H3C MSRルーター

攻撃の検出と防御の設定ガイド

New h3c Technologies Co.,Ltd.http://www.h3c.com

Document version: 6W103-20200507

Product version: R5426P02

注意

本書に記載されている情報は、予告なしに変更されることがあります。このドキュメントに記載されているすべての内容(記述、情報、推奨事項を含む)は、正確であると考えられますが、明示的であるか黙示的であるかを問わず、いかなる種類の保証もなく提供されています。H3Cは、本書に含まれている技術的または編集上の誤りまたは脱落に対して責任を負わないものとします。

内容

攻撃の検出と防御の設定

攻撃の検出と防止について

攻撃の検出と防止により、デバイスは着信パケットを検査して攻撃を検出し、プライベートネットワークを保護するための防止アクションを実行できます。防止アクションには、ロギング、パケットドロップ、ブラックリスト、およびクライアント検証が含まれます。

デバイスが防止できる攻撃

この項では、デバイスが検出および防止できる攻撃について説明します。

単一パケット攻撃

単一パケット攻撃は、不正形式パケット攻撃とも呼ばれます。攻撃者は通常、次の方法を使用して単一パケット攻撃を開始します。

- 攻撃者は欠陥のあるパケットをデバイスに送信し、デバイスの誤動作やクラッシュを引き起こします。
- 攻撃者は通常のパケットをデバイスに送信し、デバイスは接続を中断したり、ネットワークトポロジをプローブしたりします。
- 攻撃者がターゲットデバイスに偽造パケットを大量に送信すると、ネットワーク帯域幅が消費され、Denial of Service(DoS)が発生します。

表1に、デバイスが検出して防止できる単一パケット攻撃のタイプを示します。

表1単一パケット攻撃の種類

単一パケット攻撃	説明
ICMPリダイレクト	攻撃者はICMPリダイレクトメッセージを送信して、被害者のルーティングテーブルを変更します。被害者はパケットを正しく転送できません。
ICMP宛先到達不能	攻撃者は、ICMP宛先到達不能メッセージを送信して、被害者とその宛先の間の接続を切断します。
ICMPタイプ	受信者は、そのタイプに従ってICMPパケットに応答します。攻撃者は、被害者のパケット処理に影響を与えるために、特定のタイプの偽造ICMPパケットを送信します。
ICMPv6タイプ	受信者は、そのタイプに従ってICMPv6パケットに応答します。攻撃者は、被害者のパケット処理に影響を与えるために、特定のタイプの偽造されたICMPv6パケットを送信します。
Land	攻撃者は、大量のTCP SYNパケットを被害者に送信します。このパケットには、送信元と宛先のIPアドレスとして被害者のIPアドレスが含まれています。この攻撃により、被害者のハーフオープン接続リソースが枯渇し、被害者のシステムがロックされます。
大きなICMPパケット	攻撃者は、被害者をクラッシュさせるために大きなICMPパケットを送信します。大きなICMPパケットは、メモリ割り当てエラーを引き起こし、プロトコルスタックをクラッシュさせる可能性があります。

大きなICMPv6パケット	攻撃者は、被害者をクラッシュさせるために大きなICMPv6パケットを送信します。大きなICMPv6パケットは、メモリ割り当てエラーを引き起こし、プロトコルスタックをクラッシュさせる可能性があります。

単一パケット攻撃	説明
IPオプション	攻撃者は、特定のオプションタイプを使用してIPデータグラムを構築し、ネットワークトポロジをプローブするために送信します。
IPオプション異常	攻撃者は、IPオプションが異常なIPデータグラムを送信します。この攻撃は、ネットワークトポロジを調査することを目的としています。ターゲットシステムがエラーパケットを処理できない場合、ターゲットシステムは故障します。
IPフラグメント	攻撃者は、オフセットが5以下のIPデータグラムを被害者に送信し、被害者の誤動作やクラッシュを引き起こします。
IP不可能パケット	攻撃者は、送信元IPアドレスと宛先IPアドレスが同じIPパケットを送信し、被害者の誤動作を引き起こします。
小さな断片	攻撃者は、レイヤ4ヘッダーフィールドを2番目のフラグメントに強制するために、フラグメントサイズを十分に小さくします。これらのフラグメントは、一致しないため、パケットフィルタリングを通過できます。
smurf	攻撃者は、ターゲットネットワークにICMPエコー要求を送信します。これらの要求では、宛先IPアドレスはクラスA、B、またはCサブネットのネットワークまたはブロードキャストアドレスであり、送信元IPアドレスは被害者のIPアドレスです。ターゲットネットワーク上のすべての受信者は、被害者にICMPエコー応答を送信します。被害者は応答で溢れ、サービスを提供できなくなります。ネットワークの輻輳が発生する可能性があります。
TCPフラグ	攻撃者は、ターゲットホストのオペレーティングシステムをプローブするために、欠陥のあるTCPフラグを持つパケットを送信します。異なるオペレーティングシステムは、非従来型のTCPフラグを異なる方法で処理します。ターゲットシステムは、このタイプのパケットを誤って処理すると、故障します。
トレースルート	攻撃者は、tracerouteツールを使用して、被害者のネットワークのトポロジを調査します。
WinNuke	攻撃者は、Windowsシステムを実行している被害者のTCPポート139(NetBIOS)にアウトオブバンド(OOB)データを送信します。悪意のあるパケットには、不正な緊急ポインタが含まれており、被害者のオペレーティングシステムをクラッシュさせます。
UDP爆弾	攻撃者が不正な形式のUDPパケットを送信しました。IPヘッダーの長さの値が、IPヘッダーの長さにUDPヘッダーの長さの値を加えた値よりも大きくなっています。ターゲットシステムがパケットを処理すると、バッファオーバーフローが発生し、システムクラッシュが発生する可能性があります。
UDP Snork	攻撃者は、宛先ポート135(Microsoftロケーションサービス)と送信元ポート135、7、または19を持つUDPパケットを送信します。この攻撃により、NTシステムのCPUが使い果たされます。
UDP Fraggle	攻撃者は、送信元UDPポート7と宛先UDPポート19(UDP chargenポート)を持つ大量のパケットをネットワークに送信します。これらのパケットは、被害者のIPアドレスを送信元IPアドレスとして使用します。応答は被害者をあふれさせ、DoSを引き起こします。
Teardrop	攻撃者は、重複するフラグメントのストリームを送信します。被害者は、重複するフラグメントを再構成しようとするとクラッシュします。
Ping of death	攻撃者は、IPプロトコルに違反する65535バイトを超えるICMPエコー要求を被害者に送信します。被害者がパケットを再構成すると、バッファオーバーフローが発生し、システムクラッシュが発生する可能性があります。

