DDoS 耐性向上のための AWS ベストプラクティス

2016 年 6 月

注意

本文書は情報提供のみを目的としています。本文書は、発行時点における AWS の製品と対応を説明するものであり、予告なく変更される場合があります。お客様は、本文書の情報および AWS 製品またはサービスの利用について、ご自身の評価に基づき判断する責任を負います。いずれの AWS 製品またはサービスも、明示または黙示を問わずいかなる保証も伴うことなく、「現状のまま」提供されます。本文書のいかなる内容も、AWS とその関係会社、サプライヤー、またはライセンサーからの保証、表明、および契約上の責任、条件や確約を意味するものではありません。お客様に対する AWS の責任は AWS 契約によって規定されています。また、本文書は、AWS とお客様との間のいかなる契約の一部も構成するものではなく、また、当該契約が本文書によって変更されることもありません。

目次

要約

このホワイトペーパーはアマゾン ウェブ サービス (AWS) で動作するアプリケーションに対して分散型サービス妨害 (DDoS) 攻撃が行われた場合にそのアプリケーションの耐性を向上させたいお客様を対象としています。本書では DDoS 攻撃の概要、AWS が提供する機能、攻撃緩和技術さらにはアプリケーションの可用性を確保するための指針となるような DDoS 攻撃に対して耐性の高いリファレンスアーキテクチャについて説明します。

はじめに

本書はネットワーク、セキュリティ、および AWS の分野について基本的な概念に精通している IT 関連の意思決定者やセキュリティ担当者を対象としています。各セクションにはベストプラクティスまたは機能の詳細が記載された AWS ドキュメントへのリンクがあります。詳細については AWS re:Invent のカンファレンスセッションの SEC307 - Building a DDoS-Resilient Architecture with AWS[1] および SEC306 - Defending Against DDoS Attacks[2] を視聴することでも確認することができます。

DDoS 攻撃

サービス拒否 (DoS) 攻撃はウェブサイトやアプリケーションをエンドユーザーが利用できないようにする攻撃です。攻撃者はこの攻撃を成功させるためにネットワークやその他のリソースを消費するさまざまな手法を駆使して正当なエンドユーザーによるアクセスを中断させます。最も単純な攻撃形態は図 1 のように単独の攻撃者が標的に対する DoS 攻撃を 1 つのソースから実行する方法です。

図 1: DOS 攻撃の仕組み

分散型サービス妨害 (DDoS) 攻撃の場合、攻撃者は複数のソース (協力者グループによって侵入されたり、制御されたりしている可能性があるソース) を組み合わせて標的を攻撃します。図 2 に示すように DDoS 攻撃では協力者または侵入されたホストがそれぞれ攻撃に参加し、標的が対応しきれないような大量のパケットやリクエストを発生させます。

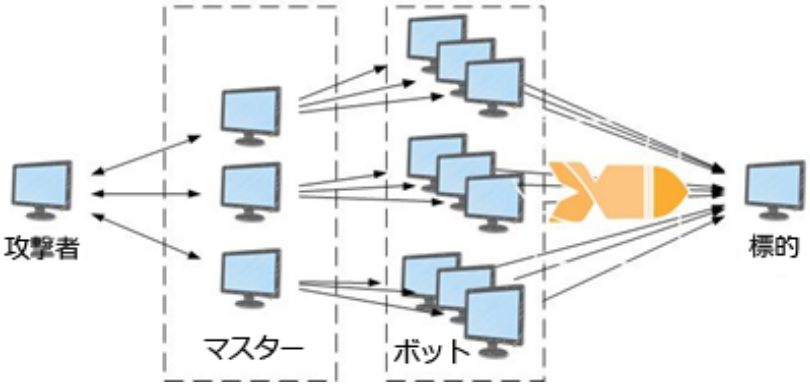

図 2: DDoS 攻撃の仕組み

DDoS 攻撃は表 1 に記載されている OSI (開放型システム間相互接続) モデルのレイヤー 3、4、6、および 7 に対して実行されるのが最も一般的です。レイヤー 3 と 4 に対する攻撃はそれぞれ OSI モデルのネットワーク層とトランスポート層に対応します。本書ではそのような攻撃をまとめてインフラストラクチャ層への攻撃と呼びます。レイヤー 6 と 7 に対する攻撃はそれぞれ OSI モデルのプレゼンテーション層とアプリケーション層に対応します。本書ではそのような攻撃をまとめてアプリケーション層への攻撃と呼びます。

番号	層	単位	説明	ベクトルの例
7	アプリケーション	データ	ネットワークプロセスからアプリケーションへ	HTTP フラッド、DNS クエリフラッド
6	プレゼンテーション	データ	データ表現と暗号化	SSL 悪用
5	セッション	データ	ホスト間通信	該当なし
4	トランスポート	セグメント	エンドツーエンドの接続と信頼性	SYN フラッド
3	ネットワーク	パケット	パスの決定と論理アドレス指定	UDP リフレクション攻撃
2	データリンク	フレーム	物理アドレス指定	該当なし
1	物理	ビット	メディア、信号、バイナリの送信	該当なし

表 1: 開放型システム間相互接続 (OSI) モデル

層によって利用される攻撃の種類が異なるので、上記のように攻撃を区別することは重要です。全体の攻撃耐性の向上のためには各層の耐性力を向上する必要があり、それぞれの攻撃手法に応じて異なる対応が必要となるからです。

インフラストラクチャ層への攻撃

最も一般的な DDoS 攻撃である UDP (User Datagram Protocol) リフレクション攻撃や SYN フラッドはインフラストラクチャ層を狙った攻撃です。攻撃者はこのような方法のいずれかを使用して、サーバー、ファイアウォール、IPS またはロードバランサーなどのネットワークやシステムリソースでは処理できないような大量のトラフィックを生成できます。このような攻撃には明確なシグネチャが存在するので、検出が容易です。しかしながら攻撃を効果的に緩和するには、攻撃者が生成するトラフィックを超える処理能力を持つネットワーク容量やシステムリソースが必要です。

