管理する必要があります。組織内のすべての ID へのアクセスを制限するために、共通のアクセス許可ガードレールの確立を推奨しています。たとえば、特定の AWS リージョンへのアクセスを制限したり、中央セキュリティチームが使用する IAM ロールなどの共通リソースをチームのメンバーが削除できないようにしたりできます。これを実行するには、ユーザーによるキーサービスの無効化を防止するなどのサービスコントロールポリシーの例を実装します。

AWS Organizations では、アカウントのグループ化と、アカウントグループごとの共通コントロールの設定ができます。このような共通コントロールを設定するには、AWS Organizations と統合されているサービスを使用します。具体的には、サービスコントロールポリシー (SCP) を使用してアカウントのグループへのアクセスを制限します。SCP は IAM ポリシー言語を使用し、すべての IAM プリンシパル (ユーザーとロール) が従うコントロールを確立します。特定の条件に基づいて、特定のサービスアクションおよびリソースへのアクセスを制限することによって、組織のアクセスコントロールのニーズを満たすことができます。ガードレールには、必要に応じて例外を定義できます。たとえば、アカウント内の特定の管理者ロールを除くすべての IAM エンティティに対して、サービスアクションを制限します。

最小権限のアクセスを付与する

最小権限の原則を設定することで、特定のタスクを実行するために必要な最小限の機能セットのみをアイデンティティが実行できるようになり、使いやすさと効率のバランスを取ることができます。この原則を適用すると、意図しないアクセスは制限され、誰がどのリソースにアクセス権限があるかを監査できます。AWS では、ルートユーザーを除く ID にはデフォルトのアクセス権限がありません。このルートユーザーは、特定のタスクにのみ使用する必要があります。

ポリシーを使用すると、フェデレーティッド ID、マシン、リソース (S3 バケットなど) が使用する IAM ロールなどの IAM やリソースエンティティにアタッチされたアクセス許可を明示的に付与できます。ポリシーの作成およびアタッチでは、AWS がアクセスを許可するために必要なサ

ービスアクション、リソース、条件を指定できます。AWS では、アクセスを最小限にするために役立つさまざまな条件を用意しています。たとえば、`PrincipalOrgID` の条件キーを使用すると、AWS Organizations の識別子が検証されるため、AWS Organization 内でアクセスを許可できます。AWS のサービスがユーザーに代わって行うリクエスト (AWS CloudFormation による AWS Lambda 関数の作成など) を制御するには、`CalledVia` の条件キーを使用します。このようにして、AWS 全体のユーザー ID とマシン ID に詳細なアクセス許可を設定できます。

AWS には、最小限の権限に従いつつアクセス権限管理を拡張できる機能も用意されています。

アクセス許可境界: アクセス許可の境界を使用して、管理者が設定できるアクセス許可の上限を設定できます。これによって、IAM ロール作成などのアクセス権限の作成および管理の権限を開発者に委任しながらも、付与できるアクセス権限を制限して、自分でその権限の範囲を拡大できないように制限できます。

属性ベースのアクセスコントロール (ABAC): AWS では、属性に基づいてアクセス権限を付与することができます。AWS では、これをタグと呼びます。タグは、IAM プリンシパル (ユーザーまたはロール) と AWS リソースにアタッチできます。IAM ポリシーを使うと、管理者は再利用可能なポリシーを作成して IAM プリンシパルの属性に基づいたアクセス許可を適用できます。たとえば、管理者は一つの IAM ポリシーを使用して、開発者のプロジェクトタグに一致する AWS リソースへのアクセス権を組織内の開発者に付与できます。開発者チームがプロジェクトにリソースを追加すると、属性に基づきそれに対するアクセス許可が自動的に適用されます。このため、リソースが追加されるたびにポリシーを更新する必要はありません。

パブリックおよびクロスアカウントアクセスの分析

AWS では、別のアカウントにあるリソースへのアクセス権を許可できます。直接クロスアカウントアクセスを付与するには、リソースにアタッチされたポリシー (S3 バケットポリシーなど) を使用するか、アイデンティティが別のアカウントの IAM ロールを引き受けることを許可します。リソースポリシーを使用する場合、組織内の ID へのアクセスを許可し、リソースを公開するタイミングを検討する必要があります。このアクションによってリソースに誰でもアクセス可能になるため、リソースの公開は最小限にする必要があります。IAM Access Analyzer は、

数学的方法 (証明可能セキュリティ) を使用して、アカウント外からリソースへのすべてのアクセスパスを識別します。また、リソースポリシーの継続的な確認と、パブリックおよびクロスアカウントアクセスの結果の報告により、広範囲なアクセス権の分析をしやすくします。

リソースを安全に共有する

複数のアカウントでワークロードを管理する場合、それらのアカウント間でリソースを共有するケースがあります。そのような場合は、AWS Resource Access Manager (AWS RAM) を使用してリソースを共有することをお勧めします。このサービスを使用すると、AWS 組織および組織単位内で AWS リソースを簡単かつ安全に共有できます。AWS RAM を使用すると、共有されている組織または組織単位内外へのアカウントの移動に伴い、共有リソースへのアクセスの許可または取り消しが自動的に行われます。これにより、確実に意図したアカウントでのみリソースを共有できます。

アクセス許可を継続的に削減する

チームやプロジェクトが開始した直後には、イノベーションと俊敏性を引き出すため、幅広いアクセス権の付与を選択する場合があります。アクセス権を継続的に評価し、必要な許可のみにアクセスを制限し、最小権限を付与することをお勧めします。AWS では、未使用のアクセス権を特定するのに役立つアクセス分析機能を提供しています。未使用のユーザーとロールを特定しやすくするため、AWS はアクセスアクティビティを分析し、最後に使用されたアクセスキーとロールの情報を提供します。最終アクセス時間タイムスタンプを使用して、未使用のユーザーとロールを特定し、それらを削除できます。さらに、サービスとアクションの最終アクセス時間情報を確認し、特定のユーザーおよびロールのアクセス許可を特定して強化することができます。 たとえば、最終アクセス時間情報を使用して、アプリケーションロールが必要とする特定の S3 アクションを特定し、それらのアクションのみにアクセスを制限できます。これらの機能は、コンソールおよびプログラムで使用でき、インフラストラクチャワークフローや自動化ツールに組み込むことができます。

緊急アクセスのプロセスを確立する

自動プロセスまたはパイプラインの問題が発生した場合に、ワークロード (特に AWS アカウント) への緊急アクセスを許可するプロセスが必要です。このプロセスには、アクセス用の緊急 AWS クロスアカウントロール、または管理者が緊急リクエストの検証と承認を行う際の特定のプロセスなど、さまざまな機能の組み合わせが含まれる場合があります。

リソース

きめ細かい承認に関する最新の AWS のベストプラクティスの詳細については、以下のリソースを参照してください。

動画

- 60 分以内に IAM ポリシーマスターになる
- 職務分離、最小権限、委任、CI/CD

ドキュメント

- 最小権限を付与する
- IAM ポリシーを管理する
- IAM ユーザー、グループ、認証情報を管理するためのアクセス許可の委任
- IAM Access Analyzer
- 必要でない認証情報を削除する
- MFA を使用して CLI でロールを引き受ける
- アクセス許可境界
- 属性ベースのアクセスコントロール (ABAC)

ハンズオン

- ラボ: ロールの作成を委任する IAM アクセス許可境界
- ラボ: EC2 の IAM タグベースのアクセスコントロール
- ラボ: Lambda クロスアカウント IAM ロールの引き受け

検出

検出により、潜在的なセキュリティ設定の誤り、脅威、予期しない動作を特定できます。検出はセキュリティライフサイクルの最重要部分であり、品質管理プロセス、法的義務またはコンプライアンス義務、脅威の特定とその対応をサポートします。検出メカニズムにはさまざまなタイプがあります。たとえば、ワークロードのログは、使用された脆弱性を分析できます。ワークロードに関連する検出メカニズムを定期的に見直し、内部および外部のポリシーと要件を満たしていることを確認する必要があります。自動化されたアラートと通知は、チームやツールが調査できるように、定義された条件に基づいて行う必要があります。これらのメカニズムは、組織が異常なアクティビティの範囲を特定し把握するのに役立つ重要な対応機能です。

