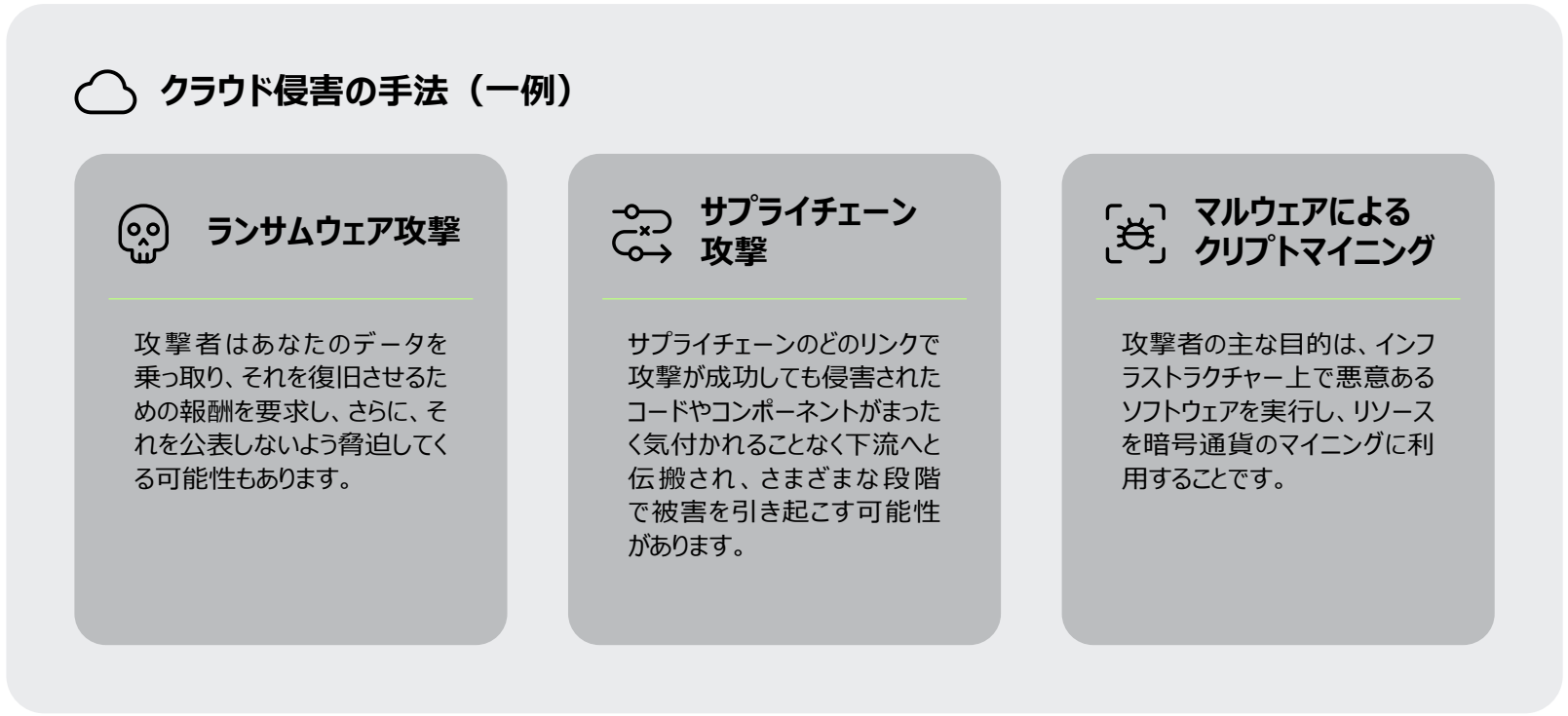
sysdig SECURE EVERY SECOND.