UDP はステートレスプロトコルです。攻撃者はこの特性を悪用してサーバーに送信されるリクエストの送信元を偽装して、よりサイズの大きなレスポンスを引き出す

ことができます。リクエストサイズとレスポンスサイズの比である増幅係数は Domain Name System (DNS) や Network Time Protocol (NTP)や Simple Service Discovery Protocol (SSDP) など使用されるプロトコルによって異なります。たとえば DNS の増幅係数は 28～54 の範囲になります。つまり、攻撃者は 64 バイトのリクエストペイロードを DNS サーバーに送信することにより 3400 バイトを超える不要なトラフィックを生成できます。図 3 ではこの概念を説明しています。

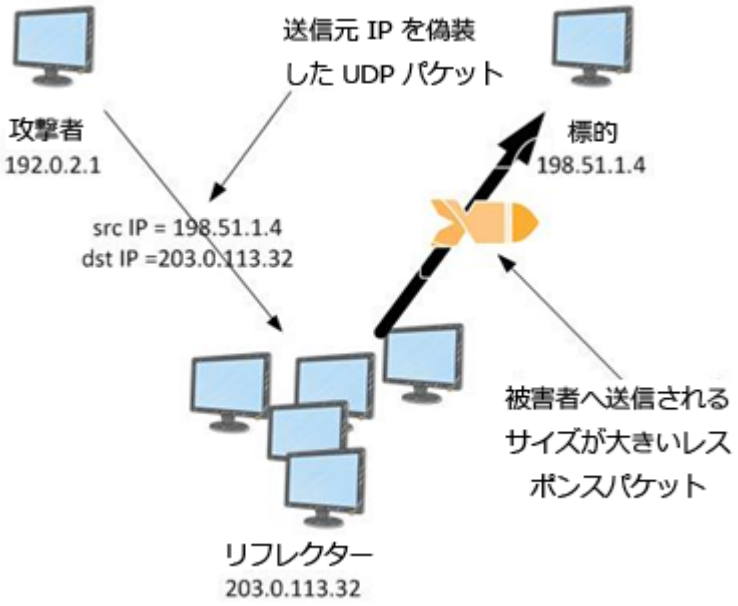

図 3: UDP リフレクション攻撃

SYN フラッドは数十 Gbps 規模になる可能性がありますが、攻撃の目的は通常、接続をハーフオープン状態のままにして、システムの利用可能なリソースを使い切ることにあります。図 4 に示すように、エンドユーザーがウェブサーバーのような TCP サービスに接続するとクライアントは SYN パケットを送信します。サーバーは SYN-ACK を返し、クライアントは ACK を返して 3 ウェイハンドシェイクを完了します。

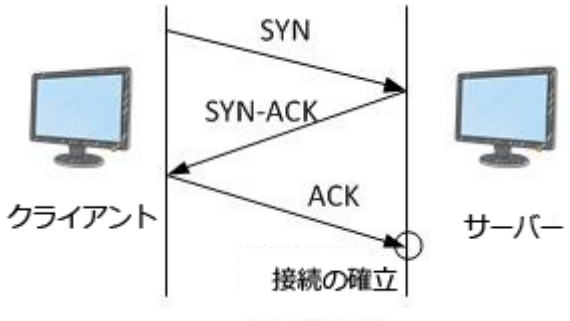

図 4: SYN 3 ウェイハンドシェイク

SYN フラッドでは ACK が返されず、サーバーはレスポンス待機状態になります。これによりリソースが枯渇することで、新規ユーザーがサーバーに接続できなくなる可能性があります。

アプリケーション層への攻撃

頻度は少ないですが、攻撃者がレイヤー 7、すなわちアプリケーション層を攻撃することでアプリケーション自体を標的にすることがあります。このような攻撃はインフラストラクチャ層への攻撃とは異なります。その理由は、攻撃者がアプリケーションの特定の機能を使用できないようにする目的でその機能を過剰に使用しようと試みる点にあります。場合によっては、このような攻撃は、大量のネットワークトラフィックを生成することなく、非常に少量のリクエストで実現できます。そのため、攻撃を検出して緩和するのがより困難になります。アプリケーション層への攻撃には HTTP フラッド、キャッシュ無効化攻撃、WordPress XML-RPC フラッドなどがあります。

HTTP フラッドでは、一見ウェブアプリケーションの実際のユーザーからのものと思わせるような HTTP リクエストを攻撃者が送信します。特定のリソースを対象とする HTTP フラッドもあれば、人間の動作をエミュレートしようとするようなより複雑な HTTP フラッドもあります。その結果、リクエスト率の制限などの一般的な攻撃緩和手法を使用するのがさらに難しくなる可能性があります。キャッシュ無効

化攻撃は HTTP フラッドの一種で、コンテンツデリバリーネットワーク (CDN) によるキャッシュ利用を回避するためにクエリ文字列中の文字列をさまざまに変化させます。この攻撃の結果、オリジンフェッチが発生し、オリジンウェブサーバーに負担がかかります。

WordPress XML-RPC フラッド (別名 WordPress ピンバックフラッド) では、攻撃者は WordPress のコンテンツ管理ソフトウェアでホストされているウェブサイトの XML-RPC API 関数を悪用して、大量の HTTP リクエストを生成できます。ピンバック機能により、WordPress でホストされているウェブサイト (サイト A) から、別の WordPress サイト (サイト B) に、サイト A がサイト B へのリンクを作成したことを通知できます。その結果、サイト B はサイト A をフェッチして、リンクの存在を確認しようとします。ピンバックフラッドの場合、攻撃者はこの機能を悪用してサイト B にサイト A を攻撃させます。このタイプの攻撃では、HTTP リクエストヘッダーの「User-Agent」に「WordPress」が記載されている必要があるため、明確なシグネチャがあります。