IPv6拡張ヘッダー	攻撃は、IPv6拡張ヘッダーを持つパケットを被害者に送信します。
IPv6 extヘッダー異常	攻撃者は、IPv6拡張ヘッダーが不規則または反復されたIPv6パケットをターゲットに送信します。
IPv6拡張ヘッダー超過	攻撃者は、上限を超えるIPv6拡張ヘッダーを持つIPv6パケットをターゲットに送信します。

スキャン攻撃

スキャンは、ネットワークへの侵入に備えるために使用される侵入前アクティビティです。スキャンにより、攻撃者はターゲットネットワークへの侵入方法を見つけ、攻撃者の身元を隠すことができます。

攻撃者は、スキャンツールを使用してネットワークを調査し、脆弱なホストを見つけ、ホスト上で実行されているサービスを検出します。攻撃者は、この情報を使用して攻撃を開始できます。

デバイスは、IPスイープおよびポートスキャン攻撃を検出して防止できます。攻撃者が複数のホストからターゲットホストに対してポートスキャンを実行すると、分散ポートスキャン攻撃が発生します。

フラッドアタック

攻撃者は、短期間に大量の偽造要求を被害者に送信してフラッド攻撃を仕掛けます。被害者は、これらの偽造要求に応答するのに忙しすぎて、合法的なユーザーにサービスを提供できず、DoS攻撃が発生します。

このデバイスは、次のタイプのフラッド攻撃を検出して防止できます。

SYNフラッド攻撃

SYNフラッド攻撃者は、TCPの3ウェイハンドシェイク特性を不正利用し、被害者を正当なユーザーに反応しないようにします。攻撃者は、偽造された送信元アドレスを持つ多数のSYNパケットをサーバーに送信します。これにより、サーバーは多数のハーフオープン接続を開き、要求に応答します。ただし、サーバーは予想されたACKパケットを受信することはありません。すべてのリソースがハーフオープン接続にバインドされているため、サーバーは新しい着信接続要求を受け入れることができません。

ACKフラッド攻撃

ACKパケットは、ACKフラグが設定されているだけのTCPパケットです。クライアントからACKパケットを受信すると、サーバーはハーフオープン接続を検索して一致を確認する必要があります。

ACKフラッド攻撃者は、大量のACKパケットをサーバーに送信します。これにより、サーバーはハーフオープン接続の検索でビジー状態になり、サーバーは通常のサービスのパケットを処理できなくなります。

SYN-ACKフラッド攻撃

SYN-ACKパケットを受信すると、サーバーは送信したSYNパケットに一致するものを検索する必要があります。SYN-ACKフラッド攻撃者は、大量のSYN-ACKパケットをサーバーに送信します。これにより、サーバーはSYNパケットの検索でビジー状態になり、通常のサービスのパケットを処理できなくなります。

FINフラッド攻撃

FINパケットは、TCP接続をシャットダウンするために使用されます。

FINフラッド攻撃者は、偽造された多数のFINパケットをサーバーに送信します。被害者は、正しい接続をシャットダウンしたり、一致する接続の検索でビジー状態であるためにサービスを提供できない場合があります。

RSTフラッド攻撃

RSTパケットは、TCP接続エラーが発生したときにTCP接続を中断するために使用されます。

RSTフラッド攻撃者は、偽造されたRSTパケットを大量にサーバーに送信します。被害者は、正しい接続をシャットダウンしたり、一致する接続の検索でビジー状態であるためにサービスを提供できない場合があります。

DNSフラッド攻撃

DNSサーバーは、受信したすべてのDNSクエリーを処理し、応答します。

DNSフラッド攻撃者は、偽造されたDNSクエリーを大量に送信します。この攻撃は、DNSサーバーの帯域幅とリソースを消費し、サーバーが合法的なDNSクエリーを処理して応答することを妨げます。

DNS応答フラッド攻撃

DNSクライアントは、受信したすべてのDNS応答を処理します。

DNS応答フラッド攻撃者は、偽造されたDNS応答を大量に送信します。この攻撃は、DNSクライアントの帯域幅とリソースを消費し、クライアントが正当なDNS応答を処理できないようにします。

HTTPフラッド攻撃

HTTPサーバーは、HTTP GET要求やPOST要求を受信すると、文字列検索、データベーストラバーサル、データ再構成、フォーマットスイッチングなどの複雑な処理を実行します。これらの処理は、大量のシステムリソースを消費します。

HTTPフラッド攻撃者は、HTTPサーバーの処理能力を超える大量のHTTP GET要求またはPOST要求を送信し、サーバーをクラッシュさせます。

SIPフラッド攻撃

SIPクライアントからSIP INVITEパケットを受信した後、サーバーは、SIPクライアントとのセッションを確立およびトレースするためのリソースを割り当てる必要があります。

SIPフラッド攻撃者は、SIPサーバーの処理能力を超えるレートで大量の偽のINVITE要求パケットを送信し、サーバーをクラッシュさせます。SIPの詳細については、『Voice Configuration Guide』を参照してください。

ICMPフラッド攻撃

ICMPフラッド攻撃者は、pingパケットなどのICMP要求パケットを高速でホストに送信します。ターゲットホストはこれらの要求への応答でビジー状態であるため、サービスを提供できません。

ICMPv6フラッド攻撃

ICMPv6フラッド攻撃者は、pingパケットなどのICMPv6要求パケットを高速でホストに送信します。ターゲットホストはこれらの要求への応答でビジー状態であるため、サービスを提供できません。

UDPフラッド攻撃

UDPフラッド攻撃者は、UDPパケットを高速でホストに送信します。これらのパケットはターゲットホストの帯域幅を大量に消費するため、ホストは他のサービスを提供できません。

TCPフラグメント攻撃

攻撃者は、RFC 1858で定義されている攻撃TCPフラグメントを送信することによって、TCPフラグメント攻撃を開始します。

- TCPヘッダーが20バイトより小さい最初のフラグメント。
- フラグメントオフセットが8バイト(FO=1)の先頭以外のフラグメント。

通常、パケットフィルターは、TCPパケットの最初のフラグメントの送信元IPアドレスと宛先IPアドレ

ス、送信元ポートと宛先ポート、およびトランスポート層プロトコルを検出します。最初のフラグメントが検出に合格すると、TCPパケットの後続のすべてのフラグメントの通過が許可されます。