E-BOOK

クラウド攻撃のしくみを分析

目次



より多くのIT資産やインフラがクラウドに移行するにつれて、攻撃者はクラウドでの攻撃能力を高め、攻撃件数を増加させています。このような攻撃の目的は、オンプレミス環境の場合とほぼ同じですが、関与する戦術、手法、手順（TTP）は変化しています。クラウド環境で**これらの攻撃がどのように機能するか**を理解し、適切な防御策を採用できるようにすることが重要です。

このようなTTPの変化を実証するために、本文書では、攻撃的なオペレーションを分析し、それらがクラウド環境ではどのように見えるかを示します。本文書の目的は、クラウド攻撃のパターンに関する一般的な理解を提供することであり、そのような攻撃を行うためのステップバイステップのガイドを提供することではありません。

この文書では、次に示す一般的な3つのクラウド攻撃パターンを取り上げます。

- **ランサムウェア攻撃：**攻撃者はあなたのデータを乗っ取り、それを復旧させるための報酬を要求し、さらに、それを公表しないよう脅迫してくる可能性もあります。
- **サプライチェーンの侵害：**どのリンクで攻撃が成功しても、侵害されたコードやコンポーネントがまったく気付かれることなく下流へと伝搬され、さまざまな段階で被害を引き起こす可能性があります。
- **マルウェアによるクリプトマイニング：**攻撃者の主な目的は、インフラストラクチャー上で悪意あるソフトウェアを実行し、リソースを暗号通貨のマイニングに利用することです。

本記事全体を通して、クラウドに関する共通のテーマが見られるでしょう。なぜなら、クラウドに固有の複雑さが設定ミスの可能性を高めるからです。

第1章
ランサムウェア攻撃

オンプレミスの世界における典型的なランサムウェアのアプローチ：

ランサムウェア攻撃は、暗号技術を用いてデータを暗号化し、暗号を解除するための鍵を受け取りたいならば、一定額の金銭を（通常、追跡を避けるために暗号通貨で）支払うよう要求します。

典型的なシナリオは、添付ファイルの形で送られるフィッシングメールから始まります。この添付ファイルは、実際には脆弱性を悪用するようなマルウェアを含んでいるにもかかわらず、無害なファイルを擬装します。このようなマルウェアを、それと知らずに実行したユーザーが感染させ、企業全体に拡大させることになります。

攻撃者は、一定時間内に支払いが行われないならばデータを漏洩するとして、脅迫することがあります。このような脅迫には屈せず、バックアップを利用してデータを復元することが強く推奨されますが、それが常に可能であるとは限りません。最も有名なランサムウェア攻撃の1つとして、世界中の大企業が被害を受けたWannacryが挙げられます。

ランサムウェア攻撃は、通常、オンプレミス環境に関連していますが、それらは**クラウド環境でも確実に起こる可能性があります**。本文書では、コンピュートインスタンスやその他の「リフト＆シフト」タイプのリソースのようなオンプレミスの類似物ではなく、「クラウドネイティブ」な側面に焦点を当てます。S3やその他のクラウドネイティブなストレージサービスは、ランサムウェア攻撃に対して脆弱です。ここではS3を例に挙げますが、その他のクラウドストレージサービスも基本的には同じように脆弱です。

S3バケットが適切に許可され、保護されていれば、日和見的な攻撃者からの影響を軽減できます。しかし、S3バケットは多くのサービスに使用されており、権限の設定が不適切なこともしばしば起こるため、攻撃者は複数の異なる場所から、そのようなS3バケットを見つけて悪用する可能性があります。後述するように、その他のエントリポイントも、バケット内にあるデータのセキュリティに影響を与える可能性があります。

ステップ1：データへのアクセス権を獲得

S3バケットにアクセスするには、さまざまな方法があります。攻撃者にとって最も簡単な方法は、過剰に寛容なポリシーを持つパブリックバケットを標的にすることです。ただし、よりロックダウン型のポリシーであっても、ランサムウェア攻撃を受ける可能性があります。クラウドストレージサービス、それらの内部に含まれているデータオブジェクト、または関連するアイデンティティおよびアクセス管理（IAM）ポリシーなどに割り当てられた不適切な権限が、関連するセキュリティインシデントの根本原因であることがよくあります。このシナリオでは、攻撃者がEC2コンピュートインスタンスへのアクセス権を取得したものと仮定します。このEC2コンピュートインスタンスには、S3バケットへのアクセス権を提供するIAMポリシーが割り当てられているものとしま