AWS には、検出メカニズムに対処する際に使用できるアプローチが多数あります。以下の各セクションでは、こうしたアプローチの使用方法を説明します。

- 設定
- 調査

設定

サービスとアプリケーションのログ記録の設定: 基本的なプラクティスは、アカウントレベルで一連の検出メカニズムを確立することです。この基本的なメカニズムセットは、アカウント内のすべてのリソースに対する幅広いアクションを記録および検出することを目的としています。これらを使用すると、自動修復を含むオプションを備えた包括的な検出機能、および機能を追加するためのパートナー統合を構築できます。

AWS では、この基本セットのサービスには以下が含まれます。

- AWS CloudTrail では、AWS マネジメントコンソール、AWS SDK、コマンドラインツールなどの AWS のサービスを通じて実行されたアクションを含む、AWS アカウントアクティビティのイベント履歴を提供します。

- AWS Config では、AWS リソース構成のモニタリングと記録が行われ、目標の構成に対する評価と修復が自動化できます。
- Amazon GuardDuty は脅威検出サービスです。悪意のある動作や不正な動作を継続的にモニタリングし、AWS のアカウントとワークロードを保護できるようにします。
- AWS Security Hub では、複数の AWS サービスや任意のサードパーティー製品からのセキュリティアラートまたは検出結果の集約、整理、優先順位付けが一元的に行われ、セキュリティアラートとコンプライアンスステータスを包括的に把握できます。

Amazon Virtual Private Cloud (VPC) など、多くの主要な AWS サービスは、アカウントレベルの基盤上に構築されており、サービスレベルのログ記録機能を提供します。VPC フローログを使用すると、ネットワークインターフェイスを出入りする IP トラフィックに関する情報をキャプチャし、接続履歴に関する貴重なインサイトを得て、異常な動作に基づいた自動アクションをトリガーできます。

EC2 インスタンスや AWS のサービスから生成されないアプリケーションベースのログ記録の場合、ログは Amazon CloudWatch Logs を使用して保存、分析できます。エージェントは、オペレーティングシステムと実行中のアプリケーションからログを収集し、自動的に保存します。ログが CloudWatch Logs で利用可能になったらリアルタイムで処理したり、Insights を使用して分析したりできます。

複雑なアーキテクチャによって生成される大量のログとイベントデータから意味のある情報を抽出する機能は、ログの収集や集約と同様に重要な機能です。詳細については、信頼性の柱についてのホワイトペーパーのモニタリングセクションを参照してください。CloudWatch Logs エージェントがキャプチャするログファイルにアプリケーションデータが誤って入ってきた場合や、クロスリージョンロギングがログ集約用に設定されていて、国境を越えて特定の種類の情報を送付することに関する法的考慮事項がある場合など、ログ自体に機密とみなされるデータが含まれる可能性があります。

1 つのアプローチとして、ログの配信時にイベントでトリガーされる Lambda 関数を使用して、S3 バケットなどの中央ログ記録の場所に転送する前にログデータをフィルタリング、編集できます。未編集のログは、法律および法務チームが定める「妥当な時間」が経過するまでロ

ーカルバケットに保持できます。経過した時点で、S3 ライフサイクルルールが自動的にログを削除できます。S3 Object Lock を使用して Amazon S3 でログの保護を強化できます。S3 Object Lock では、Write-Once-Read-Many (WORM) モデルを使用してオブジェクトを保存できます。

ログ、検出結果、メトリクスの一元分析: セキュリティ運用チームは、ログを収集し、検索ツールを使用することで関心のある潜在的なイベントを検出しますが、その場合、不正なアクティビティや意図しない変更が発生する可能性があります。ただし、収集されたデータを分析して手動で情報を処理するだけでは、複雑なアーキテクチャから流れる大量の情報に対応するには不十分です。分析とレポートだけでは、適切なリソースを割り当てて、イベントをタイミング良く実行する作業が容易になる訳ではありません。

熟練したセキュリティオペレーションチームを構築するには、セキュリティイベントと調査結果の流れを、チケットシステム、バグ/問題システム、その他のセキュリティ情報とイベント管理 (SIEM) システムなどの、通知およびワークフローシステムに深く統合することをお勧めします。これにより、メールや静的レポートからワークフローが排除され、イベントや調査結果のルーティング、エスカレート、管理が可能になります。多くの組織はセキュリティアラートをチャット/コラボレーションや開発者の生産性プラットフォームに統合しています。自動化に着手している組織は、API 主導の、低レイテンシーのチケット発行システムによって、「何を最初に自動化するか」を計画する際にかなりの柔軟性が得られます。

このベストプラクティスは、ユーザーアクティビティやネットワークイベントを示すログメッセージから生成されたセキュリティイベントだけでなく、インフラストラクチャ自体で検出された変更から生成されたセキュリティイベントにも適用できます。変更による影響が小さく、IAM と Organizations の設定の組み合わせではその実行を阻止できないような状況下では、変更を検出し、変更が適切かどうかを判断し、その情報を正しい修復ワークフローにルーティングする機能は、安全なアーキテクチャを維持、検証するうえで不可欠です。

GuardDuty と Security Hub は、他の AWS のサービスでも利用できるログレコードの集約、重複排除、分析メカニズムを提供します。具体的には、GuardDuty は、VPC DNS サービスからの情報と、CloudTrail および VPC フローログで確認できる情報を取り込み、集約し、分析しま

す。Security Hub は、GuardDuty、AWS Config、Amazon Inspector、Macie、AWS Firewall Manager、さらには AWS Marketplace で購入できる多数のサードパーティー製セキュリティ製品、また適切に作成された独自コードからの出力を取り込んで集約し、分析できます。GuardDuty と Security Hub のどちらにも、複数のアカウントにわたって調査結果とインサイトを集約できるマスターメンバーモデルがあります。Security Hub は、オンプレミスの SIEM を導入しているお客様に AWS 側のログ/アラートのプリプロセッサ/アグリゲータとしてよく使用され、お客様はそこから Lambda ベースのプロセッサとフォワーダーを介して Amazon EventBridge を取り込むことができます。

リソース

ログのキャプチャと分析に関する現在の AWS の推奨事項の詳細については、以下のリソースを参照してください。

動画

- クラウドにおける脅威管理: Amazon GuardDuty と AWS Security Hub
- リソースの設定とコンプライアンスを一元的にモニタリングする

ドキュメント

- Amazon GuardDuty をセットアップする
- AWS Security Hub
- 開始方法: Amazon CloudWatch Logs
- Amazon EventBridge
- CloudTrail ログを分析するための Athena の設定
- Amazon CloudWatch
- AWS Config
- CloudTrail で証跡を作成する
- 集中ロギングソリューション

ハンズオン

- ラボ: セキュリティハブを有効化する
- ラボ: 発見的統制の自動デプロイ
- ラボ: Amazon GuardDuty ハンズオン

調査

実行可能なセキュリティイベントを実装する: 使用する検知メカニズムごとに、ランブックまたはプレイブック形式の調査プロセスも用意する必要があります。たとえば、Amazon GuardDuty を有効にすると、さまざまな調査結果が生成されます。調査結果タイプごとにランブックエントリが必要です。たとえば、トロイの木馬が検出された場合、調査して修復するよう指示する簡単な説明をランブックに記載する必要があります。

イベントへの対応を自動化する: AWS では、Amazon EventBridge を使用して、関心のあるイベントと予期しない変更についての情報の調査を自動化されたワークフローに組み込むことができます。このサービスには、スケーラブルなルールエンジンが備わっており、ネイティブの AWS イベント形式 (CloudTrail イベントなど)、独自のアプリケーションから生成できるカスタムイベントの両方を仲介できます。Amazon EventBridge では、インシデントレスポンスシステム (Step Functions) を構築するワークフローシステムや中央のセキュリティアカウントにイベントをルーティングできます。また、バケットにルーティングして詳細分析を実行することもできます。