攻撃TCPパケットの最初のフラグメントは、パケットフィルター内のどの一致にもヒットしないため、後続のフラグメントはすべて通過できます。受信ホストがフラグメントを再構成した後、TCPフラグメント攻撃が発生します。

TCPフラグメント攻撃を防止するには、TCPフラグメント攻撃防止をイネーブルにして、攻撃TCPフラグメントをドロップします。

ログインDoS攻撃

ログインDoS攻撃では、悪意のあるユーザーがログイン要求を大量に送信してデバイスの通常の動作を妨害しようとする可能性があります。これらの要求は認証リソースを消費するため、デバイスは合法的なユーザーのログインを許可できなくなります。

ログイン攻撃防御を設定して、ログインDoS攻撃を防ぐことができます。この機能は、ユーザーがログイン試行の最大回数に失敗した後、一定期間ユーザーのログイン試行をブロックします。

ログイン辞書攻撃

ログインディクショナリ攻撃は、事前に用意された値のリスト(ディクショナリ)から可能なすべてのパスワードを試行してログインを試行する自動プロセスです。短時間に複数回のログイン試行が発生する可能性があります。

ログイン遅延機能を設定して、ログインディクショナリ攻撃を遅らせることができます。この機能を使用すると、デバイスは、ユーザーの失敗したログイン試行を検出した後、別のログイン要求の受け入れを遅延できます。

HTTP低速攻撃

攻撃者は、HTTP接続メカニズムを悪用してHTTPサーバーへの接続を確立し、サーバーリソースを使い果たすために接続を長時間保持します。一般的に使用されるHTTP低速攻撃のタイプは次のとおりです。

- **Slow headers**: 攻撃者は、HTTP GETまたはPOSTメソッドを使用してサーバーに接続します。HTTPヘッダーには、ヘッダーの終わりを示す2つのCRLFシーケンスが含まれていません。後続の通信では、攻撃者は、接続を維持するために他のHTTPヘッダーフィールドを入力して、パケットを定期的にサーバーに送信します。サーバーはヘッダーの終わりマーカーを予期しており、接続を長期間維持します。
- **Slow POST**: このタイプの攻撃は、次のいずれかの状況で発生します。
    - 攻撃者は、サーバーにデータを送信するHTTP POST要求を送信し、**Content-Length**フィールドをより大きな値に設定します。その後のペイロードtransisthmianでは、攻撃者は接続を維持するために毎回少数のデータを送信します。サーバーは、接続を解放することなく、攻撃者からのペイロードデータを期待し続けます。
    - 攻撃者は、チャンク転送エンコーディングでHTTPパケットを送信します。HTTPパケットが長さ0のチャンクで終了していない場合、サーバーは接続を解放せずに攻撃者からのペイロードデータを予期しています。

HTTP低速攻撃を防止するには、HTTP低速攻撃の検出と防止を設定します。この機能は、このような攻撃を検出し、攻撃者のアドレスをブラックリストに追加します。送信元アドレスがブラックリストにあるパケットはドロップされます。

ブラックリスト機能

IPブラックリスト

IPブラックリスト機能は、送信元または宛先IPアドレスを使用してパケットをフィルタリングします。

- **Source IP blacklist**: パケットの送信元IPアドレスが送信元IPブラックリストエントリと一致する場合、パケットをブロックします。
- **Destination IP blacklist**: パケットの宛先IPアドレスが宛先IPブラックリストエントリと一致する場合、パケットをブロックします。

ACLベースのパケットフィルタリングと比較して、IPブラックリストフィルタリングは単純であり、より高速で効果的なスクリーニングを提供します。

アドレスオブジェクトグループのブラックリスト

アドレス オブジェクト グループ ブラックリスト機能は、パケットをアドレス オブジェクト グループ別にフィルタリングする攻撃防御方式です。アドレス オブジェクト グループ ブラックリスト機能は、アドレス オブジェクト グループ機能と併用する必要があります。アドレス オブジェクト グループは、IPアドレスオブジェクトのセットです。アドレス オブジェクト グループの詳細については、「オブジェクト グループの設定」を参照してください。アドレス オブジェクト グループ ブラックリスト フィルタリングは、IPブラックリスト フィルタリングと比較して、サブネットのアクセスコントロールを実行し、フィルタリングのユーザービリティを向上させます。

アドレスオブジェクトグループのホワイトリスト

アドレス オブジェクト グループ ホワイトリスト機能は、ホワイトリストに記載されたアドレス オブジェクト グループからのパケットを攻撃検出から免除します。ホワイトリストに記載されたアドレス オブジェクト グループからのパケットは、攻撃パケットであるかどうかにかかわらず直接転送されます。アドレス オブジェクト グループ ホワイトリスト機能は、アドレス オブジェクト グループ機能と一緒に使用する必要があります。アドレス オブジェクト グループは、IPアドレス オブジェクトのセットです。アドレス オブジェクト グループの詳細については、「オブジェクト グループの設定」を参照してください。

クライアントの検証

TCPクライアントの確認

TCPクライアント検証機能は、次のフラッド攻撃からTCPサーバーを保護します。

- SYN
- ACK
- SYN-ACK。
- FIN
- RST

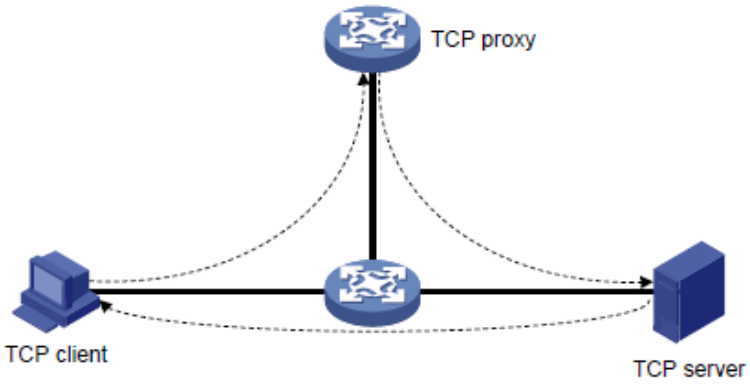
TCPクライアント検証機能は、デバイス上でTCPプロキシをイネーブルにします。TCPクライアント検証は、次のモードで動作できます。