AWSなどのクラウドサービスプロバイダー(CSP)は、通常はAPIによって動作するクラウドメタデータサービスを展開することで、クラウドリソースに関する情報提供や操作を可能にしています。攻撃者はコンピュートリソースへのアクセス権を取得すると、メタデータサービスのAPIをクエリできるようになります。そして攻撃者は、単純なHTTPリクエストを通じて、IAMポリシーのアクセスキーとシークレットキーを取得できます。攻撃者は、これらの情報を利用することで、そのコンピュートリソースで使用されているIAMアカウントに関連付けられているS3バケットにアクセスできるようになります。

ステップ2：ランサムウェアライクな攻撃のスクリプト化

上記のようなIAMポリシーが、S3バケットに「書き込み」権限を設定していると仮定します。この場合、攻撃者は、このバケット内にあるすべてのファイルを暗号化されたバージョンで上書きするような操作をスクリプト化できます。あるいは、攻撃者が特に破壊的な操作を行いたいと考えており、かつS3バケットに「削除」権限が割り当てられているならば、攻撃者はデータを窃取した後、そのS3バケット内にあるファイルを削除することもできます。

巧妙な攻撃者は、キー管理サービス（KMS）を利用して、データを暗号化し、データへのアクセスを効果的に制限する場合があります。さらに、攻撃者は、特定のソースIPアドレス（攻撃者が管理する独自アドレス）だけに変更を許可するといったポリシー条件を利用することで、他のいかなるKMSアクションのポリシーの変更も制限することさえ可能となります。KMSサービスと暗号化サービスは防御的なコントロールとして使用できますが、ランサムウェアが証明しているように、暗号化は防御に使用できると同時に、攻撃にも使用できるのです。

ステップ3：ランサムウェア攻撃を進める

クラウドストレージサービスは、バージョニングやKMSなど、この種の攻撃を一見難しくするようなオプションをいくつか備えています。しかし、これらのオプションは、私たちが想定しているような防弾保護を提供しない可能性があります。バージョニング機能は、S3バケットに保存されているオブジェクトの以前のコピーを保持し、復元できるようにするものです。これは、ランサムウェアのような何らかのデータの破損や喪失に見舞われた場合に、非常に役立つ機能です。

問題は、これらの古いバージョンが「s3:DeleteObjectVersion」のような簡単なAPIコールで削除できることです。このAPIコールを使うと、過去のバージョンをすべて削除できます。このような手法は、典型的なWindowsベースのランサムウェア攻撃と似ており、そのような攻撃で、攻撃者はすべてのシャドウボリュームインスタンスを削除しようとします。

KMSと暗号化サービスを利用することで、キーを持つユーザーだけがデータを復号化して閲覧することを可能にする、もう1つの保護レイヤーを提供できます。しかし、関連するIAMポリシーが寛容すぎる場合、攻撃者は、組織のデータを保護するために使用されているキーを削除するような、関連するKMS APIを呼び出せる可能性があります。CSPによっては、暗号化キーの消去がビジネスに悪影響を及ぼす可能性があるため、実際にこの措置が実施されるまでに数日かかる場合があります。また、CSPは安全対策としてこのような措置を遅らせる場合もあります。しかし、この攻撃手法が使用される可能性があることから、迅速な攻撃の検知と対応のためにクラウドログのモニタリングが重要であることが強調されます。キーが消去されると、バケットやバケット内のデータを復号することは誰にもできなくなり、事実上、データは失われたことになります。

まとめとベストプラクティス

ランサムウェア攻撃はクラウド環境で起こる可能性は依然としてありますが、それは典型的なオンプレミスでの攻撃とは見え方が異なります。攻撃者は、エンドポイントやローカルディレクトリサービス、IAMシステムを悪用する代わりに、CSPが提供する権限ポリシーや標準的なセキュリティ機能を利用します。ほとんどの組織の実装において、ポリシーは非常に複雑となります。これは、ポリシーが極めて柔軟であることによるものですが、アーキテクチャーの複雑さやクラウドリソースを操作または利用するユーザーの特性といった他の要因もあります。このような現実が、過剰な権限の付与やクラウドの設定ミスが多くの場合、クラウド環境におけるセキュリティインシデントの根本的な原因になっている理由です。