変更を検出してこの情報を正しいワークフローにルーティングするには、AWS Config ルールを使用することもできます。AWS Config は、スコープ内サービスへの変更を検出し (ただし、Amazon EventBridge よりもレイテンシーが大きい)、AWS Config ルールを使用して分析できるイベントを生成します。分析されたイベントは、ロールバックやコンプライアンスポリシーの適用のほか、変更管理プラットフォームや運用チケット発行システムなどのシステムへの情報の転送に使用できます。AWS Config イベントに対応する独自の Lambda 関数を作成するだけでなく、AWS Config ルール開発キットおよびオープンソースライブラリの AWS Config ルールも利用できます。

リソース

通知およびワークフローと監査コントロールとを統合するための、現在の AWS のベストプラクティスの詳細については、以下のリソースを参照してください。

動画

- Amazon Detective
- Amazon GuardDuty と AWS Security Hub の調査結果を修復する
- AWS でセキュリティオペレーションを管理するためのベストプラクティス
- AWS Config を使用して継続的なコンプライアンスを達成する

ドキュメント

- Amazon Detective
- Amazon EventBridge
- AWS Config ルール
- AWS Config ルールリポジトリ (オープンソース)
- AWS Config ルール開発キット

ハンズオン

- ソリューション: AWS Account Activity を用いたリアルタイムの洞察
- ソリューション: 集中ロギング

インフラストラクチャの保護

インフラストラクチャ保護には、ベストプラクティスと組織の義務または規制上の義務に準拠するために必要な、多層防御などの制御手法が含まれています。クラウドでの継続的な運用を成功させるには、このような手法を使用することが非常に重要です。

インフラストラクチャ保護は、情報セキュリティプログラムの重要な部分です。意図しない不正アクセスや潜在的な脆弱性から、ワークロード内のシステムとサービスを保護できます。たとえば、信頼境界 (ネットワークとアカウントの境界など)、システムセキュリティの設定とメンテナンス (強化、最小化、パッチ適用など)、オペレーティングシステムの認証と承認 (ユーザー、キー、アクセスレベルなど)、その他の適切なポリシー適用ポイント (ウェブアプリケーションファイアウォールや API ゲートウェイなど) を定義します。

AWS では、いくつかの方法でインフラストラクチャを保護できます。以下の各セクションでは、こうしたアプローチの使用方法を説明します。

- ネットワークの保護
- コンピューティングの保護

ネットワークの保護

ネットワーク設計を慎重に計画し管理することで、ワークロード内のリソースを分離し境界を作るための基礎が形成されます。ワークロードのリソースの多くは VPC 内で動作し、セキュリティのプロパティを継承するため、自動化によって支えられた検査および保護メカニズムで設計をサポートすることが重要です。同様に、純粋なエッジサービスやサーバーレスを使用して VPC の外部で動作するワークロードの場合は、よりシンプルなアプローチでベストプラクティスを適用します。サーバーレスセキュリティに関する具体的なガイダンスについては、「AWS Well-Architected サーバーレスアプリケーションレンズ」を参照してください。

ネットワークレイヤーを作成する: 共通の達成可能要件を持つ EC2 インスタンス、RDS データベースクラスター、Lambda 関数などのコンポーネントは、サブネットで形成されるレイヤーにセグメント化できます。たとえば、インターネットアクセスを必要としない VPC 内の RDS データベースクラスターは、インターネットへのルート、またはインターネットからのルートがないサブネットに配置する必要があります。このコントロールに対する階層的なアプローチは、意図しないアクセスを許可する可能性がある単一レイヤーの誤設定の影響を軽減します。AWS Lambda の場合は、VPC 内で関数を実行して、VPC ベースのコントロールを進めることができます。

数千の VPC、AWS アカウント、オンプレミスネットワークを含むネットワーク接続の場合は、AWS Transit Gateway を使用する必要があります。AWS Transit Gateway は、スポークのように機能するすべての接続されたネットワーク間でトラフィックがどのようにルーティングされるかを制御するハブとして機能します。Amazon VPC と AWS Transit Gateway の間のトラフィックは、AWS プライベートネットワーク上にとどまります。これにより、分散型サービス妨害 (DDoS) 攻撃や一般的な脆弱性攻撃 (SQL インジェクション、クロスサイトスクリプティング、クロスサイトリクエストフォージェリ、破損した認証コードの不正使用など) といった外部からの脅威ベクトルが軽減されます。AWS Transit Gateway のリージョン間ピアリングはまた、リージョン間トラフィックを単一障害点や帯域幅のボトルネックなしで暗号化します。

すべてのレイヤーでトラフィックを制御する: ネットワークトポロジを設計する際には、各コンポーネントの接続要件を調べる必要があります。たとえば、コンポーネントがインターネットアクセス (インバウンドおよびアウトバウンド) や、VPC、エッジサービス、外部データセンターへの接続を必要とする場合です。

VPC では、設定したプライベート IPv4 アドレス範囲または AWS によって選択された IPv6 アドレス範囲を使用して、AWS リージョンにまたがるネットワークトポロジを定義できます。インバウンドトラフィックとアウトバウンドトラフィックの両方に、多層防御アプローチを用いた複数のコントロールを適用する必要があります。これには、セキュリティグループ (ステートフルインスペクションファイアウォール)、ネットワーク ACL、サブネット、ルートテーブルの使用などが含まれます。VPC 内では、アベイラビリティーゾーンにサブネットを作成できます。各サブネットには、トラフィックがサブネット内でたどるパスを管理するためのルーティングルールを定義するルートテーブルを関連付けることができます。インターネットまたは VPC にアタッチされた NAT あるいは他の VPC ゲートウェイを経由するルートを設定することで、インターネットルーティングが可能なサブネットを定義できます。

インスタンス、RDS データベース、またはその他のサービスが VPC 内で起動されると、ネットワークインターフェイスごとに独自のセキュリティグループが設定されます。このファイアウォールはオペレーティングシステムレイヤーの外側にあり、許可されるインバウンドトラフィックとアウトバウンドトラフィックのルールを定義するために使用できます。また、セキュリティグループ間の関係も定義できます。たとえば、データベース層のセキュリティグループ

内のインスタンスは、関連するインスタンスに適用されるセキュリティグループを参照して、アプリケーション層のインスタンスからのトラフィックのみを受け入れます。TCP 以外のプロトコルを使用している場合を除き、ロードバランサーや CloudFront なしでインターネットから EC2 インスタンスに直接アクセスできるようにする必要はありません (セキュリティグループによって制限されているポートでも)。 これにより、オペレーティングシステムやアプリケーションの問題による意図しないアクセスから保護できます。サブネットには、ステートレスファイアウォールとして機能する、サブネットにアタッチされたネットワーク ACL を設定することもできます。レイヤー間で許可されるトラフィックの範囲を絞り込むようにネットワーク ACL を設定する必要があります。インバウンドルールとアウトバウンドルールの両方を定義する必要があることに注意してください。

一部の AWS サービスは、インターネットにアクセスして API 呼び出しをする (AWS API のエンドポイントがある場合) コンポーネントを必要としますが、その他の AWS サービスでは VPC 内のエンドポイントを使用します。Amazon S3 や DynamoDB を含む多くの AWS サービスは VPC エンドポイントをサポートしており、このテクノロジーは AWS PrivateLink で一般化されています。インターネットへのアウトバウンド接続を必要とする VPC アセットでは、これらは、AWS が管理する NAT ゲートウェイ、アウトバウンド専用のインターネットゲートウェイ、ユーザーが作成して管理するウェブプロキシを経由するアウトバウンド (一方向) でのみ可能です。