- **Safe Reset**: TCP接続発信側からのパケットに限り、単方向TCPプロキシをイネーブルにします。攻撃はクライアントから発生することが多いため、ほとんどのシナリオでは単方向TCPプロキシ

で十分です。

図1に示すように、TCPクライアントからのパケットがプロキシデバイスを通過するが、サーバーからのパケットは通過しない場合、セーフリセットモードのみを使用できます。

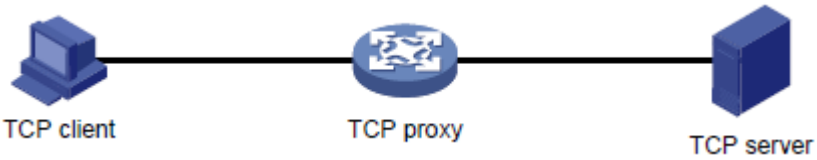
図1 セーフリセットモードの適用



- **SYN cookie**: TCPクライアントおよびサーバーの双方向TCPプロキシをイネーブルにします。

  図2に示すように、クライアントおよびサーバーからのパケットがTCPプロキシデバイスを通過する場合、セーフリセットまたはSYNクッキーのいずれかを使用できます。

  図2 セーフリセット/SYN Cookieモードの適用



セーフリセットモードのTCPプロキシ

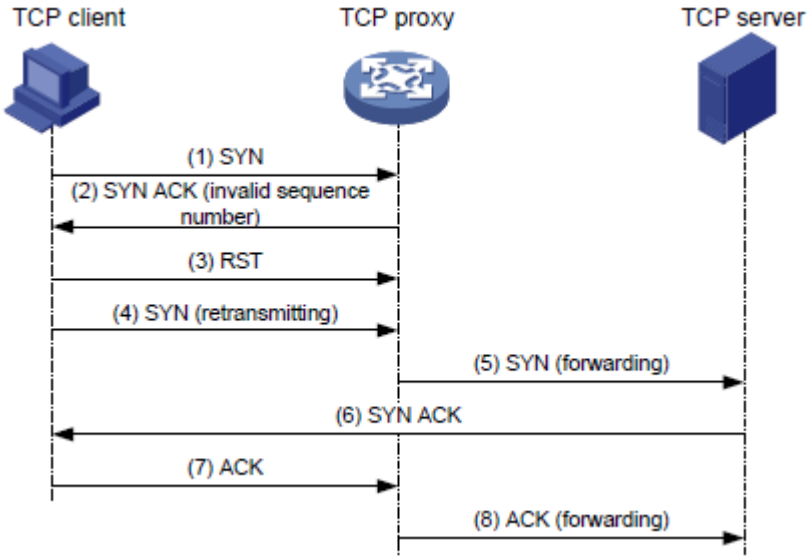
図3に示すように、安全リセットモードは以下のように機能する。

1. 保護されたサーバー宛てのSYNパケットを受信した後、TCPプロキシは、無効なシーケンス番号を持つSYN ACKパケットを送り返します。
2. TCPプロキシがクライアントからRSTパケットを受信した場合、クライアントは正当であると確認されます。
3. TCPプロキシは、クライアントのIPアドレスを信頼できるIPリストに追加します。クライアントは接続を再開し、TCPプロキシはTCPパケットをサーバーに直接転送します。

セーフリセットモードでは、TCPクライアントがTCPプロトコルスイートに準拠している必要があります。クライアントがTCPプロトコルスイートに準拠していない場合、TCPプロキシは正当なクライアントのサーバーへのアクセスを拒否します。

クライアント検証では、TCP接続の確立に通常のTCP接続の確立よりも多くの時間がかかります。

図3 セーフリセットモードのTCPプロキシ



SYNクッキーモードのTCPプロキシ

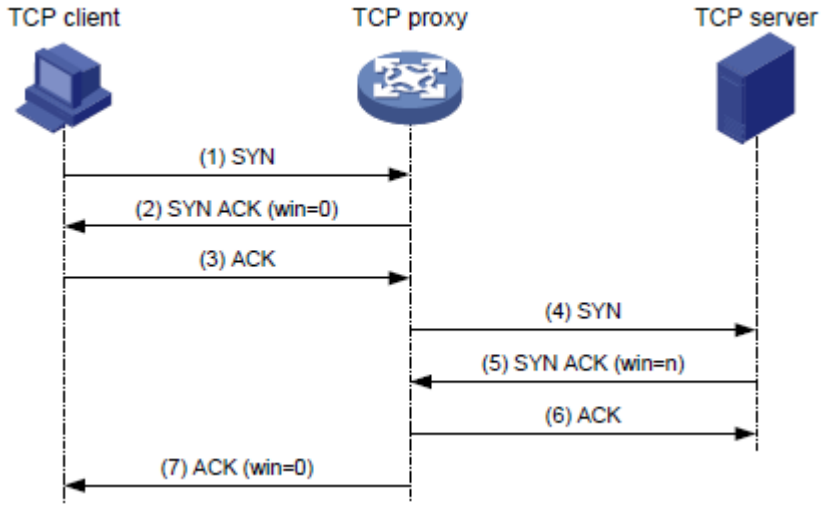
図4に示すように、SYNクッキーモードでは、次のように2つのTCP接続を確立する必要があります。

1. TCPプロキシは、クライアントから保護されたサーバーへのSYNパケットを受信した後、ウィンドウサイズ0のSYN ACKパケットを返信します。クライアントがACKパケットで応答した場合、クライアントは正当であると検証されます。プロキシデバイスは、クライアントとのTCP接続を確立します。
2. TCPプロキシデバイスは、異なるウィンドウサイズを持つ新しい3ウェイハンドシェイクを介してサーバーとの接続を確立します。この接続では、クライアントとプロキシデバイス間の接続とは異なるシーケンス番号が使用されます。

SYNクッキーモードでは、TCPプロキシはクライアントと通信するサーバープロキシであり、クライアントプロキシはサーバーと通信します。次の要件が満たされている場合、このモードを選択します。

- TCPプロキシデバイスは、保護対象サーバーの入力および出力を通過するキーパスに展開されます。
- クライアントとサーバーの間で交換されるすべてのパケットは、TCPプロキシデバイスを通過します。