ここで紹介したランサムウェアのシナリオに対する答えとして、「最小権限の原則」を実施することが挙げられます。この原則に従うならば、ポリシーは「絶対に必要なユーザーやマシンのID」にのみ権限を付与することになります。これが解決策だと言うのは簡単ですが、適切な権限を長期にわたって実装し管理することはより困難です。ここで、**クラウドネイティブアプリケーション保護プラットフォーム（CNAPP）**ツールの出番となります。CNAPPツールを使うと、環境全体におけるポリシーと権限に関する可視性を確保した上で、問題となっているポリシーを修正できるようになります。

また、CSPは、データが変更されるのを防ぐための追加的な保護機能を提供する場合もあります。たとえば、S3のオブジェクトロック機能を使用すると、一定期間内にデータが変更または削除されるのを防止できます。この機能はリーガルホールドのような目的でよく使われますが、S3バケット内のデータが変更可能であることが正常なケースでは役に立たない可能性があります。Glacierのようなコールドストレージは、ランサムウェア攻撃の発生時にデータ損失を防ぐのに便利です。なぜなら、これにより、クリーンなバックアップからの復元が可能となるからです。

第2章

サプライチェーン攻撃

デジタルな**サプライチェーン攻撃**とは、脅威アクターが、顧客に提供されるリソースのハッキングを目的として、企業や組織が持つソフトウェア、システム、またはリソースを直接または間接的に改竄する攻撃です。間接的なデジタルサプライチェーン攻撃は、攻撃者がパートナーやサプライヤー、または被害組織の技術スタックの一部として使用されているコンポーネントを標的とするものです。多くの場合、マルウェアは、顧客やそのユーザーをハッキングするために、導入されたコードに追加されます。その不正なコードは、新しいインストールやパッチやホットフィックスを通じて配布されることがあります。改竄されたコードは、そのソフトウェアを使う顧客やユーザーに被害を与えるだけでなく、組織の評判をも傷付けます。また、注意すべき点として、「コード」はさまざまな形を取り、いったんそれが構築され提供されることでさまざまなアーティファクトとなる場合があることが挙げられます。これには、クラウド環境内にあるアプリケーション、インフラ、またはその他の特定のオブジェクトなどが含まれます。

顧客やパートナーとの信頼関係の喪失は、売上の減少、利用件数の低下、さらには競争力のある選択肢への乗り換えを引き起こす要因となり得ます。

最もよく知られている攻撃の1つとして、**SolarWinds社を標的にした攻撃が挙げられます。攻撃者はSolarWinds社のネットワークに侵入し、同社のビルドプロセスに悪意あるソフトウェアをインジェクトすること**に成功しました。このマルウェア（**APT29**、Nobeliumに関連するもの）は、Orion（ネットワーク管理システム）製品のアップデートの一部としてバンドルされていました。ビルドプロセスの一環として、この成果物はデジタル署名され、多くの顧客によってダウンロードされました。この悪意あるキャンペーンの最終的な標的は、SolarWinds社やその顧客だけでなく、多数の米国政府機関および民間組織でした。この攻撃により影響を受けた企業の総数は数万社に上りました。

デジタルサプライチェーンには、攻撃を受けやすい段階がいくつかあります。それらは、「設計」、「開発」、「配布/導入」、「保守」、そして最後に「廃棄」です。ここでは、継続的インテグレーション/継続的デリバリー（CI/CD）パイプラインのハッキングを通じて、「配布/導入」フェーズが攻撃される実例を紹介します。CI/CDパイプラインは、特に統合、テスト、導入の各フェーズにおいて、開発とデリバリーのプロセスを改善するために自動化と監視を導入するも