検査と保護を実装する: 各レイヤーでトラフィックを検査およびフィルタリングします。HTTP ベースのプロトコルを介してトランザクションを実行するコンポーネントの場合、一般的な攻撃からの保護にはウェブアプリケーションファイアウォールが役立ちます。AWS WAF は、Amazon API Gateway API、Amazon CloudFront、または Application Load Balancer に転送される設定可能なルールに一致する HTTP リクエストを監視してブロックできるウェブアプリケーションファイアウォールです。AWS WAF の使用を開始するには、AWS で管理されるルールと独自のルールと組み合わせて使用するか、既存のパートナー統合を使用できます。

AWS Organizations 全体にわたって AWS WAF、AWS Shield Advanced による保護、Amazon VPC セキュリティグループを管理するには、AWS Firewall Manager を使用できます。AWS Firewall Manager を使用すると、アカウントとアプリケーション全体にわたってファイア

ウォールルールを一元的に設定および管理できるため、一般的なルールの適用を簡単に拡張できます。また、AWS Shield Advanced やウェブアプリケーションへの不要なリクエストを自動的にブロックするソリューションを使用して、攻撃に迅速に対応できます。

ネットワーク保護を自動化する: 保護メカニズムを自動化し、脅威インテリジェンスと異常検出に基づく自己防御型ネットワークを提供します。たとえば、現在の脅威に適応し、その影響を軽減できる侵入検知および防止ツールなどです。ウェブアプリケーションファイアウォールは、ネットワーク保護を自動化できる 1 つの例です。たとえば、AWS WAF セキュリティオートメーションソリューション (https://github.com/awslabs/aws-waf-security-automations) を使用して、既知の脅威アクターに関連付けられた IP アドレスからのリクエストを自動的にブロックします。

リソース

ネットワークの保護に関する AWS のベストプラクティスの詳細については、以下のリソースを参照してください。

動画

- 多くの VPC 用の AWS Transit Gateway リファレンスアーキテクチャ
- Amazon CloudFront、AWS WAF、AWS Shield によるアプリケーションの高速化と保護
- 大規模な DDoS 攻撃の検出

ドキュメント

- Amazon VPC ドキュメント
- AWS WAF の開始方法
- ネットワークアクセスコントロールリスト
- VPC 用のセキュリティグループ
- VPC の推奨ネットワーク ACL ルール

- AWS Firewall Manager
- AWS PrivateLink
- VPC エンドポイント
- Amazon Inspector

ハンズオン

- ラボ: VPC の自動デプロイ
- ラボ: ウェブアプリケーションファイアウォールの自動デプロイ

コンピューティングの保護

脆弱性管理を実行する: コード、依存関係、インフラストラクチャ内の脆弱性のスキャンとパッチ適用を頻繁に実施し、新たな脅威から保護します。

ビルドとデプロイのパイプラインを使用すると、脆弱性管理の多くの部分を自動化できます。

- サードパーティー製の静的コード分析ツールを使用して、チェックされていない関数入力境界やより最近の CVE などの一般的なセキュリティ問題を特定できます。サポートされている言語に Amazon CodeGuru を使用できます。
- サードパーティー製の依存関係チェックツールを使用して、コードがリンクしているライブラリが最新バージョンであるかどうか、ライブラリ自体に CVE が含まれていないかどうか、ライブラリにソフトウェアポリシー要件を満たすライセンス条件があるかどうかを判断します。

- Amazon Inspector を使用すると、インスタンスに対する設定評価を実行して既知の共通脆弱性識別子 (CVE) を確認したり、セキュリティベンチマークに対して評価したり、欠陥の通知を完全に自動化したりすることができます。Amazon Inspector は本番環境インスタンス上またはビルドパイプライン上で実行され、調査結果があると開発者とエンジニアに通知します。調査結果にはプログラムを使用してアクセスし、バックログやバグ追跡システムにチームを誘導することができます。EC2 Image Builder は、自動パッチ適用、AWS が提供するセキュリティポリシーの適用、その他のカスタマイズにより、サーバーイメージ (AMI) を保持するために使用できます。
- コンテナを使用する場合は、ビルドパイプラインの ECR イメージスキャンをイメージリポジトリに対して定期的に実行し、コンテナ内の CVE を探します。
- Amazon Inspector やその他のツールは、設定や CVE の有無を特定するのには効果的ですが、アプリケーションレベルでワークロードをテストするには他の方法が必要になります。ファジングは、オートメーションを使用して不正な形式のデータを入力フィールドやアプリケーションの他の領域に挿入するバグを見つけるためのよく知られた手法です。

これらの機能の多くは、AWS のサービス、AWS Marketplace の製品、またはオープンソースツールを使用して実行できます。

攻撃対象領域を縮小する: オペレーティングシステムを強化し、使用するコンポーネント、ライブラリ、外部から利用可能なサービスを最小限に抑えることで、攻撃対象領域を縮小します。攻撃対象領域を縮小するには、発生する可能性があるエントリポイントと潜在的な脅威を特定する脅威モデルが必要です。攻撃対象領域を縮小する一般的な方法では、まずオペレーティングシステムパッケージやアプリケーション (EC2 ベースのワークロード)、あるいはコード内の外部ソフトウェアモジュールなどの、未使用のコンポーネント (すべてのワークロード) を減らします。一般的なオペレーティングシステムやサーバーソフトウェア向けの強化およびセキュリティ設定ガイドが数多くあります。たとえば、Center for Internet Security に用意されているガイドを出発点として使用し、その後も繰り返し使用することができます。

ユーザーがリモートからアクションを実行できるようにする: インタラクティブアクセスの機能を排除すると、人為的ミスのリスクが軽減され、手動での設定や管理が行われる可能性が低

くなります。たとえば、直接アクセスや踏み台ホスト経由のアクセスを許可する代わりに、AWS Systems Manager などのツールを使用し、変更管理ワークフローで EC2 インスタンスを管理します。AWS Systems Manager では、自動化ワークフロー、ドキュメント (プレイブック)、実行コマンドなどの機能を使用して、さまざまなメンテナンスおよびデプロイタスクを自動化できます。AWS CloudFormation スタックは、パイプラインから構築され、AWS マネジメントコンソールや API を直接使用することなく、インフラストラクチャのデプロイおよび管理タスクを自動化できます。

マネージドサービスを実装する: Amazon RDS、AWS Lambda、Amazon ECS などのリソースを管理するサービスを実装し、責任共有モデルの一部としてのセキュリティメンテナンスタスクを削減します。たとえば、Amazon RDS は、リレーショナルデータベースのセットアップ、運用、スケーリングを支援し、ハードウェアのプロビジョニング、データベースのセットアップ、パッチ適用、バックアップなどの管理タスクを自動化します。つまり、「AWS Well-Architected フレームワーク」で説明されているその他の方法でアプリケーションを保護することに集中できる時間が増加します。AWS Lambda では、サーバーのプロビジョニングや管理を行わずにコードを実行できるため、インフラストラクチャやオペレーティングシステムではなく、コードレベルでの接続、呼び出し、セキュリティのみに集中できます。

ソフトウェアの整合性を検証する: ワークロードで使用されるソフトウェア、コード、ライブラリが信頼できるソースからのものであり、改ざんされていないことを検証するメカニズム (コード署名など) を実装します。たとえば、バイナリとスクリプトのコード署名証明書を検証して作成者を確認し、作成者が作成してから改ざんされていないことを確認する必要があります。さらに、ダウンロードするソフトウェアのチェックサムをプロバイダーからのチェックサムと比較し、改ざんされていないことを確認できます。

コンピューティング保護を自動化する: 脆弱性管理、攻撃対象領域削減、リソース管理などのコンピューティング保護メカニズムを自動化します。自動化により、ワークロードの他の側面の保護に時間を使えるようになり、人為的ミスを犯すリスクを軽減できます。

リソース

コンピューティングの保護に関する AWS のベストプラクティスの詳細については、以下のリソースを参照してください。

動画

- Amazon EC2 インスタンスメタデータサービスのセキュリティに関するベストプラクティス
- AWS でブロックストレージを保護する
- サーバーレスおよびコンテナサービスを保護する
- Amazon EKS で高セキュリティワークロードを実行する
- Amazon EKS で Policy Guardrails によるセキュリティ設計を行う