図4 SYNクッキーモードのTCPプロキシ



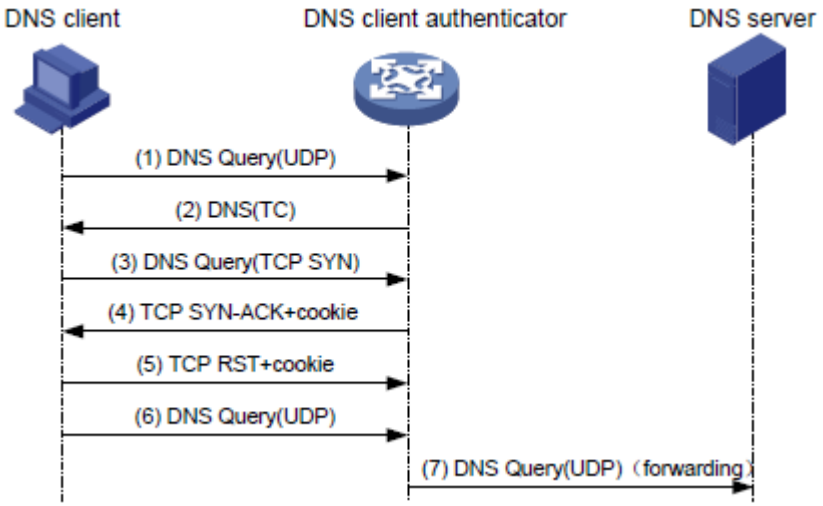
DNSクライアントの確認

DNSクライアント検証機能は、DNSフラッド攻撃からDNSサーバーを保護します。DNSクライアントからDNSサーバーへのパケットが通過するデバイス上で設定されます。DNSクライアント検証機能が設定されたデバイスは、DNSクライアント オーセンティケータと呼ばれます。

図5に示すように、DNSクライアントの検証は次のように機能します。

1. 保護されたサーバー宛てのUDP DNSクエリーを受信すると、DNSクライアント認証はDNS Truncate(TC)パケットで応答します。DNS Truncateパケットでは、クライアントがTCPパケットでクエリーを開始する必要があります。
2. オーセンティケータがクライアントからポート53へのTCP SYNパケットでDNSクエリーを受信すると、オーセンティケータは誤ったシーケンス番号を含むSYN-ACKパケットで応答します。
3. オーセンティケータがクライアントからRSTパケットを受信すると、オーセンティケータはクライアントが正規であることを確認します。
4. オーセンティケータは、クライアントのIPアドレスを信頼できるIPリストに追加し、信頼できるクライアントの以降のパケットをサーバーに転送します。

図5 DNSクライアントの検証プロセス



DNSクライアントの検証機能では、クライアントが標準のTCP/IPプロトコル群とDNSプロトコルを使用する必要があります。標準以外のプロトコルを使用する正当なクライアントは、DNSクライアント認証システムによって不正であると検証されます。

クライアント検証では、最初のDNS解決に通常のDNS解決よりも多くの時間がかかります。

DNS応答の確認

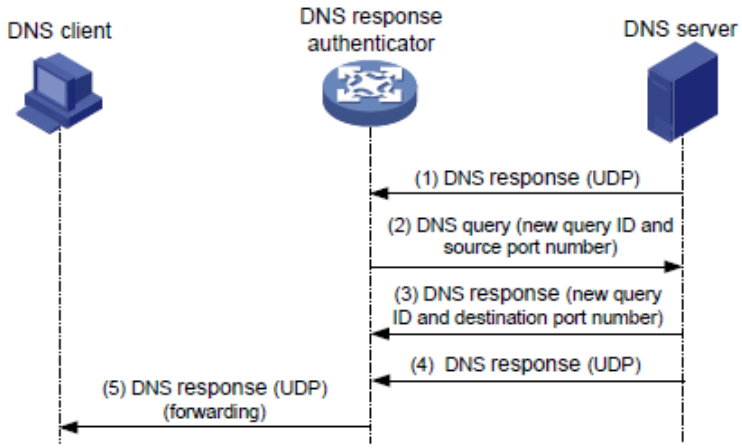
DNS応答検証機能は、DNS応答フラッド攻撃からDNSクライアントを保護します。これは、DNSサーバーからDNSクライアントへのパケットが通過するデバイス上で設定されます。DNS応答検証機能が設定されたデバイスは、DNS応答オーセンティケータと呼ばれます。

図6に示すように、DNS応答検証は次のように機能します。

1. 保護されたクライアント宛てのUDP DNS応答を受信すると、DNS応答オーセンティケータは、ローカルで生成されたクエリーIDとポート番号を含むDNSクエリーパケットを送り返します。
2. DNSクエリーを受信した後、有効なDNSサーバーは、新しいクエリーIDと宛先ポートを含むDNS応答で応答します。
3. DNS応答オーセンティケータは、応答内のクエリーIDと宛先ポートを確認します。クエリーIDと

宛先ポートが、オーセンティケータが送信したクエリーIDとポート番号と同じである場合、DNSサーバーは確認に合格します。オーセンティケータは、サーバーからの後続のパケットを転送します。

図6 DNS応答の検証プロセス



HTTPクライアントの検証

HTTPクライアント検証機能は、HTTPフラッド攻撃からHTTPサーバーを保護します。HTTPクライアントからHTTPサーバーへのHTTP GETまたはPOST要求パケットが通過するデバイス上で設定されます。HTTPクライアント検証機能が設定されたデバイスは、HTTPクライアント オーセンティケータと呼ばれます。