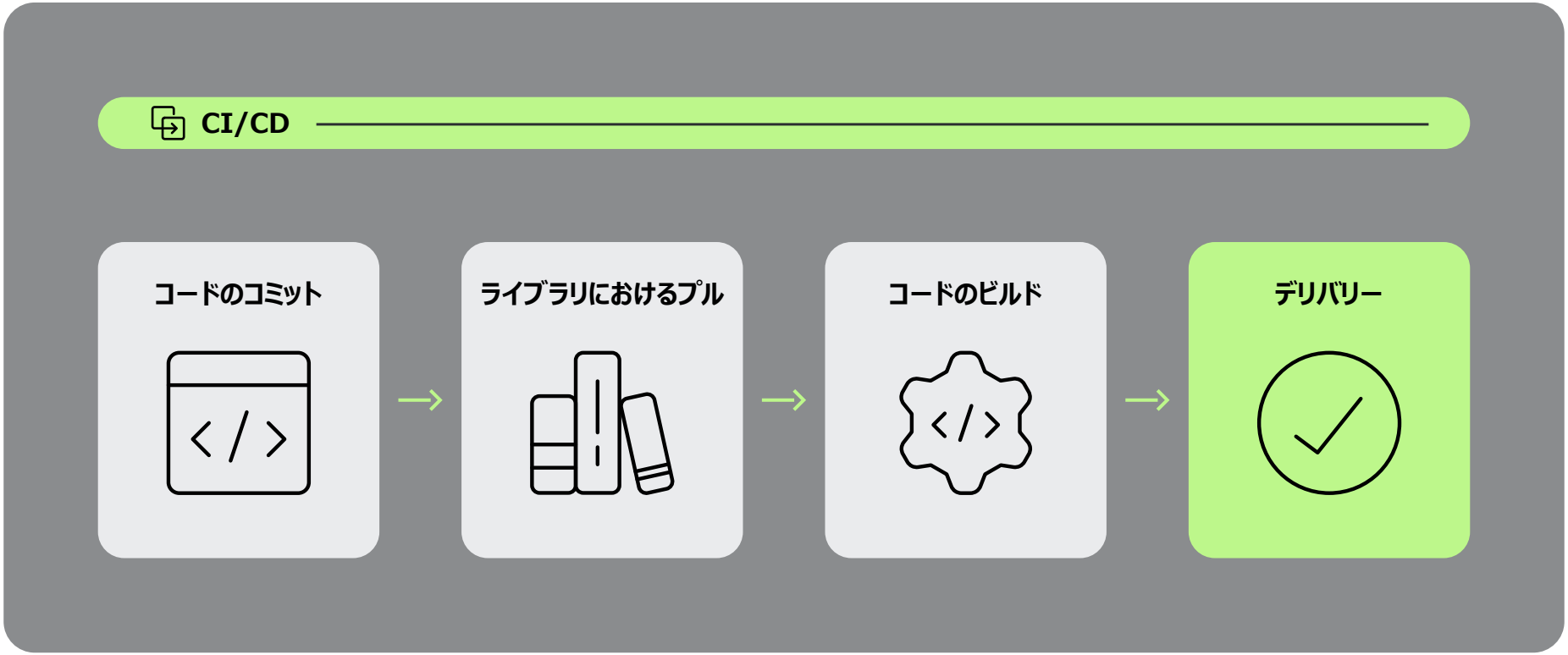
CI/CDパイプラインとそのパイプラインを動かすシステムは、保存されているクレデンシャルの量と、自動化タスクを実行するために付与されるアクセス権が理由で、脅威アクターにとってお気に入りの標的となっています。

ステップ1：CI/CDパイプラインの内部に足掛かりを得る

まず、攻撃者は有効なアカウントを通じて環境内に足掛かりを得る必要があります。これは、標的となる環境のセキュリティポスチャーに応じて、いくつかの方法で達成されます。例としては、ソースコードから漏れたAPIキーとパスワードを見つけること、スピアフィッシング攻撃の利用、リモート脆弱性攻撃の影響を受けやすいサービスを持つシステムを侵害することなどが挙げられます。ある事例では、認証プラグイン間における設定の衝突が理由で、誰でも自らを登録することにより、セキュリティが強化されているはずのCI/CDシステムやサービスにアクセスできるようになっていました。

また、最初にソフトウェア依存関係のリポジトリをハッキングすることで、攻撃することも可能です。コードが外部ソースの依存関係を参照する際、そのようなインポートにより不正なコードがロードされることがないと信頼しています。脅威アクターは、ソフトウェア依存関係のコードリポジトリをハッキングすることで、それに不正なコードをインジェクトできます。そのような悪意あるコードを使用することで、企業や組織の環境情報を漏洩させ、ソフトウェアユーザーを攻撃できます。参照される依存関係には、それ自身の依存関係、つまり推移的依存関係も定期的に含まれるため、問題はさらに深刻になります。