ドキュメント

- AWS Lambda のセキュリティの概要
- Amazon EC2におけるセキュリティ
- AWS Systems Manager
- Amazon Inspector
- 独自の AWS Systems Manager ドキュメントを作成する
- 要塞ホストを Amazon EC2 Systems Manager に置き換える

ハンズオン

- ラボ: EC2 ウェブアプリケーションの自動デプロイ

データ保護

ワークロードを設計する前に、セキュリティに影響を与える基本的なプラクティスを用意しておく必要があります。たとえば、データ分類は機密性のレベルに基づいてデータを分類する方法を提供し、暗号化では、不正なアクセスに対し、データを判読できなくすることでデータを保護します。これらは、規制義務への対応ミスを回避したり、規制義務を順守したりする目的の達成に役立つ重要な方法です。

AWS にはデータ保護対策として使用できるさまざまな方法が多数あります。以下のセクションでは、こうしたアプローチの使用方法を説明します。

- データ分類
- 保管中のデータを保護する
- 転送中のデータを保護する

データ分類

データ分類方法を確立すると、重要度と機密性に基づいて組織データをカテゴリ別に分類して、各カテゴリに適した保護と保持方法でデータを管理できるようになります。

ワークロード内のデータを特定する: ワークロードで処理しているデータの種類と分類、関連するビジネスプロセス、データ所有者、適用される法律・コンプライアンス上の要件、保存場所、結果として実行が必要な統制について理解する必要があります。これには、データが一般公開されることを意図しているかどうか、データが顧客個人識別情報 (PII) などの内部使用のみかどうか、データが知的財産である、法的な秘匿特権がある、機密性が高いと特記されているなど、より制限されたアクセス用であるかどうかを示す分類が含まれます。適切なデータ分類システムを慎重に管理し、各ワークロードの保護要件のレベルに合わせることで、データに適したコントロールとアクセスと保護のレベルをマッピングできます。たとえば、パブリックコンテンツは誰でもアクセスできますが、重要なコンテンツは暗号化され、コンテンツを復号するためのキーには承認アクセスを要求することで保護しながら保存されます。

データ保護コントロールを定義する: リソースタグ、機密性ごと (およびエンクレーブ、関心のあるコミュニティごと) の個別の AWS アカウント 、IAM ポリシー、Organizations SCP、AWS KMS、AWS CloudHSM を使用することで、暗号化によるデータ分類と保護のポリシーを定義および実装できます。たとえば、非常に重要なデータを含む S3 バケット、または、秘密データを処理する EC2 インスタンスを含むプロジェクトがある場合、それらに「Project=ABC」タグを付けることができます。直属のチームのみがこのプロジェクトコードの意味を知っていて、属性ベースのアクセス統制手段を使用する方法を提供します。キーポリシーと許可を使用して AWS KMS 暗号化キーへのアクセスレベルを定義し、安全なメカニズムを通じて適切なサービスだけが機密コンテンツにアクセスできるようにします。タグに基づいて承認決定を判断する場合、AWS Organizations 内のタグポリシーを使用して、タグの許可が適切に定義されていることを確認する必要があります。

データライフサイクル管理を定義する: ライフサイクル戦略は、機密性レベル、法的および組織の要件に基づいて定義される必要があります。データを保持する期間、データ破壊プロセス、データアクセス管理、データ変換、データ共有などの側面を考慮する必要があります。データ分類方法を選択するときは、可用性とアクセスのバランスを取ります。また、各レベルにとって安全でありながら使いやすい方式を採用するために、複数レベルのアクセスと微妙な差異も実装する必要がります。常に多層防御方式を採用し、データおよびデータの変換、削除、コピーのメカニズムに人間がアクセスする機会を減らします。たとえば、アプリケーション認証を厳格にし、「遠距離操作」を実行するために必要なアクセス許可をユーザーでなくアプリケーションに付与します。 さらに、ユーザーが信頼できるネットワークパスからアクセスしていることを確認して、復号鍵へのアクセスを要求します。ユーザーにデータへの直接アクセス権を付与するのではなく、ダッシュボードや自動レポートなどのツールを使用して、データからの情報をユーザーに提供します。

識別と分類を自動化する: データを自動的に識別・分類することで、適切な統制の実装をサポートできます。人が直接アクセスするよりも自動化した方が、人為的ミスや開示リスクは小さくなります。機械学習技術を活用して、AWS 内の機密情報を自動的に検出、分類、保護する Amazon Macie などのツールの使用を検討しましょう。Amazon Macie では、個人識別情報 (PII)

や知的財産などの機密データが認識されます。このサービスのダッシュボードやアラートを使用して、データのアクセスや移動の状況を確認できます。

リソース

データ分類の詳細については、以下のリソースを参照してください。

ドキュメント

- データ分類に関するホワイトペーパー
- Amazon EC2 リソースのタグ付け
- Amazon S3 オブジェクトのタグ付け

保管中のデータを保護する

保管中のデータとは、ワークロードの任意の期間に永続的ストレージに保持されるすべてのデータを指します。たとえば、ブロックストレージ、オブジェクトストレージ、データベース、アーカイブ、IoT デバイス、データが保持されているその他のストレージ媒体などがあります。暗号化と適切なアクセスコントロールが実装されている場合は、保管中のデータを保護することで不正アクセスのリスクを軽減できます。

暗号化とトークン分割は、いずれも重要なデータ保護スキームですが、特徴に違いがあります。

トークン分割とは、機密情報を表すトークン (顧客のクレジットカード番号を表すトークンなど) を定義するためのプロセスです。トークンはそれ自体に意味があってはいけません。また、トークン化中のデータから派生してはいけません。このため、暗号化ダイジェストはトークンとしては使用できません。トークン分割方式を慎重に計画することで、コンテンツの保護を強化し、コンプライアンス要件を確実に満たすことができます。たとえば、クレジットカード番号の代わりにトークンを利用すると、クレジットカード処理システムに関するコンプライアンスの範囲を狭めることができます。

暗号化とは、プレーンテキストに復号化するために必要な秘密鍵がないとコンテンツを読めないように変換する方法です。必要に応じてトークン分割と暗号化の両方を使用して、情報の安全を確保し、保護することができます。この他に、マスキング手段を使用すると、残りのデータが機密とみなされないポイントまでデータの一部を編集できます。たとえば、PCI-DSS では、カード番号の最後の4桁をコンプライアンススコープの境界外に保持して、インデックスを作成できます。

安全なキー管理を実装する: キーの保存、ローテーション、アクセス制御方法を定義することで、未認証のユーザーからコンテンツを保護したり、認証ユーザーへの不必要な開示からコンテンツを保護したりできます。AWS KMS を利用すると、暗号化キーの管理が簡単になり、多くの AWS のサービスと統合できます。このサービスでは、マスターキーのための、耐久性と安全性が高く、冗長なストレージを利用できます。キーのエイリアスのほか、キーレベルのポリシーも定義できます。ポリシーは、キー管理者やキーユーザーを定義するのに役立ちます。さらに、AWS CloudHSM はクラウドベースのハードウェアセキュリティモジュール (HSM) であり、AWS クラウド上で独自の暗号化キーを簡単に生成して使用できます。FIPS 140-2 レベル 3 検証済みの HSM を使用することで、データセキュリティに関する企業、契約、規制のコンプライアンス要件を満たすことができます。

保管中に暗号化を適用する: データを保存する際は暗号化の使用を徹底してください。AWS KMS は、さまざまな AWS のサービスとシームレスに統合するため、保管中のすべてのデータを簡単に暗号化できます。たとえば、Amazon S3 では、すべての新しいオブジェクトが自動的に暗号化されるように、バケットの暗号化をデフォルトで設定できます。さらに、Amazon EC2 では、リージョン全体のデフォルトの暗号化オプションを設定することで、暗号化の適用をサポートしています。