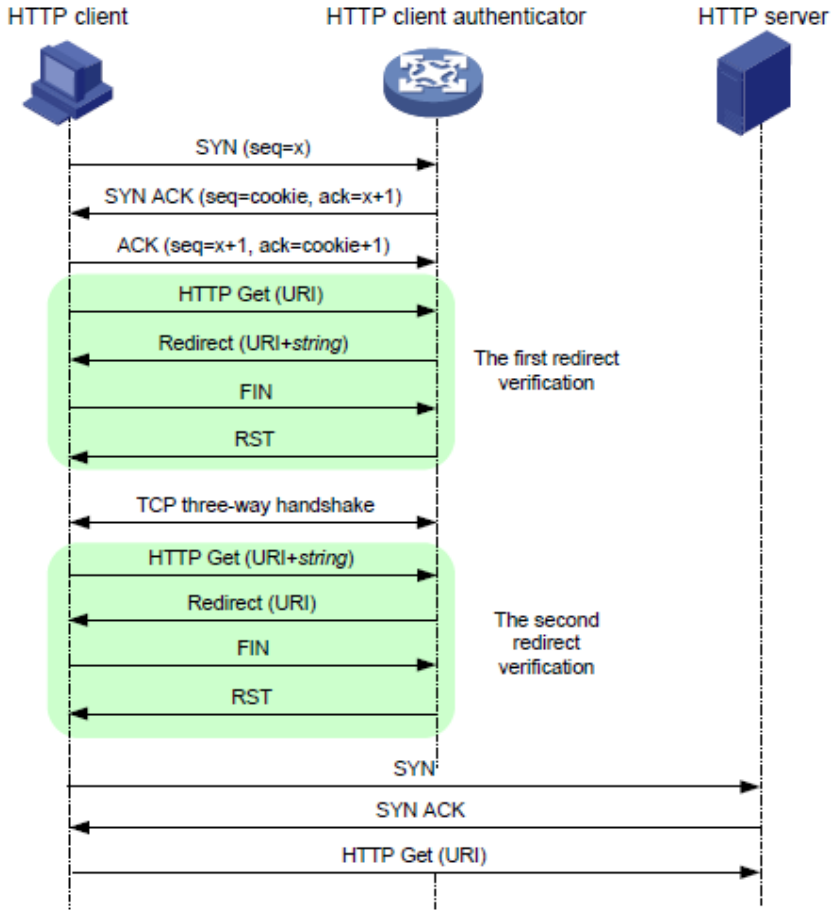
GET要求ベースの検証

図7に示すように、HTTPクライアン トオーセンティケータは、次のようにHTTP GET要求を使用してHTTPクライアントを検証します。

1. HTTPクライアント認証は、保護されたHTTPサーバー宛てのSYNパケットを受信すると、SYN cookieモードでTCPクライアントの検証を実行します。クライアントがTCPクライアントの検証に合格すると、クライアントと認証の間にTCP接続が確立されます。TCPクライアントの検証の詳細については、「TCPクライアントの検証」を参照してください。
2. オーセンティケータは、クライアントからHTTP GETパケットを受信すると、最初のリダイレクト検証を実行します。オーセンティケータは、クライアント情報を記録し

   HTTPリダイレクトパケット。HTTPリダイレクトパケットにはリダイレクトURIが含まれ、クライアントがTCP接続を終了する必要があります。
3. HTTPリダイレクトパケットを受信した後、クライアントはTCP接続を終了し、オーセンティケータとの新しいTCP接続を確立します。
4. オーセンティケータは、HTTP GETパケットを受信すると、2回目のリダイレクト検証を実行します。オーセンティケータは、次の情報を検証します。
   - クライアントは最初のリダイレクション検証に合格しました。
   - HTTP GETパケットのURIは、リダイレクトURIです。
5. クライアントが2回目のリダイレクト検証に合格した場合、オーセンティケータはそのIPアドレスを信頼できるIPリストに追加し、リダイレクトパケットに応答します。リダイレクトパケットには、クライアントが最初に伝送したURIが含まれ、クライアントにTCP接続の終了を要求します。
6. オーセンティケータは、信頼できるクライアントの以降のパケットをサーバーに直接転送します。

図7 HTTPクライアントの検証プロセス

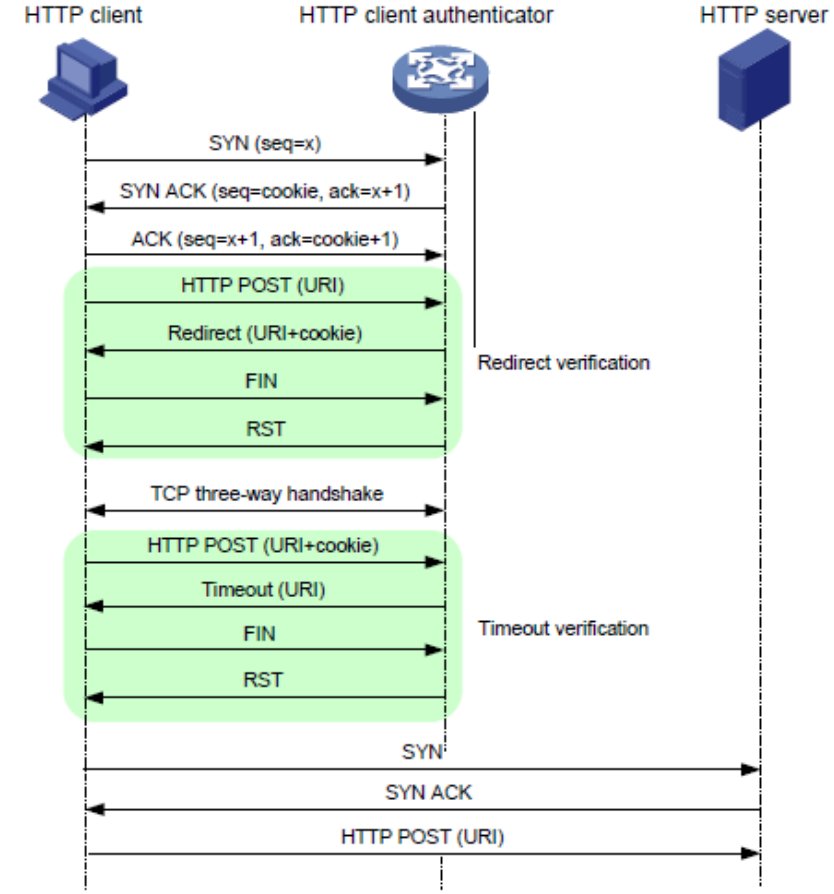


POST要求ベースの検証

図8に示すように、HTTPクライアント認証はHTTP POST要求を使用して、次のようにHTTPクライアントを検証します。

1. HTTPクライアントオーセンティケータは、保護されたHTTPサーバー宛てのSYNパケットを受信すると、SYN CookieモードでTCPクライアントの検証を実行します。クライアントがTCPクライアントの検証に合格すると、クライアントとオーセンティケータの間にTCP接続が確立されます。TCPクライアントの検証の詳細は、「TCPクライアントの検証」を参照してください。
2. オーセンティケータは、クライアントからHTTP POST要求を受信すると、リダイレクト検証を実行します。オーセンティケータはクライアント情報を記録し、HTTPリダイレクトパケットで応答します。HTTPリダイレクトパケットには、リダイレクトURIとSet-Cookieヘッダーが含まれ、クライアントにTCP接続の終了を要求します。
3. HTTPリダイレクトパケットを受信した後、クライアントはTCP接続を終了し、オーセンティケータとの新しいTCP接続を確立します。
4. オーセンティケータは、HTTP POST要求を受信すると、タイムアウト検証を実行します。オーセンティケータは、次の情報を検証します。
   - クライアントはリダイレクションの検証に合格しました。
   - HTTP POST要求に有効なCookieが含まれています。
5. クライアントがタイムアウトの検証に合格すると、オーセンティケータはそのIPアドレスを信頼できるIPリストに追加し、HTTPタイムアウトパケットで応答します。タイムアウトパケットには、クライアントが最初に伝送したURIが含まれ、クライアントにTCP接続の終了を要求します。
6. オーセンティケータは、信頼できるクライアントの以降のパケットをサーバーに直接転送します。