アプリケーション層への攻撃は、ドメインネームシステム (DNS) サービスを標的にする可能性もあります。このような攻撃で最も一般的なものは DNS クエリフラッドで、攻撃者は正しい形式の DNS クエリを大量に使用して DNS サーバーのリソースを枯渇させます。このような攻撃にはキャッシュ無効化の要素も含まれており、攻撃者はサブドメイン文字列をランダムに変化させることでリゾルバのローカル DNS キャッシュを回避します。その結果、権威 DNS サーバーに対する攻撃にリゾルバも加担することになります。

セキュアソケットレイヤー (SSL) を使用して配信されるウェブアプリケーションの場合、攻撃者は SSL ネゴシエーションプロセスを攻撃できます。SSL は計算コストが高いので、攻撃者は判読不可能なデータを送信することでサーバーの可用性に影響を及ぼすことができます。この攻撃の別の形態には、攻撃者が SSL ハンドシェイ

クを完了しても暗号化方式の再ネゴシエーションを絶え間なく繰り返す攻撃があります。同様に、攻撃者は数多くの SSL セッションを開いたり閉じたりして、サーバーリソースを枯渇させることもできます。

緩和手法

AWS のインフラストラクチャは、DDoS 攻撃に対する耐性を備えるように設計されており、過剰なトラフィックを自動的に検出してフィルタリングする DDoS 攻撃緩和システムによってサポートされています。アプリケーションの可用性を保護するには、このようなインフラストラクチャの機能を活用できるアーキテクチャを実装する必要があります。

最も一般的な AWS ユースケースの 1 つは、静的なコンテンツと動的なコンテンツをインターネット経由でユーザーに提供するウェブアプリケーションです。ウェブアプリケーションで一般的に使用される DDoS 攻撃に対する耐性の高いリファレンスアーキテクチャについては図 5 を参照してください。

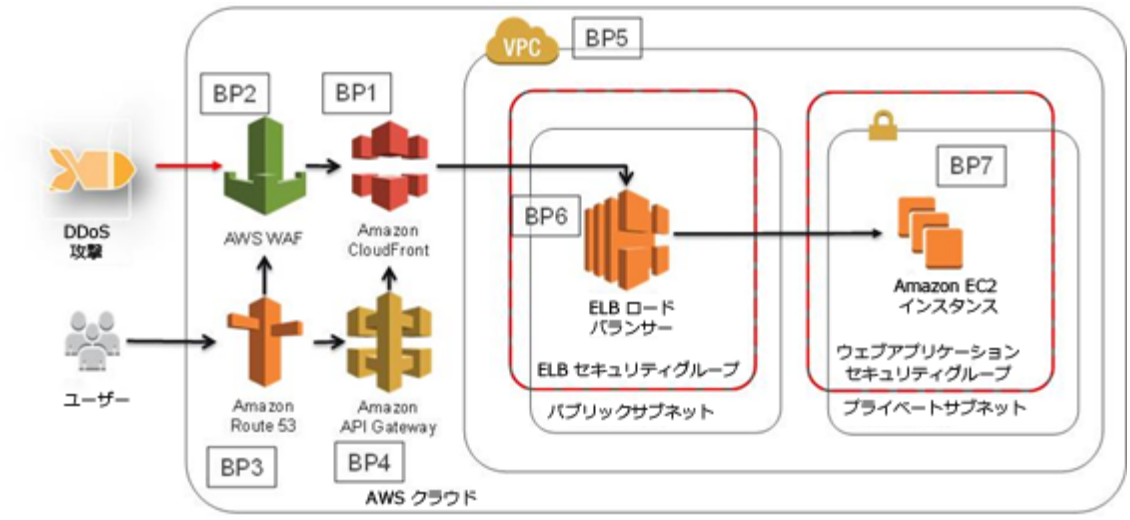

図 5: DDoS 攻撃に対する耐性の高いリファレンスアーキテクチャ

このリファレンスアーキテクチャには、DDoS 攻撃に対するウェブアプリケーションの耐性を向上させるのに役立つさまざまな AWS のサービスが含まれています。このアーキテクチャのベストプラクティスには、本書の中で説明される際に参照しやすいように略号が付けられています。たとえば、Amazon CloudFront が提供する機能について説明するセクションには、ベストプラクティスを表す略号 (BP1 など) が付けられています。このようなサービスの概要と提供可能な機能については表 2 をご覧ください。

	AWS エッジロケーション			**AWS リージョン**		
	AWS WAF を備えた Amazon CloudFront (BP1、BP2)	Amazon API Gateway (BP4)	Amazon Route 53 (BP3)	Elastic Load Balancing (BP6)	Amazon VPC (BP5)	Auto Scaling を備えた Amazon EC2 (BP7)
レイヤー 3 (UDP リフレクションなど) 攻撃の緩和	✔	✔	✔	✔	✔	
レイヤー 4 (SYN フラッドなど) 攻撃の緩和	✔	✔	✔	✔		
レイヤー 6 (SSL など) 攻撃の緩和	✔	✔	該当なし	✔		
攻撃対象領域の縮小	✔	✔	✔	✔	✔	
規模を拡大してアプリケーション層のトラフィックによる影響を緩和	✔	✔	✔	✔		✔
レイヤー 7 (アプリケーション層) 攻撃の緩和	✔	✔	✔			
過剰なトラフィックとより大規模な DDoS 攻撃の地理的な隔離と分散	✔	✔	✔			

表 2: ベストプラクティスのまとめ

Elastic Load Balancing や Amazon Elastic Compute (EC2) などの AWS リージョン内で利用できるサービスを利用することで、所定のリージョン内で DDoS 攻撃に対する高い耐性を持ち、予期しないトラフィック量に対処できるように拡張するシステムを構築することができます。Amazon CloudFront、AWS WAF、Amazon Route 53、Amazon API Gateway などの AWS エッジロケーションで利用できるサービスを利用することで、エッジロケーションのグローバルネットワークを活用し、アプリケーションの耐障害性を向上させつつ大量のトラフィックに対応することができます。このようなサービスを利用してインフラストラクチャ層とアプリケーション層への DDoS 攻撃に対する耐性を高めるメリットについては、次のセクションで説明します。

インフラストラクチャ層での防御 (BP1、BP3、BP6、BP7)