アクセスの制御を適用する: 保管中のデータ保護には、アクセス (最小権限を使用)、バックアップ (信頼性に関するホワイトペーパーを参照)、分離、バージョニングなどのさまざまな制御方法を利用できます。データへのアクセスは、CloudTrail などのこのホワイトペーパーで前述した探査メカニズムと、S3 アクセスログなどのサービスレベルログを使用して監査する必要があります。パブリックにアクセス可能なデータをインベントリし、時間の経過とともに利用可能なデータ量の削減を可能にする方法を計画する必要があります。Amazon S3 Glacier のボール

トロックと S3 オブジェクトロックは、必須のアクセス制御を提供する機能です。ボールトポリシーがコンプライアンスオプションを使用してロックされると、ロックの有効期限が切れるまではルートユーザーでも変更できません。このメカニズムは、SEC、CFTC、FINRA の帳簿および記録管理要件を満たしています。詳細については、こちらのホワイトペーパーを参照してください。

暗号化キーの使用を監査する: 暗号化キーの使用方法を理解したうえで、監査を実施し、キーのアクセス制御メカニズムが適切に実践されていることを検証します。たとえば、AWS KMS キーを使用する AWS のサービスでは、AWS CloudTrail でそれぞれの使用が記録されます。その後、Amazon CloudWatch Insights などのツールを使用して AWS CloudTrail にクエリし、すべてのキーの使用が有効であることを確認できます。

各種メカニズムを使用してデータから人を遠ざける: 通常の運用状況で、すべてのユーザーが機密データおよびシステムに直接アクセスできないようにします。たとえば、変更管理ワークフローを使用して、直接アクセスや踏み台ホストを許可する代わりに、ツールを使用して EC2 インスタンスを管理します。これは、タスクの実行に使用するステップを含む自動化ドキュメントを使用する AWS Systems Manager Automation を使用して実現できます。これらのドキュメントはソース管理に保存し、実行前にピアレビューを行い、シェルアクセスと比較してリスクを最小限に抑えるために徹底的にテストできます。ビジネスユーザーは、データストアに直接アクセスする代わりにダッシュボードを使用し、クエリを実行できます。CI/CD パイプラインを使用しない場合は、通常無効になっている特権アクセスメカニズムを適切に提供するために必要な制御とプロセスを決定します。

保管中のデータ保護を自動化する: 自動化ツールを使用して保管中のデータの制御を継続的に検証し、提供します。たとえば、すべてのストレージリソースが暗号化されていることを確認します。AWS Config ルールを使用して、EBS ボリュームがすべて暗号化されていることを自動検証できます。AWS Security Hub は、セキュリティ標準に対する自動チェック機能を通じて、さまざまな制御を検証することもできます。さらに、AWS Config ルールでは、準拠していないリソースを自動的に修復できます。

リソース

保管中のデータの保護に関する AWS のベストプラクティスの詳細については、以下のリソースを参照してください。

動画

- AWS での暗号化のしくみ
- AWS でブロックストレージを保護する
- AWS CloudHSM でセキュリティ目標を達成する
- AWS Key Management Service を実装するためのベストプラクティス
- AWS の暗号化サービスの詳細

ドキュメント

- 暗号化を使用し Amazon S3 のデータを保護する
- Amazon EBS 暗号化
- Amazon RDS リソースの暗号化
- 暗号化を使用しデータを保護する
- AWS のサービスでの AWS KMS の使用方法
- Amazon EBS 暗号化
- AWS Key Management Service (KMS)
- AWS CloudHSM
- AWS KMS 暗号化の詳細についてのホワイトペーパー
- AWS KMS でのキーポリシーの使用
- バケットポリシーとユーザーポリシーの使用
- AWS Crypto Tools ドキュメント

転送中のデータを保護する

転送中のデータとは、システム間で送信されるすべてのデータを指します。これには、ワークロード内のリソース間での通信や他のサービスとエンドユーザーとの通信が含まれます。転送中のデータに適切なレベルの保護を提供することにより、ワークロードのデータの機密性と整合性を守ることができます。

安全なキーと証明書の管理を実装する: 暗号化キーと証明書を安全に保存し、厳格なアクセスコントロールで適切な時間間隔でローテーションします。これを実現する最善の方法として、AWS Certificate Manager (ACM) などのマネージド型サービスを使用します。これにより、AWS のサービスおよび内部接続リソースで使用するためのパブリックおよびプライベートの Transport Layer Security (TLS) 証明書のプロビジョニング、管理、デプロイが容易になります。TLS 証明書は、ネットワーク通信を保護し、プライベートネットワーク上のリソースだけでなく、インターネットでウェブサイトの ID を確立するために使用されます。ACM は、Elastic Load Balancing、Amazon CloudFront のディストリビューション、API ゲートウェイの API などの AWS リソースと統合し、証明書の自動更新も処理します。ACM を使用してプライベートルート CA をデプロイする場合、証明書とプライベートキーの両方を EC2 インスタンス、コンテナなどで使用するために提供できます。

転送時の暗号化を適用する: 組織的、法的、コンプライアンスの要件を満たすために、適切な基準と推奨事項に基づいて定義された暗号化要件を適用します。AWS のサービスには、通信に TLS を使用し、AWS API との通信の際に伝送中データの暗号化を利用できる、HTTPS エンドポイントが用意されています。HTTP などの安全でないプロトコルは、セキュリティグループを使用して VPC で監査およびブロックできます。HTTP リクエストは Amazon CloudFront 内または Application Load Balancer の HTTPS に自動的にリダイレクトすることもできます。コンピューティングリソースを完全に制御して、サービス全体に伝送中データの暗号化を実装できます。また、外部ネットワークからお使いの VPC に VPN で接続して、トラフィックの暗号化を促進できます。特別な要件がある場合は、AWS Marketplace からサードパーティーのソリューションを入手できます。

ネットワーク通信を認証する: 認証をサポートするネットワークプロトコルを使用すると、当事者間で信頼を確立できます。これにより、プロトコルで使用される暗号化が追加され、通信が変更または傍受されるリスクが軽減します。認証を実装する一般的なプロトコルには、多くの AWS のサービスで使用される Transport Layer Security (TLS) と、AWS Virtual Private Network (AWS VPN) で使用される IPsecがあります。

意図しないデータアクセスを自動検知する: Amazon GuardDuty などのツールを使用して、データ分類レベルに基づいて定義された境界の外部にデータを移動する攻撃 (DNS プロトコルを使用して不明または信頼されないネットワークにデータをコピーするトロイの馬など) を自動検知します。Amazon GuardDuty に加えて、ネットワークトラフィック情報をキャプチャする Amazon VPC フローログを Amazon EventBridge とともに使用して、異常な接続 (成功と拒否の両方) の検出をトリガーできます。S3 の Access Analyzer は S3 バケット内で誰がどのデータにアクセス可能かを評価するのに役立ちます。

リソース

転送中データの保護に関する AWS のベストプラクティスの詳細については、以下のリソースを参照してください。

動画

- AWS Certificate Managerを使用してウェブサイトの証明書をELBに追加する方法
- AWS Certificate Manager プライベート CA の詳細

ドキュメント

- AWS Certificate Manager
- Application Load Balancer の HTTPS リスナー
- AWS VPN
- API ゲートウェイのエッジ最適化

インシデント対応

非常に成熟した予防的、発見的統制が実装されていてもなお、組織はセキュリティインシデントの潜在的な影響に対応し、影響を緩和するメカニズムを実装する必要があります。準備することで、インシデントの際にチームが効果的に動作し、問題を切り分けて封じ込め、運用を既知の正常な状態に復元する能力に強く影響します。セキュリティインシデントが起こる前にツールとアクセス権を整備し、ゲームデー (実践訓練) を通じてインシデント対応を定期的に実施しておけば、ビジネスの中断を最小限に抑えながら復旧することができます。