図8 POST要求ベースの検証プロセス



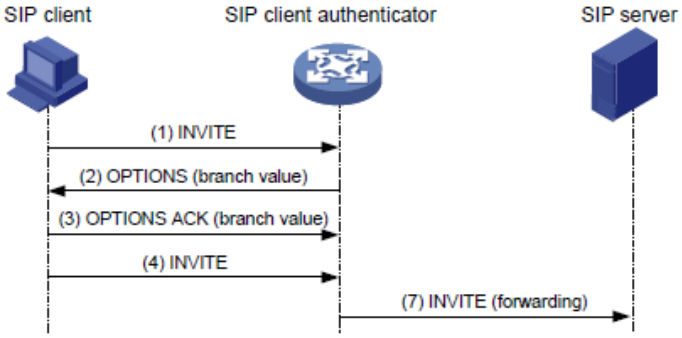
SIPクライアントの確認

SIPクライアント検証機能は、SIPフラッド攻撃からSIPサーバーを保護します。SIPクライアントからSIPサーバーへのSIP INVITE要求パケットが通過するデバイス上で設定されます。SIPクライアント検証機能が設定されたデバイスは、SIPクライアントオーセンティケータと呼ばれます。

図9に示すように、SIPクライアント検証は次のように機能します。

1. SIPクライアントオーセンティケータは、保護対象サーバー宛てのUDP INVITEパケットを受信すると、ブランチ値を含むOPTIONSパケットを返信します。
2. OPTIONSパケットを受信した後、クライアントはSIPクライアントオーセンティケータにOPTIONS ACKを送信します。
3. OPTIONS ACKを受信すると、SIPクライアントオーセンティケータはOPTIONS ACKのブランチ値を確認します。
   - OPTIONS ACKの分岐値が、SIPクライアント認証者が送信したOPTIONSパケットの分岐値と同じである場合、クライアントは検証に合格します。認証者は、クライアントからの後続のパケットを転送します。
   - OPTIONS ACKの分岐値が、SIPクライアント認証者が送信したOPTIONSパケットの分岐値と異なる場合、クライアントは検証に失敗します。認証者はクライアントからのパケットをドロップします。

図9 SIPクライアントの検証プロセス



攻撃の検出と防御のタスクを一目で把握

攻撃の検出と防御を設定するには、次のタスクを実行します。

1. 攻撃防御ポリシーの設定と適用
   a. 攻撃防御ポリシーの作成
   b. 攻撃防御ポリシーの設定必要に応
      じて、次のタスクを選択します。
      - 単一パケット攻撃防御ポリシーの設定
      - スキャン攻撃防御ポリシーの設定
      - フラッドアタック防御ポリシーの設定
      - HTTP低速攻撃防御ポリシーの設定
   c. (任意)攻撃検出免除の設定
   d. 攻撃防御ポリシーの適用必要に応
      じて、次のタスクを選択します。
      - インターフェイスへの攻撃防御ポリシーの適用
      - デバイスへの攻撃防御ポリシーの適用
2. (任意)単一パケット攻撃イベントのログ非集約のイネーブル化
3. (任意)トップ攻撃統計情報ランキング機能のイネーブル化
4. (省略可能)クライアント検証の構成
   この機能は、フラッド攻撃防御ポリシーとは別に使用することも、併用することもできます。
   - TCPクライアント検証の設定
   - DNSクライアント検証の構成
   - HTTPクライアント検証の設定
   - SIPクライアント検証の設定
5. (任意)ブラックリスト機能の設定
   この機能は、個別に使用することも、スキャン攻撃防御ポリシーと組み合わせて使用することもできます。
   - IPブラックリスト機能の設定
   - アドレスオブジェクトグループのブラックリストの設定
6. (任意)アドレスオブジェクトグループホワイトリストの設定
7. (任意)ログイン攻撃防止機能の設定通常、この機能は
   別に使用されます。

- ログイン攻撃防止の設定
- ログイン遅延の有効化

攻撃防御ポリシーの設定と適用

攻撃防御ポリシーの作成

このタスクについて

攻撃防御ポリシーには、一連の攻撃検出および防御設定が含まれています。

検出シグニチャや保護アクションなどの攻撃防御設定を設定するには、最初に攻撃防御ポリシーを作成し、そのビューを入力する必要があります。

制限事項およびガイドライン

△注意:

攻撃防御をトリガーするためのデフォルトのしきい値は、ネットワークに適していない場合があります。実際のアプリケーションのシナリオに従って、適切なしきい値を設定してください。しきい値が小さいと、インターネットまたはWebページのアクセス速度に影響する場合があります。しきい値が大きいと、ネットワークが攻撃に対して脆弱になる場合があります。

手順

1. システムビューに入ります。

   system-view
2. 攻撃防御ポリシーを作成し、そのビューを入力します。

   attack-defense policy *policy-name*

単一パケット攻撃防御ポリシーの設定

このタスクについて

外部ネットワークに接続されているインターフェイスに単一パケット攻撃防御ポリシーを適用します。

単一パケット攻撃検出は、パケットシグニチャに基づいて着信パケットを検査します。攻撃パケットが検出された場合、デバイスは次のアクションを実行できます。

- 出力ログ(デフォルトのアクション)。
- 攻撃パケットをドロップします。

また、何のアクションも実行しないようにデバイスを設定することもできます。

制限事項およびガイドライン

loggingキーワードを指定すると、攻撃検出および防御モジュールは、単一パケット攻撃イベントをログに記録し、ログメッセージを情報センターに送信できます。

インフォメーションセンターでは、ログメッセージのフィルタリングと出力先を含む出力ルールを設定できます。

単一パケットの攻撃ログは、コンソールと監視端末以外に出力できますが、コンソールや監視端末を出力先に設定しても、出力先の設定は有効になりません。

デバイスに保存されている単一パケット攻撃ログを表示するには、display logbufferコマンドを使用します。ログバッファへのログ出力(デフォルトではイネーブル)をディセーブルにしないようにしてください。