別の事例では、**開発サービスプロバイダーが未知の脅威アクターによりハッキングを受け**ました。このハッキングにより、脅威アクターは数千件の組織のコードリポジトリにアクセスできるようになりました。この侵害は、企業や組織が自らのコードリポジトリに不正な変更がないか継続的に監視し、CI/CDパイプラインからのアラートに対応することの必要性を浮き彫りにしました。



ステップ2：アクセス権の昇格

脅威アクターは運が良ければ、1件の侵入先のアカウントを通じて、自分が必要とする権限をすべて手に入れることができる場合があります。たとえば、保護されたリポジトリブランチにコード変更をプッシュする権限を持つユーザーは、格好の標的となります。また、GitLab RunnersのようなCI/CDサービスをサポートするプログラムも、過剰な権限で設定されていれば標的になり得ます。しかし、そのような幸運に恵まれない場合、脅威アクターは内部からクレデンシャルを見つけることで、より高度な権限を持つアカウントへと移動する必要があります。

Dockerコンテナは、Dockerが提供しているセキュリティ隔離を無効にする「--privileged」フラグを使用している場合、安全でないコンテナを実行しているホストを攻撃するために悪用される可能性があります。ネストされたコンテナを実行するか、または脆弱性を突いてコンテナを脱出することで、脅威アクターは当該ホスト上でスーパーユーザー権限を獲得できます。いったんホストへのアクセスが確立されると、脅威アクターは「ランド＆エキスパンド」戦略を実行することで、保存された認証情報を含む可能性のある設定ファイルやスクリプトを含むその他のリソースを検索できるようになります。

脅威アクターが CI/CDジョブを実行できるのであれば、新しいジョブを作成するか、または既存のジョブを変更することで、任意のあるいは悪意あるカスタムコードを実行できる可能性があります。環境変数はしばしばCI/CDタスクのクレデンシャルを安全に保存するために使用されます。そのようなカスタムコードは、環境変数に保持されたクレデンシャルをダンプするために使われるか、またはエラーレポートの中からクレデンシャルを漏洩させるクラッシュを引き起こすことさえあります。

ステップ3：悪意あるコードをインジェクトする

いったん脅威アクターが必要なアクセス権をすべて入手した場合、いくつかの方法で不正なコードを導入できるようになります。メインのリポジトリにコードをコミットし、CI/CDサービスに変更を導入させることもできます。しかし、それでは目立ちすぎ、疑念を持たれる可能性があります。そこで、脅威アクターは代わりにCI/CDパイプラインの定義を変更し、ビルドプロセスの一部として不正なコードを挿入しようとします。これにより、リポジトリは改竄されることなく、不正なコードが、ユーザー向けの最終的なプロダクションデリバリーにまで入り込むことになります。この結果、ハッキングされたリリースを使用するすべてのユーザーは、さらなる攻撃を受けやすくなります。

取るべき対策

デジタルサプライチェーンとCI/CDのパイプラインは、複雑なプロセスと環境を生み出し、その結果、重大なアタックサーフェスがもたらされます。この複雑さゆえに、安全性を確保するためには、理路整然とした包括的なレビューが必要となります。開発チームは、デジタルサプライチェーン全体を独自に保護するための専門知識が不足している可能性があります。また、すべてのパートナーやサプライヤーを考慮した場合、すべての要素を管理することさえできない可能性があります。

クラウド環境と開発環境は、強固なアクセス制御ポリシーを通じて保護する必要があります。そのようなポリシーは、最小権限の原則に基づいてアクセスを制限し、認証キーとパスワードのローテーションを定期的に実施し、**その他の認証要素**（2FAチャレンジや地理ベースのアプリ認証サービスなど）を要求するものでなければなりません。コードリポジトリを保護するには、CI/CDプロセスにチェックを追加し、デジタル署名とハッシュの検証を通じて、ビルドの完全性と真正性を検証する必要があります。コードとリリースは、セキュリティ上の欠陥や異常な動作がないかを調べるために、定期的にテストされなければなりません。また、システムは、日常的なセキュリティアップデートを通じて保守する必要があります。クラウド環境、CI/CD、コードリポジトリは、異常な動作がないかを調べるために、オペレーションセンターや対応チームが監視する必要があります。