クラウドレスポンスの設計目標

NIST SP 800-61 Computer Security Incident Handling Guide などで定義されている一般的な処理やメカニズムは依然として正しいものですが、クラウド環境におけるセキュリティインシデントへの対応に関連する、以下の具体的な設計目標を評価することをお勧めします。

- **対応目標を確立する:** ステークホルダー、法律顧問、組織のリーダーと協力してインシデント対応の目標を決定します。一般的な目標としては、問題の抑制と軽減、影響を受けるリソースの復旧、フォレンジックのためのデータの保持、帰属などがあります。
- **計画を文書化する:** インシデントへの対応、インシデント時のコミュニケーション、インシデントからの復旧に役立つ計画を作成します。
- **クラウドを使用して応答する:** イベントとデータが発生する応答パターンを実装します。
- **あるものと必要なものを把握する:** ログ、スナップショット、その他の証拠を、一元化されたセキュリティクラウドアカウントにコピーして保持します。管理ポリシーを適用するタグ、メタデータ、メカニズムを使用します。たとえば、Linux `dd` コマンドまたは Windows の同等コマンドの使用を選択して、調査目的でデータの完全なコピーを作成することもできます。

- **再デプロイメカニズムを使用する:** セキュリティの異常が設定ミスに起因する場合は、適切な設定でリソースを再デプロイして差異を取り除くだけで解決できる場合があります。可能であれば、レスポンスメカニズムを安全にして、未知の状態の環境でも複数回実行できるようにしてください。
- **可能な場合は自動化する:** 問題やインシデントが繰り返し発生することが確認された場合、プログラムによってトリアージを行い、一般的な状況に応答するメカニズムを構築します。ユニークで新しく、機密性の高いインシデントには、手動よる応答を使用します。
- **スケーラブルなソリューションを選択する:** クラウドコンピューティングに対する組織のアプローチのスケーラビリティに合わせて、検出と応答の間の時間を短縮するように努めます。
- **プロセスを学習、改善する:** プロセス、ツール、人材のギャップを見つけたら、それらを修正する計画を実装します。シミュレーションはギャップを見つけてプロセスを改善する安全な方法です。

AWS では、インシデント対応の処理に際して、いくつかのアプローチを使用できます。以下のセクションでは、こうしたアプローチの使用方法を説明します。

- **教育**。クラウドテクノロジーとその利用方法について、セキュリティオペレーションやインシデント応答の担当者を教育します。
- **準備**。クラウド上のインシデントを検知して応答できるように、インシデント応答チームを準備し、検知機能を有効にし、必要なツールやクラウドサービスへの適切なアクセスを確保します。さらに、信頼性の高い一貫した応答を保証するために、手動と自動の両方で必要なランブックを準備します。他のチームと協力して、予想される基本的なオペレーションを確立しておけば、その知識を使って通常のオペレーションからの逸脱を特定できます。
- **シミュレーション**。クラウド環境内で予想されるセキュリティイベントと予想外のセキュリティイベントの両方をシミュレーションして、準備の効果を把握します。

- **イテレーション**。シミュレーションの結果を反復することによって、反応姿勢の規模を改善し、価値を生み出すまでの時間を短縮し、リスクをさらに軽減できます。

教育

自動化されたプロセスにより、組織はワークロードのセキュリティを向上させるための対策に集中して時間を費やすことができます。自動化されたインシデント対応により、イベントの関連付け、シミュレーションの実践、新しい応答手順の考案、調査の実施、新しいスキルの開発、新しいツールのテストや構築に時間を使えるようになります。自動化が進んだとはいえ、セキュリティ組織内のチーム、スペシャリスト、応答者には継続的な教育が必要です。

成功するためには、一般的なクラウドの経験を超えて、人材に大幅な投資をする必要があります。プログラミングスキル、開発プロセス (バージョン管理システムやデプロイメントの実践など)、インフラストラクチャの自動化を学ぶための追加トレーニングをスタッフに提供することで、組織はメリットを得ることができます。学習には、インシデント対応のゲームデーを通した実践的な訓練が最適です。こうすることで、チームの専門家は、他の人に教えながらツールを改善し、テクニックを磨くことができます。

準備

インシデント発生時には、インシデント応答チームはインシデントに関わるさまざまなツールやワークロードリソースにアクセスする必要があります。イベントが発生する前に、チームに業務を遂行するために事前にプロビジョニングされた適切なアクセス権があることを確認します。すべてのツール、アクセス、計画は、イベントが発生する前に文書化され、タイムリーに応答できるようにしておく必要があります。

重要な人員と外部リソースを特定する: クラウド上でのインシデントレスポンスへのアプローチを他のチーム (顧問弁護士、経営陣、ビジネスステークホルダー、AWS サポートサービスなど) と連携して定義する場合、重要な人物、ステークホルダー、関連する担当者を特定する必要があります。依存性を減らし、応答時間を短縮するために、チームや専門のセキュリティチ

ーム、応答者が利用するサービスについて教育を受け、実践的な演習をする機会を持つようにしてください。

対応能力を強化するために、外部の専門知識と異なる視点を備えた外部の AWS セキュリティパートナーを探しておくことをお勧めします。信頼できるセキュリティパートナーは、馴染みのない潜在的なリスクや脅威を特定するのに役立ちます。

インシデント管理計画を作成する: インシデントへの応答、インシデント時の伝達、インシデントからの復旧に役立つ計画を作成します。たとえば、ワークロードと組織にとって起こる可能性が最も高いシナリオで、インシデント対応計画を作成してみましょう。内部および外部に伝達およびエスカレーションする方法を含めます。インシデント対応計画をプレイブックの形で作成します。ワークロードと組織にとって起こる可能性が最も高いシナリオから始めます。現在生成されているイベントから始めてもよいでしょう。どこから始めていいか分からない場合は、AWS Trusted Advisor や Amazon GuardDutyの検出結果を参照してください。 マークダウンなどのシンプルな形式を使用して簡単に保守できますが、他のドキュメントを参照しなくても実行できるように、重要なコマンドやコードスニペットが含まれています。

シンプルに開始して、繰り返します。セキュリティの専門家やパートナーと緊密に協力して、プロセスを確実に実行するために必要なタスクを特定します。実行するプロセスのマニュアルの説明を定義します。その後、プロセスをテストし、ランブックパターンを反復して、対応のコアロジックを改善します。例外の定義およびそれらのシナリオに代わる解決方法を決定します。たとえば、開発環境では、設定ミスのある Amazon EC2 インスタンスを終了することができます。本番環境で同じイベントが発生した場合、インスタンスを終了させるのではなく、インスタンスを停止して、重要なデータが失われないことおよび終了が許容されることをステークホルダーに確認します。内部および外部に伝達およびエスカレーションする方法を含めます。プロセスへの手作業での対応に慣れてきたら、自動化して解決までの時間を短縮します。

アクセスを事前プロビジョニングする: インシデント応答者が AWS やその他の関連システムに事前プロビジョニングされた正しいアクセス権を持っていることを確認しておき、調査から復旧までの時間を短縮します。該当する担当者へのアクセス方法をインシデントの発生中に確認するようなことがあれば、対応にかかる時間が遅くなり、アクセスが共有されている場合や、

切迫した状況で適切にプロビジョニングされていない場合は、他のセキュリティの脆弱性が発生することもあります。チームメンバーがどのレベルのアクセス権を必要とするか (たとえば、どのような行動を取る可能性が高いかなど) を把握し、事前にアクセス権をプロビジョニングしておく必要があります。セキュリティインシデントに応答するために特別に作成されたロールまたはユーザー形式のアクセス権には、十分なアクセス権を確保するために、多くの場合特権が付与されます。したがって、これらのユーザーアカウントの使用を制限し、日常の活動のために使用しないようにする必要があります。また、使用した場合はアラートを通知するようにします。