従来のデータセンター環境では、処理能力を過剰に持つこと (オーバープロビジョニング)、DDoS 攻撃緩和システムの導入、DDoS 攻撃緩和サービスを利用したトラフィックのスクラビングなどの手法を使用することで、インフラストラクチャ層への DDoS 攻撃を緩和できます。AWS では多額の投資を行ったり不必要に複雑にしたりすることなく、規模を拡大して大量のトラフィックによる影響を緩和できるようにアプリケーションを設計することができます。非常に大規模な DDoS 攻撃を緩和するには、利用可能なネットワークキャパシティや伝送経路の多様性を考慮することで EC2 インスタンスなどの AWS リソースを攻撃トラフィックから保護することが重要です。

インスタンスサイズ (BP7)

AWS のお客様の多くは、計算能力の規模を自由に変更する目的で Amazon EC2 を使用しています。これにより、要件の変化に応じて素早く規模の拡大または縮小を行えます。必要に応じて、アプリケーションにインスタンスを追加することによ

り、水平方向に規模を調整できます。また、より大きなインスタンスを使用して垂直方向に規模を調整することもできます。一部のインスタンスタイプは、大量のトラフィックを処理する能力を向上させることができる 10 ギガビットネットワークインターフェイスや拡張ネットワーキングなどの機能をサポートしています。

10 ギガビットのネットワークインターフェイスでは、各インスタンスがより多くのトラフィックに対応できます。これにより、Amazon EC2 インスタンスに到達したトラフィックによるインターフェイスの輻輳を防ぐことができます。拡張ネットワーキングをサポートするインスタンスでは、従来の実装と比較して、I/O パフォーマンスが向上し、CPU 使用率が低下します。これにより、パケット量の多いトラフィックを処理するインスタンスの能力が向上します。AWS では、お客様はインバウンドデータ転送の費用を負担する必要はありません。

10 ギガビットネットワークインターフェイスと拡張ネットワークをサポートする Amazon EC2 インスタンスの詳細については、Amazon EC2 インスタンスのタイプ[3] を参照してください。拡張ネットワークを有効にする方法については、VPC 内の Linux インスタンスでの拡張ネットワークの有効化[4]を参照してください。

リージョンの選択 (BP7)

Amazon EC2 などの AWS のサービスの多くは、世界中の複数の場所で利用できます。このような地理的に離れたエリアは AWS リージョンと呼ばれます。アプリケーションの構築時には、お客様は自らの要件に基づいて 1 つ以上のリージョンを選択できます。一般的にはパフォーマンス、コスト、データ主権などを考慮し選択されます。AWS では、各リージョン毎にインターネット接続とピアリング関係を用意することで、同じような地理的条件にあるエンドユーザーに最適なレイテンシーとスループットを提供しています。

また、DDoS 攻撃耐性の観点からリージョンの選択肢を検討することも重要です。リージョンの多くは、大規模なインターネットエクスチェンジの近くに存在しています。DDoS 攻撃の多くは国をまたいで発生するため、国際的な通信事業者や大規模なピアが強固な基盤を常に維持しているようなエクスチェンジポイントに近いことが有利に働きます。そうすることで、大量のトラフィックを処理している場合でも、エンドユーザーはアプリケーションにアクセスできるようになります。

リージョンの選択の詳細については、リージョンとアベイラビリティーゾーン[5]を参照し、十分な情報に基づいて決定できるように各リージョンの特性についてアカウントチームにお問い合わせください。

ロードバランシング (BP6)

より大規模な DDoS 攻撃が発生した場合、1 つの Amazon EC2 インスタンスのサイズを超える可能性があります。このような攻撃を緩和するには、過剰なトラフィックの負荷分散という選択肢について検討する必要があります。Elastic Load Balancing (ELB) を使用すると、多数のバックエンドインスタンスにトラフィックを分散させることで、アプリケーションへの過負荷リスクを軽減できます。ELB は自動的に規模を調整できるため、フラッシュクラウドや DDoS 攻撃などの大量の予期しないトラフィックを管理できます。

ELB は正しい形式の TCP 接続のみを受け入れます。つまり、SYN フラッドや UDP リフレクション攻撃のような一般的な DDoS 攻撃の多くは ELB によって拒否され、アプリケーションまで到達しません。ELB はこのようなタイプの攻撃を検出すると、追加のトラフィックによる影響を緩和するように自動的に拡張しますが、追加料金は発生しません。

ELB を使用して負荷を分散し、Amazon EC2 インスタンスを保護する方法の詳細については、Elastic Load Balancing の概要[6]を参照してください。

AWS エッジロケーションを使用した大規模配信 (BP1、BP3)

高いスケーラビリティと多くのネットワークプロバイダと接続した環境を利用することはエンドユーザーへのレイテンシーとスループットを最適化しつつ DDoS 攻撃の影響緩和（可用性への影響を最小限に抑えながら障害を隔離できるため）に高い効果があります。AWS エッジロケーションにより、ネットワークインフラストラクチャの追加レイヤーが提供され、Amazon CloudFront と Amazon Route 53 を使用するウェブアプリケーションに上記のようなメリットをもたらします。このようなサービスでは、エンドユーザーにより近い場所からコンテンツが配信され、DNS クエリが解決されます。

エッジでのウェブアプリケーションの配信 *(BP1)*

Amazon CloudFront はコンテンツデリバリーネットワーク (CDN) サービスの一種で、静的、動的、ストリーミング、インタラクティブコンテンツなどから構成されるウェブサイト全体を配信する目的で使用できます。TCP 持続接続と変更可能な TTL (Time-to-Live) を使用することでコンテンツの配信を高速化できます。これはコンテンツをエッジロケーションでキャッシュできない場合でも同様です。Amazon CloudFront は正しい形式の接続のみを受け入れ、SYN フラッドや UDP リフレクション攻撃などの多くの一般的な DDoS 攻撃がオリジンに到達しないようにします。これにより、静的コンテンツを提供していなくても Amazon CloudFront を使用してウェブアプリケーションを保護できます。DDoS 攻撃は地理的にソースの近くで隔離されるため、トラフィックが他の場所に影響を及ぼすことはありません。このような機能により、大規模な DDoS 攻撃にさらされているときであってもエンドユーザー向けのトラフィックを確保し続ける能力が大幅に向上します。Amazon CloudFront を使用することで、AWS またはインターネットの他の場所にあるオリジンを保護できます。