第3章

マルウェアによるクリプトマイニング

現在、マルウェアの主な目的の1つとして、金銭的な利益を得ることが挙げられます。クラウド環境は、すでに、ソフトウエアベースの暗号通貨マイナーにとって絶好の温床となっていました。これは、すべてのユーザーが利用可能な、無限で、かつ弾力的なコンピューティングリソースのおかげです。暗号通貨の普及と価格の上昇に伴い、この手法は攻撃者にとってより有利なものとなっています。攻撃者は、企業や組織の持つクラウドコンピューティング能力を無断で利用することで、クラウド関連の費用を一切かけずに暗号通貨マイナーを実行できるのです。

ご存知のように、暗号通貨マイナーは、被害者が持つコンピューティングパワーを利用して、非管理型ウォレット上で暗号通貨を採掘します。この傾向は年々強まっており、コンテナベースのクラウドリソースを標的とする動きが広まっています。犯罪者はますますクラウドやコンテナ化されたリソースを悪用しており、多数のインスタンスを通じて、暗号通貨マイニングを実行する独自のコンテナを目立たなくしています。

> 暗号通貨の普及と価格の上昇に伴い、攻撃者にとってクライプトマイニングはより儲かるものになっています。

ステップ1：公共向けのワークロードを悪用する

公共向けのワークロードにアクセスすることは、さまざまなセキュリティメカニズムが理由で、達成するのが難しく思えるかもしれません。しかし、パッチの適用されていないサービス、設定ミス、または不適切な権限が原因で、公共向けのアプリケーションが悪用されるケースを、私たちは過去に何度も目にしてきました。たとえば、次に示すマルウェアファミリーは、いったんアクセス権を得た時点で、マイニングマルウェアの導入を自動化する方法を実装しています。

- **Sysrv Helloボットネット**
- **Muhstikボットネット**
- **RinBot**

Webアプリケーションの脆弱性を悪用するか（リモートコード実行やSQLインジェクションなど）、またはOSレベルのインフラの脆弱性を悪用することで、攻撃者はアクセス権を獲得し、権限を昇格させ、ワークロードを完全に制御できるようになります。

ここ数年、コンテナやコンテナプラットフォームの導入が進むにつれて、より多くのコンテナの導入がセキュアに実施されず、その結果、攻撃者により利用されているのを目にするようになりました。

公共向けのワークロード、アプリケーション、およびサービスを悪用することは、アクセス権を得るために最も頻繁に使用される方法の1つです。その後、攻撃者は、クラウド環境内に最初の足掛かりを確保し、自らの利益のために被害者のリソースを使用し始めます。

ステップ2：クラウドテナント内でのピボッティング

攻撃者が侵入先のホスト上で制御権を獲得した時点で、攻撃者が最終的なゴールに達したと考えるべきではありません。もちろん、この時点で、攻撃者は、被害者のリソースを使って暗号通貨を採掘することや、ホスト内の権限を昇格させることなど、自分の好きなようにインスタンスを利用できます。しかし、賢い攻撃者は、これがより複雑な攻撃チェーンの一部として、より大規模な攻撃を開始する起点になり得ることを知っています。

最新のインフラはほとんどがクラウドでホスティングされているため、攻撃者は、自らの目的を達成するために他のクラウドリソースを使用するチャンスがあり、1つのコンテナインスタンスや1つのK8sポッドだけをハッキングするのに比べて、より高いゴールを目指すことができます。クラウド環境では、インスタンスで利用可能な内部サービスがあるため、そのようなサービスを使ってさらに情報を引き出すことができる可能性があります。

よく使われる例として、クラウドテナント内でアクセス可能なインスタンスメタデータサービスがあります。これを使うことで、ユーザーはテナント内のリソースに関する情報を取得できます。しかし、この情報は攻撃者が自分たちの利益のために利用することもできます。この場合、インスタンスの詳細と共に、インスタンスにアタッチされたロールを取得することも可能です。そして、さらに重要なことは、これらのロールに関連する一時的なクレデンシャルを引き出せることです。