ツールを事前デプロイする: 復旧までの調査時間を短縮できるように、セキュリティ担当者は適切なツールを AWS に事前にデプロイしておきます。

セキュリティエンジニアリングとオペレーションの機能を自動化するために、AWS の包括的な API とツールセットを使用できます。ID 管理、ネットワークセキュリティ、データ保護、モニタリング機能を完全に自動化し、すでに導入されている一般的なソフトウェア開発方法を使用して提供できます。セキュリティオートメーションを構築すれば、担当者がセキュリティ上の位置づけを監視し、イベントに手動で応答する代わりに、システムが監視、レビューを行い応答を開始できます。

インシデント対応チームが同じ方法でアラートに対応し続けると、アラート疲れになるリスクがあります。時間の経過とともに、チームはアラートに対する感度が鈍くなり、通常の状況の処理で間違いを犯したり、異常なアラートを見逃したりする可能性があります。自動化を利用すれば、繰り返し発生する通常のアラートを処理する機能を使用してアラート疲れを回避し、機密性の高いインシデントや独自のインシデントの処理を人間に任せることができます。

プロセス内のステップをプログラムで自動化すれば、手動プロセスを改善できます。イベントに対する修復パターンを定義したら、そのパターンを実行可能なロジックに分解して、そのロジックを実行するコードを記述できます。その後、対応者は、そのコードを実行して問題を修正します。時間の経過とともに、より多くのステップを自動化し、最終的には一般的なインシデントのクラス全体を自動的に処理できるようになります。

EC2 インスタンスのオペレーティングシステム内で実行されるツールでは、AWS Systems Manager Run Command の使用を評価する必要があります。このコマンドを使うと、Amazon EC2 インスタンスのオペレーティングシステムにインストールしたエージェントを使用して、インスタンスをリモートで安全に管理できます。その場合、AWS Systems Manager Agent (SSM Agent) が必要です。これはデフォルトで多くの Amazon マシンイメージ (AMI) にインストールされています。ただし、一度インスタンスが侵害されると、そのインスタンス上で実行されているツールやエージェントからの応答は信頼できる応答とみなされません。

フォレンジック機能を準備する: 外部のスペシャリスト、ツール、オートメーションなど、適切なフォレンジック調査能力を特定し、準備します。インシデントレスポンスアクティビティの一部に、インシデントに関連するディスクイメージ、ファイルシステム、RAM ダンプ、その他のアーティファクトの分析が含まれる場合があります。使用可能なカスタマイズされたフォレンジックワークステーションを構築し、影響を受けたデータボリュームのコピーをマウントします。フォレンジック調査技術には専門的なトレーニングが必要なため、外部のスペシャリストとの連携が必要になる場合があります。

シミュレーション

ゲームデーを実施する: ゲームデーは、シミュレーションや演習とも呼ばれ、現実的なシナリオでインシデント管理計画や手順を練習するための体系的な機会を提供する内部イベントです。ゲームデーは基本的に、準備をすることで対応能力を反復的に高めていくものです。ゲームデーのアクティビティを実行すべき理由は、次のとおりです。

- 準備態勢を検証する
- 自信の向上 - シミュレーションやトレーニングスタッフから学ぶ
- コンプライアンスまたは契約上の義務に準拠する
- 認定のためのアーティファクトを生成する
- 俊敏性 - 段階的な改善

- 高速化とツールの改善
- コミュニケーションとエスカレーションを詳細化する
- まれで予期外の事態に備える

このような理由から、SIRS のアクティビティへの参加には、ストレスの多いイベントで組織の効果性を高めるという価値があります。現実的で有益な SIRS アクティビティを開発するのは簡単な作業ではありません。すでに把握しているイベントを処理する手順や自動化のテストには一定のメリットがありますが、予想外の事象に対して自身をテストして継続的に改善するクリエイティブな SIRS アクティビティへの参加も重要です。

イテレーション

封じ込めと復旧機能を自動化する: インシデントの封じ込めと修復を自動化すれば、対応時間と組織への影響を低減できます。

プレイブックからプロセスやツールを作成して実践したら、ロジックをコードベースのソリューションに分解します。そうすることによって、多くの応答者が応答を自動化し、応答者によるばらつきや推測作業を取り除くためのツールとして使用することができます。これにより、対応のライフサイクルを高速化できます。次の目標は、このコードを人間の対応者ではなく、アラートやイベント自体によって呼び出すことで完全な自動化を実現し、イベント駆動型の対応を有効にすることです。

イベント駆動型の対応システムでは、検出メカニズムによって対応メカニズムがトリガーされ、自動的にイベントが修正されます。イベント駆動型の応答機能を使用すれば、検出メカニズムから応答メカニズムが動作するまでの時間を短縮できます。このイベント駆動型アーキテクチャを作成するには、AWS Lambdaを使用できます。AWS Lambdaは、イベントに応答してコードを実行し、基盤となるコンピューティングリソースを自動的に管理するサーバーレスコンピューティングサービスです。たとえば、AWS CloudTrail サービスが有効な AWS アカウントがあるとします。AWS CloudTrail が (`cloudtrail:StopLogging` API 呼び出しを通じて) 無効になっている場合は、Amazon EventBridge を使用して特定の

`cloudtrail:StopLogging` イベントをモニタリングし、AWS Lambda 関数を起動して `cloudtrail:StartLogging` を呼び出してログを再開できます。

リソース

インシデントレスポンスに関する最新の AWS のベストプラクティスの詳細については、以下のリソースを参照してください。

動画

- AWS 環境のセキュリティインシデントの準備と対応
- インシデント対応とフォレンジックの自動化
- ランブック、インシデントレポート、インシデント対応の DIY ガイド

ドキュメント

- AWS インシデント対応ガイド
- AWS Step Functions
- Amazon EventBridge
- CloudEndure Disaster Recovery

ハンズオン

- ラボ : AWS コンソールと CLI を使用したインシデント対応
- ラボ : Jupyter を使用したインシデント対応プレイブック - AWS IAM
- ブログ : AWS Step Functions を使用してセキュリティインシデント対応を調整する

まとめ

セキュリティは、継続的な取り組みです。発生したインシデントは、アーキテクチャのセキュリティを向上させるための機会として扱う必要があります。強力な ID コントロール、セキュリティイベントへの対応の自動化、複数レベルでのインフラストラクチャの保護、暗号化による適切に分類されたデータの管理により、あらゆる組織に実装する必要がある多層防御が可能になります。このホワイトペーパーで説明したプログラム関数と AWS の機能やサービスがあれば、このような取り組みもより簡単に実現できます。

AWS は、ビジネス価値を実現しながら、情報、システム、アセットを保護するアーキテクチャの構築と運用を支援します。

寄稿者

本ドキュメントは、次の人物および組織が寄稿しました。

- Ben Potter、Well-Architected プリンシパルセキュリティリード、アマゾン ウェブ サービス
- Bill Shinn、シニアプリンシパル、CISO オフィス、アマゾン ウェブ サービス
- Brigid Johnson、シニアソフトウェア開発マネージャー、AWS Identity、アマゾン ウェブ サービス
- Byron Pogson、シニアソリューションアーキテクト、アマゾン ウェブ サービス
- Darran Boyd、プリンシパルセキュリティソリューションアーキテクト、金融サービス、アマゾン ウェブ サービス
- Dave Walker、プリンシパルスペシャリストソリューションアーキテクト、セキュリティとコンプライアンス、アマゾン ウェブ サービス
- Paul Hawkins、シニアセキュリティストラテジスト、アマゾン ウェブ サービス

- Sam Elmalak、シニアテクノロジーリーダー、アマゾン ウェブ サービス

その他の資料

追加情報については、次の資料を参照してください。

- AWS Well-Architected フレームワークホワイトペーパー

改訂履歴

日付	説明
2020 年 7 月	アカウント、ID、アクセス権限の管理に関するガイダンスを更新。
2020 年 4 月	すべての分野、新しいベストプラクティス、サービス、機能のアドバイスを拡張する更新。
2018 年 7 月	新しい AWS のサービスと機能を反映するための更新、および参照の更新。
2017 年 5 月	システムセキュリティの設定とメンテナンスのセクションを更新し、新しい AWS のサービスと機能を反映。
2016 年 11 月	初版発行