Amazon CloudFront を使用してウェブアプリケーションのパフォーマンスを最適化する方法の詳細については、CloudFront の概要[7]を参照してください。

エッジロケーションでのドメイン名の解決 *(BP3)*

Amazon Route 53 は高可用性を備え、拡張性に優れたドメインネームシステム (DNS) サービスで、ウェブアプリケーションにトラフィックを誘導するために使用されます。トラフィックフロー、レイテンシーベースルーティング、地域ベース、ヘルスチェックおよびモニタリングなどの数多くの高度な機能が含まれています。このような機能を使用すると、レイテンシー、ヘルスチェック状態、その他考慮すべき要素を最適化するために、サービスが DNS リクエストにどのように応答するかを制御できます。これにより、ウェブアプリケーションのパフォーマンスを向上させ、サイトの停止を回避することができます。

Amazon Route 53 では、シャッフルシャーディングとエニーキャストストライピングを使用することにより、DNS サービスが DDoS 攻撃の標的となっている場合でもエンドユーザーはアプリケーションにアクセスできます。シャッフルシャーディングでは、委任セットに含まれる各ネームサーバーがエッジロケーションとインターネットパスの一意のセットに対応します。これにより、耐障害性が向上し、お客様間の重複が最小限に抑えられます。エンドユーザーが委任セットに含まれるネームサーバーを利用できない場合、再試行することで異なるエッジロケーションの別のネームサーバーからレスポンスを受信できます。エニーキャストストライピングを使用すると、各 DNS リクエストが最適な場所で処理されます。これにより、負荷が分散されるとともに DNS レイテンシーが改善され、エンドユーザーはレスポンスをより迅速に受け取ることができます。さらに、Amazon Route 53 では DNS クエリの送信元の異常やその量を検出できるので、信頼できると考えられるユーザーからのリクエストを優先的に処理できます。

Amazon Route 53 のホストゾーンが多数ある場合、再利用可能な委任セット (Reusable Delegation Sets) を作成することで、ドメインごとに同一の正式なネームサーバーセットを提供できます。これにより、ホストゾーンの維持管理が容易になります。DDoS 攻撃が発生した場合、AWS は単一の緩和策を適用するだけで、再利用可能な委任セットが使用されているすべてのホストゾーンを網羅できます。

Amazon Route 53 を使用してエンドユーザーをアプリケーションに誘導する方法の詳細については、Amazon Route 53 の概要[8]を参照してください。再利用可能な委任セットの詳細については、再利用可能な委任セットでのアクション[9]を参照してください。

アプリケーション層での防御 (BP1、BP2、BP6)

本書で説明している手法の多くは、インフラストラクチャ層への DDoS 攻撃による可用性への影響を緩和するのに効果的です。アプリケーション層への攻撃からアプリケーションを防護するには、悪意のあるリクエストの影響を緩和したり、そのようなリクエストをブロックしたりするために検出や規模の拡張が可能なアーキテクチャを実装する必要があります。ネットワークベースの DDoS 攻撃緩和システムは、アプリケーション層への複雑な攻撃には一般的に効果がないため、上記のような点を考慮することが重要です。

悪意のあるウェブリクエストの検出とフィルタリング (BP1、BP2)

ウェブアプリケーションファイアウォール (WAF) は、アプリケーションの脆弱性を悪用しようとする攻撃からウェブアプリケーションを保護するために使用されることが一般的です。このような例としては、SQL インジェクションやクロスサイトリクエストフォージェリなどがあります。ウェブアプリケーション層の DDoS 攻撃を検出し緩和することができる WAF も存在します。

AWS では、Amazon CloudFront と AWS WAF を使用してこのような攻撃からアプリケーションを防御できます。Amazon CloudFront を使用すると、静的コンテンツをキャッシュして AWS エッジロケーションから提供することで、オリジンへの負荷を軽減できます。さらに、Amazon CloudFront は長時間にわたる少量ずつの読み込み/書き込み処理を悪用する攻撃 (Slowloris など) に対して接続を自動的に閉じることができます。また、Amazon CloudFront の地域制限機能を使用すると特定の地域のユーザーがコンテンツにアクセスできないように制限できます。この機能はエンドユーザーにサービスを提供することを想定していないような地域からの攻撃をブロックする場合に役立ちます。

HTTP フラッドまたは WordPress ピンバックフラッドなど、他の種類の攻撃に対しては AWS WAF を使用した緩和策を適用できます。ブロックする送信元 IP アドレスがわかっている場合は、ブロックするアクションを含むルールを作成してウェブ ACL に関連付けることができます。その後、ウェブ ACL に IP アドレス一致条件を作成すると、攻撃に参加している送信元 IP アドレスをブロックすることができます。また、URI、クエリ文字列、HTTP メソッド、またはヘッダーの値などによりブロックする条件を持つルールを作成することもできます。後者の方法は、明確なシグネチャを持つ攻撃の場合に有効です。たとえば、WordPress ピンバック攻撃には、「User-Agent」に「WordPress」が必ず含まれています。

DDoS 攻撃ではシグネチャを特定したり、攻撃に参加している IP アドレスを正確に特定したりするのは難しい場合がありますが、ウェブサーバーのログを確認することでこの情報を特定できる場合もあります。また、AWS WAF コンソールを使用して、Amazon CloudFront が AWS WAF に転送したリクエストの例を表示することもできます。ここで表示されたリクエストの例は、アプリケーション層への攻撃を緩和するために必要なルールを決定するのに役立ちます。ランダムなクエリ文字列を含むリクエストが多数ある場合、Amazon CloudFront でクエリ文字列の転送を無効