攻撃者は、これらの一時的なクレデンシャルを所有した時点で、クラウド環境に直接アクセスし、クラウドメタデータAPIを使用してさらに多くの情報を取得し、攻撃を続行するための新しい攻撃ベクトルを見つけることが可能となります。

ステップ3：最終ジャックポット

攻撃者はクラウド環境にログインした後、インスタンスから取得した権限の評価を開始します。これにより、攻撃者は、クラウドアカウントへの変更か、または新しいリソースの生成を許可するような権限が特定のサービスにあるかどうかをチェックできるようになります。

ご存知のように、クラウド環境で権限の設定ミスを見つけるのはそれほど難しいことではありません。クラウド環境では権限がきめ細かく設定できるため、**最小権限の原則**を適用することは難しく、運用を維持することも困難です。しかし、たった1つ権限の設定を間違えただけで、攻撃者は環境内の権限を昇格させることが可能となることに注意する必要があります。その結果、攻撃者は、当該テナント内にあるリソースを使用して目的を達成できるようになるからです。

このように、攻撃者は、寛容なAssumeRole設定のような設定ミスを見つけることができれば、コンピュートサービスやマネージド型のコンテナサービスを利用して、暗号通貨のマイニングを始めることができる可能性があります。侵入された企業や組織は、攻撃者の悪意ある金目当ての活動に対して巨額の費用を支払うことになります。

アドバイスと予防

これまで見てきた攻撃経路では、この種の悪用を防ぐために、さまざまな軽減策が用意されています。

まず明らかなのは、よく知られた脆弱性、特に公共向けアプリケーションの脆弱性にパッチを当てることです。ご存知のように、これらのアプリケーションは攻撃者に最も狙われやすく、クラウドインフラストラクチャーの中で最初に安全を確保しなければならない部分です。サービスやアプリケーション、特にインターネット経由でアクセス可能なアプリケーションにパッチを当て保守することは、この種の攻撃を避けるために不可欠です。

もう1つの推奨される緩和策としては、インスタンスメタデータサービスへのアクセスを制限することが挙げられます。先述したように、インスタンスメタデータサービスは、攻撃者に悪用される可能性のある有用な情報が豊富に保存されている場所だからです。

まとめ

クラウドはオンプレミスより安全なのか？

攻撃者の動機と目標は、私たちが使用するテクノロジーが大きく変化したにもかかわらず、ここ数年あまり変わっていません。しかし、先述したように、攻撃者はクラウドを利用するためには、自らの戦術を調整しなければなりませんでした。クラウドはそのアーキテクチャーと組み込み型の保護機能により、ある意味では特定のタスク（ランサムウェアなど）をより困難にしています。その一方で、サプライチェーン攻撃のように、クラウドを利用することで、攻撃者が成功しやすくなっているケースもあります。クラウド環境では、プロセスを境界から解放し、それをよりオープンなものにできるからです。**私たち防御する側は、クラウド技術がもたらすメリットとリスクを理解した上で、それに適応しなければなりません。**

Sysdigが、お客様の環境を1秒1秒セキュアに保つのにいかに役立つかをご覧ください。

次のステップに進む。

デモを依頼 →

Sysdigについて

クラウド環境では、1秒1秒が重要となります。攻撃は驚くほどの速さで進行するため、セキュリティチームはビジネスを減速させることなく、攻撃から守る必要があります。Sysdigは、ランタイムインサイトとオープンソースのFalcoを利用することで、リスクにおける変化を即座に検知し、クラウド攻撃をリアルタイムで阻止します。Sysdigは、クラウドのワークロード、アイデンティティ、サービス全体におけるシグナルを相互に関連付けることで、隠れた攻撃経路を明らかにすると共に、真のリスクに優先順位を付けます。予防から防御に至るまで、Sysdigは企業がイノベーションという最重要事項に集中できるよう支援します。詳細は、sysdig.jpをご覧ください。

Sysdig. Secure Every Second.

sysdig

E-BOOK

EBK-005-JA REV. A 9/24