にすることができます。これは、キャッシュ無効化攻撃を緩和するのに役立ちます。

エンドユーザーからの正常なトラフィックに見せかけたウェブトラフィックによる攻撃もあります。このようなタイプの攻撃を緩和するには、AWS Lambda サービスを使用してレートベースのブラックリストを実装できます。レートベースのブラックリストでは、ウェブアプリケーションが処理できるリクエスト数のしきい値を設定できます。ボットまたはクローラーがこの制限を超えると、AWS WAF を使用してそれ以降のリクエストを自動的にブロックできます。

地域制限を使用して Amazon CloudFront ディストリビューションへのアクセスを制限する方法の詳細については、コンテンツの地域的ディストリビューションの制限[10]を参照してください。

AWS WAF の使用の詳細については、AWS WAF の概要[11]と CloudFront が AWS WAF に転送したウェブリクエストのサンプルの表示[12]を参照してください。

AWS Lambda と AWS WAF でレートベースのブラックリストを設定する方法については、How to Configure Rate-Based Blacklisting with AWS WAF and AWS Lambda[13]を参照してください。

規模の拡大による吸収 (BP6)

アプリケーション層への攻撃に対処するもう 1 つの方法は、攻撃に耐えうる規模でそもそも運用する方法です。ウェブアプリケーションの場合、ELB を使用することでトラフィックの急増に対処するためにあらかじめ多めにプロビジョニングされた、または自動スケーリングが設定された多数の Amazon EC2 インスタンスにトラフィックを分散できます。これはトラフィックの急増の原因が攻撃によらずに発生した急増（フラッシュクラウド）であっても、アプリケーション層への DDoS 攻撃であっても変わりません。Amazon CloudWatch アラームを使用して Auto Scaling

を開始すると定義したイベントに応じて Amazon EC2 群の規模が自動的に拡大/縮小されます。これにより、予期していない量のリクエストを処理する場合でもアプリケーションの可用性が担保されます。また、Amazon CloudFront または ELB を使用すると SSL ネゴシエーションがディストリビューションまたはロードバランサーによって処理され、使用中のインスタンスが SSL ベースの攻撃の影響を受けないようにできます。

Amazon CloudWatch を使用して Auto Scaling を呼び出す方法の詳細については、Amazon CloudWatch を使用した自動スケーリングインスタンスとグループのモニタリング[14]を参照してください。

攻撃対象領域の縮小

AWS で構築する際に考慮するべきもう 1 つの重要な点は、攻撃者がアプリケーションを標的にする機会を制限することです。たとえば、エンドユーザーが特定のリソースと直接やりとりすることが想定されていない場合、インターネットからそのリソースにアクセスできないようにします。同様に、エンドユーザーまたは外部アプリケーションが特定のポートまたはプロトコルでアプリケーションと通信することが想定されていない場合は、トラフィックが受け付けられないようにします。このような概念は、攻撃対象領域の縮小 (Attack Surface Reduction) として知られています。このセクションでは、攻撃対象領域を縮小し、アプリケーションがインターネットにさらされる範囲を制限するのに役立つベストプラクティスを紹介します。インターネットにさらされていないリソースは攻撃するのが難しくなり、攻撃者がアプリケーションを標的にできる選択肢が著しく制限されます。

AWS リソースの難読化 (BP1、BP4、BP5)

多くのアプリケーションでは AWS リソースをインターネットに完全に公開する必要はありません。たとえば、ELB の背後にある Amazon EC2 インスタンスを公開してアクセス可能にする必要はありません。このシナリオでは、エンドユーザーは特定の TCP ポートから ELB にアクセスし、ELB のみが Amazon EC2 インスタンスと通信できるようにできます。これは使用中の Amazon Virtual Private Cloud (VPC) でセキュリティグループとネットワークアクセスコントロールリスト (NACL) を設定することにより実現できます。Amazon VPC では、AWS クラウドの論理的に分離されたセクションをプロビジョニングし、定義した仮想ネットワークで AWS リソースを起動できます。

セキュリティグループとネットワーク ACL は、VPC 内の AWS リソースへのアクセスを制御できるという点で似ています。セキュリティグループを使用すると、インバウンドとアウトバウンドのトラフィックをインスタンスレベルで制御でき、ネットワーク ACL は同様の機能を VPC サブネットレベルで提供します。さらに、Amazon EC2 セキュリティグループ (SG) ルールまたはネットワーク ACL で制御するインバウンドデータ転送は無料です。これにより、セキュリティグループまたはネットワーク ACL によってドロップされたトラフィックに対して、追加料金が発生することはありません。

セキュリティグループ (BP5)

セキュリティグループは、インスタンスを起動する際に指定するか、後でインスタンスをセキュリティグループに関連付けることで設定できます。トラフィックを許可する許可ルールを作成しないかぎり、インターネットからセキュリティグループへのトラフィックはすべて暗黙的に拒否されます。たとえば、ELB と多数の Amazon EC2 インスタンスで構成されるウェブアプリケーションがある場合、ELB に 1 つのセキュリティグループ (「ELB セキュリティグループ」) を、インスタンス

にもう 1 つのセキュリティグループ (「ウェブアプリケーションサーバーセキュリティグループ」) を作成したとします。その後、許可ルールを作成し、インターネットから ELB セキュリティグループへのトラフィックを許可し、ELB セキュリティグループからウェブアプリケーションサーバーセキュリティグループへのトラフィックを許可できます。その結果、インターネットからのトラフィックは Amazon EC2 インスタンスと直接通信できなくなり、攻撃者がアプリケーションについて情報を得ることがより難しくなります。

ネットワークアクセスコントロールリスト (ACL) (BP5)

ネットワーク ACL では、許可ルールと拒否ルールの両方を指定できます。これは、アプリケーションへの特定のタイプのトラフィックを明示的に拒否する場合に便利です。たとえば、サブネット全体で拒否する IP アドレス (CIDR による範囲)、プロトコル、宛先ポートを定義できます。アプリケーションを TCP トラフィックにのみ使用している場合、すべての UDP トラフィックを拒否するルールを作成することもその逆も可能です。このツールは送信元 IP アドレスやその他のシグネチャが判明している場合に攻撃を緩和する独自のルールを作成できるため、DDoS 攻撃に対処する際に役立ちます。

オリジンの保護 (BP1)

VPC の中にオリジンを持つ Amazon CloudFront を使用している場合、AWS Lambda サービスを使用して Amazon CloudFront からのトラフィックのみを許可するようにセキュリティグループルールを自動的に更新するようにします。これにより、Amazon CloudFront と AWS WAF をバイパスできないようになるので、オリジンの安全性が向上します。

セキュリティグループを自動的に更新してオリジンを保護する方法の詳細については、[How to Automatically Update Your Security Groups for Amazon CloudFront and AWS WAF by Using AWS Lambda[15]] を参照してください。

また、Amazon CloudFront ディストリビューションのみがオリジンにリクエストを転送するようにしたい場合もあります。オリジンカスタムヘッダーを使用すると、Amazon CloudFront がリクエストをオリジンに転送する際、既存のリクエストヘッダーの値を追加したり、上書きしたりすることができます。この機能を利用することでオリジンに対して行われたリクエストが Amazon CloudFront から送信されたかどうかを検証できます。

オリジンカスタムヘッダーでオリジンを保護する方法の詳細については、[オリジンへのカスタムヘッダーの転送[16]]を参照してください。

API エンドポイントの保護 (BP4)

通常、API をインターネットに公開する必要がある場合、API フロントエンドが DDoS 攻撃の標的になる危険性があります。Amazon API Gateway は完全なマネージドサービスです。これにより、Amazon EC2、AWS Lambda、またはその他のウェブアプリケーションで動作するアプリケーションの「フロントドア」のような役割を果たす API を作成できます。Amazon API Gateway を使用すると、お客様は独自のサーバーを運用して API フロントエンドを用意する必要がなくなり、アプリケーションの他のコンポーネントをインターネットから見えにくくできます。これにより、AWS リソースが DDoS 攻撃の標的になるのを防ぐことができます。Amazon API Gateway は Amazon CloudFront と統合されており、そのサービスの特徴として備える高い DDoS 攻撃耐性のメリットを享受できます。また、REST API の各メソッドに標準の制限またはバーストレート制限を設定することで、バックエンドを過剰なトラフィックから保護することもできます。

Amazon API Gateway での API の作成の詳細については、Amazon API Gateway の概要[17]を参照してください。

運用技術

本書の緩和技術を使用すると、DDoS 攻撃耐性を基本的な機能として備えたアプリケーションを設計できます。多くの場合、DDoS 攻撃がアプリケーションを標的にしているタイミングを知り、そのデータに基づいて措置を講じることができるようにすることも効果的です。他のリソースを活用して脅威を評価したり、アプリケーションのアーキテクチャを見直したり、その他の支援を依頼したりすることも検討する必要があります。このセクションでは、異常な動作の可視化、警告と自動化、さらには追加サポートを得るための AWS との連携に関するベストプラクティスについて説明します。

可視化

アプリケーションの正常な動作を理解すると、異常を検出した際により迅速に対応することができます。重要なメトリクスが期待値とは大きく異なる場合、攻撃者がアプリケーションの可用性を標的にしている可能性が考えられます。Amazon CloudWatch を使用すると、AWS で動作するアプリケーションを監視できるようになり、メトリクスの収集と追跡、ログファイルの収集と監視、アラームの設定、AWS リソースの変更を伴う自動的な対処も可能になります。DDoS 攻撃を検出したり、その攻撃に対処したりするために一般的に使用される Amazon CloudWatch のメトリクスの説明については、表 3 を参照してください。

トピック	メトリクス	説明
Auto Scaling	GroupMaxSize	Auto Scaling グループの最大サイズ

Amazon CloudFront	Requests	HTTP/S リクエストの数
Amazon CloudFront	TotalErrorRate	全リクエスト中の HTTP ステータスコードが 4xx または 5xx である割合
Amazon EC2	CPUUtilization	割り当て済みの EC2 コンピュートユニットのうち、現在使用されている比率
Amazon EC2	NetworkIn	すべてのネットワークインターフェイスでインスタンスが受信したバイト数
ELB	SurgeQueueLength	ルーティングを保留中のリクエストの総数
ELB	UnHealthyHostCount	ロードバランサーに登録された、異常なインスタンスの数
ELB	RequestCount	指定された間隔 (1 分または 5 分) の間に完了したリクエストの数、または接続の数
ELB	Latency	リクエストがロードバランサーから送信され、レスポンスが受信されるまでの経過時間 (秒)
ELB	HTTPCode_ELB_4xx HTTPCode_ELB_5xx	ロードバランサーで生成される HTTP 4xx または 5xx エラーコードの数
ELB	BackendConnectionErrors	ロードバランサーと登録されたインスタンス間で正常に確立されなかった接続数
ELB	SpilloverCount	サージキューがいっぱいなため拒否されたリクエストの数
Amazon Route 53	HealthCheckStatus	ヘルスチェックエンドポイントのステータス

表 3: 推奨される Amazon CloudWatch のメトリクス

図 5 に示された DDoS 攻撃耐性の高いリファレンスアーキテクチャに基づいて設計されたアプリケーションでは、インフラストラクチャ層への一般的な攻撃はアプリケーションに到達する前にブロックされます。その結果、このような攻撃は Amazon CloudWatch のメトリクスには現れません。

一方で、アプリケーション層への攻撃はそのようなメトリクスの多くで値を上昇させる可能性があります。たとえば、HTTP フラッドが原因で Amazon CloudFront、ELB、Amazon EC2 のメトリクスに対するリクエスト数や CPU とネットワークの使用率が上昇する可能性があります。バックエンドインスタンスが過剰なリクエストを処理できない場合は、Amazon CloudFront で TotalErrorRate が、さらには ELB で SurgeQueueLength、UnHealthyHostCount、Latency、BackendConnectionErrors、SpilloverCount、または HTTPCode の値が上昇することがあります。この場合、アプリケーションが正規のエンドユーザーにサービスを提供できないため、HTTP リクエストの量が減少する可能性があります。アプリケーションのバックエンドを拡大するか、本書で前述した AWS WAF で過剰なトラフィックをブロックすることでこの状況を改善できます。

Amazon CloudWatch を使用してアプリケーションに対する DDoS 攻撃を検出する方法の詳細については、Amazon CloudWatch の概要[18]を参照してください。

アプリケーションを標的とするトラフィックを可視化する目的で使用できるその他のツールとして、VPC フローログがあります。従来のネットワークでは、ネットワークフローログを使用して接続とセキュリティの問題についてトラブルシューティングを行い、ネットワークアクセスルールが想定どおりに機能していることを確認していました。VPC フローログを使用すると、VPC のネットワークインターフェイスとの間で送受信される IP トラフィックに関する情報を取得できます。

各フローログレコードには、送信元と送信先の IP アドレス、送信元と送信先のポート、プロトコル、キャプチャウィンドウ中に転送されたパケット数とバイト数が含まれます。この情報は、ネットワークトラフィックの異常を特定し、具体的な攻撃ベクトルを特定するのに役立ちます。たとえば、ほとんどの UDP リフレクション攻撃には特定の送信元ポート (DNS リフレクション攻撃では送信元ポート 53 など) があります。これは、フローログレコードで識別できる明確なシグネチャです。こ

れに対応して、インスタンスレベルで特定の送信元ポートをブロックするか、プロトコルそのものが不要な場合はそのプロトコル全体をブロックするネットワーク ACL ルールを作成できます。

VPC フローログを使用してネットワークの異常や DDoS 攻撃のベクトルを特定する方法の詳細については、VPC フローログ[19]と VPC Flow Logs – Log and View Network Traffic Flows[20] を参照してください。

サポート

実際の攻撃が発生する前に DDoS 攻撃に対する計画を立てることはとても重要です。本書で概要を説明したベストプラクティスは、事前対策を目的としており、DDoS 攻撃の標的となりうるアプリケーションを起動する前に実装する必要があります。アカウントチームは、お客様のユースケースとアプリケーションを確認して、具体的な質問に回答したり、直面する可能性のある課題について支援したりすることができます。

しかし、場合によっては DDoS 攻撃が行われている最中に AWS に連絡し、追加サポートを求めることが有益な場合があります。お客様の問題は迅速に処理され、支援が可能な専門家に転送されます。ビジネスサポートに加入すると、クラウドサポートエンジニアに電子メール、チャット、または電話で 24 時間 365 日いつでも相談できます。

AWS でミッションクリティカルなワークロードを実行している場合は、エンタープライズサポートへの加入を検討してください。エンタープライズサポートでは、お客様の緊急事態が最優先で処理され、シニアクラウドサポートエンジニアが対応します。さらに、エンタープライズサポートでは、お客様を支援し、技術的な問題について専属の連絡先となるテクニカルアカウントマネージャ (TAM) が割り当てられます。また、予定されたイベント、アプリケーションの起動、または移行の際に

は、リアルタイムで運用サポートを受けられる Infrastructure Event Management を利用することもできます。

お客様独自のニーズに合わせてサポートプランを選択する方法の詳細については、AWS サポートプランの比較[21]を参照してください。

結論

本書で概説しているベストプラクティスを使用すると、DDoS 攻撃に対する高い耐性をもったアーキテクチャを構築できます。これにより、インフラストラクチャ層とアプリケーション層に対する一般的な DDoS 攻撃の多くからアプリケーションの可用性を保護できます。どの程度このベストプラクティスに従ってアプリケーションを設計できるかは緩和できる DDoS 攻撃のタイプ、ベクトル、および規模に影響します。AWS は一般的な DDoS 攻撃に対してアプリケーションの可用性の保護を向上させるために、このようなベストプラクティスを使用することをお勧めします。

寄稿者

本書は、以下の個人および組織が寄稿しました。

- Andrew Kiggins (AWS ソリューションアーキテクト)
- Jeffrey Lyons (AWS DDoS オペレーションエンジニアリング)

注釈

[1] https://www.youtube.com/watch?v=OT2y3DzMEmQ
[2] https://www.youtube.com/watch?v=Ys0gG1koqJA
[3] https://aws.amazon.com/ec2/instance-types/

[4] http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/enhanced-networking.html

[5] http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html

[6] http://docs.aws.amazon.com/ElasticLoadBalancing/latest/DeveloperGuide/elb-getting-started.html

[7] http://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/GettingStarted.html

[8] http://docs.aws.amazon.com/Route53/latest/DeveloperGuide/getting-started.html

[9] http://docs.aws.amazon.com/Route53/latest/APIReference/actions-on-reusable-delegation-sets.html

[10] http://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/georestrictions.html

[11] http://docs.aws.amazon.com/waf/latest/developerguide/getting-started.html

[12] http://docs.aws.amazon.com/waf/latest/developerguide/web-acl-testing.html#web-acl-testing-view-sample

[13] https://blogs.aws.amazon.com/security/post/Tx1ZTM4DT0HRH0K/How-to-Configure-Rate-Based-Blacklisting-with-AWS-WAF-and-AWS-Lambda

[14] http://docs.aws.amazon.com/autoscaling/latest/userguide/as-instance-monitoring.html

[15] https://blogs.aws.amazon.com/security/post/Tx1LPI2H6Q6S5KC/How-to-Automatically-Update-Your-Security-Groups-for-Amazon-CloudFront-and-AWS-W

[16] http://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/forward-custom-headers.html

[17] https://aws.amazon.com/api-gateway/getting-started/

[18] http://docs.aws.amazon.com/AmazonCloudWatch/latest/DeveloperGuide/GettingStarted.html

[19] http://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/flow-logs.html

[20] https://aws.amazon.com/blogs/aws/vpc-flow-logs-log-and-view-network-traffic-flows/

[21] https://aws.amazon.com/premiumsupport/compare-plans/