Information Centerの設定の詳細については、『Network Management and Monitoring

Configuration Guide』を参照してください。

手順

1. システムビューに入ります。

   system-view

2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*

3. 特定の単一パケット攻撃タイプのシグニチャ検出を設定し、攻撃に対するアクションを指定します。

   - 既知の単一パケット攻撃に対するシグニチャ検出を設定し、攻撃に対するアクションを指定します。
     signature detect { fraggle | fragment | impossible | land | large-icmp | large-icmpv6 | smurf | snork | tcp-all-flags | tcp-fin-only | tcp-invalid-flags | tcp-null-flag | tcp-syn-fin | tiny-fragment | traceroute | udp-bomb | winnuke } action { { drop | logging } * | none }
     signature detect { ip-option-abnormal | ping-of-death | teardrop } action { drop | logging } *
   - ICMPパケット攻撃のシグニチャ検出を設定し、攻撃に対するアクションを指定します。
     signature detect icmp-type { *icmp-type-value* | address-mask-reply | address-mask-request | destination-unreachable | echo-reply | echo-request | information-reply | information-request | parameter-problem | redirect | source-quench | time-exceeded | timestamp-reply | timestamp-request } action { { drop | logging } * | none }
   - ICMPv6パケット攻撃のシグニチャ検出を設定し、攻撃に対するアクションを指定します。
     signature detect icmpv6-type { *icmpv6-type-value* | destination-unreachable | echo-reply | echo-request | group-query | group-reduction | group-report | packet-too-big | parameter-problem | time-exceeded } action { { drop | logging } * | none }
   - IPオプション攻撃のシグニチャ検出を設定し、攻撃に対するアクションを指定します。
     signature detect ip-option { *option-code* | internet-timestamp | loose-source-routing | record-route | route-alert | security | stream-id | strict-source-routing } action { { drop | logging } * | none }
   - IPv6拡張ヘッダー攻撃のシグニチャ検出を設定し、攻撃に対するアクションを指定します。
     signature detect ipv6-ext-header *ext-header-value* action { { drop | logging } * | none }
   - 異常なIPv6拡張ヘッダー攻撃に対するシグニチャ検出を設定し、攻撃に対するアクションを指定します。
     signature detect ipv6-ext-header-abnormal action { { drop | logging } * | none }
   - IPv6拡張ヘッダー超過攻撃のシグニチャ検出を設定し、攻撃に対するアクションを指定します。
     signature detect ipv6-ext-header-exceed limit *limit-value* action { { drop | logging } * | none }

   デフォルトでは、シグニチャ検出は単一パケット攻撃に対して設定されていません。

4. (省略可能)安全なICMPまたはICMPv6パケットの最大長を設定します。

   signature { large-icmp | large-icmpv6 } max-length *length*

   デフォルトでは、安全なICMPまたはICMPv6パケットの最大長は4000バイトです。

5. (任意)特定のレベルの単一パケット攻撃に対するアクションを指定します。
   signature level { **high** | **info** | **low** | **medium** } **action** { { **drop** | **logging** }* | **none** }

   デフォルトのアクションは、情報レベルおよび低レベルの単一パケット攻撃のロギングです。

   デフォルトのアクションは、中レベルおよび高レベルの単一パケット攻撃に対するロギングおよび

ドロップです。

6. (任意)特定のレベルの単一パケット攻撃に対するシグニチャ検出をイネーブルにします。

   signature level **{** high **|** info **|** low **|** medium **}** detect

   デフォルトでは、シグニチャ検出はすべてのレベルの単一パケット攻撃に対してディセーブルになっています。

スキャン攻撃防御ポリシーの設定

このタスクについて

外部ネットワークに接続されているインターフェイスに、スキャン攻撃防御ポリシーを適用します。

スキャン攻撃検出は、ターゲットシステムへの接続の着信パケットレートを検査します。送信元が事前に定義されたしきい値以上のレートで接続を開始した場合、デバイスは次のアクションを実行できます。

- 出力ログ。
- 攻撃者のIPアドレスからの後続のパケットをドロップします。
- 攻撃者のIPアドレスをIPブラックリストに追加します。

IPスイープ攻撃とポートスキャン攻撃に対してロギングが指定されている場合、IPスイープ攻撃とポートスキャン攻撃の両方のしきい値に到達すると、IPスイープ攻撃のログだけが出力されます。

制限事項およびガイドライン

攻撃者をブラックリストに載せるには、ブラックリスト機能をグローバルに、または防御ポリシーが適用されるインターフェイスでイネーブルにする必要があります。ブラックリストの詳細については、「IPブラックリスト機能の設定」を参照してください。

loggingキーワードを指定すると、攻撃検出および防御モジュールは、スキャン攻撃イベントをログに記録し、ログメッセージを情報センターに送信できます。

インフォメーションセンターでは、ログメッセージのフィルタリングと出力先を含む出力ルールを設定できます。

情報センターは、スキャン攻撃ログをコンソールと監視端末以外に出力することができますが、コンソールや監視端末を出力先として設定した場合、出力先の設定は有効になりません。

デバイスに保存されているスキャン攻撃ログを表示するには、**display logbuffer**コマンドを使用します。デフォルトでイネーブルになっているログバッファへのログ出力をディセーブルにしないようにしてください。

Information Centerの設定の詳細については、『Network Management and Monitoring Configuration Guide』を参照してください。

手順

1. システムビューに入ります。

   system-view

2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*

3. スキャン攻撃検出を設定します。
   scan detect level { { **high** | **low** | **medium** } | **user-defined** { **port-scan-threshold** *threshold-value* | **ip-sweep-threshold** *threshold-value* } * **period** *period-value* } **action** { { **block-source timeout** *minutes* | **drop** } | **logging** } *

   デフォルトでは、スキャン攻撃検出は設定されていません。

フラッドアタック防御ポリシーの設定

このタスクについて

内部サーバーを保護するために、外部ネットワークに接続されているインターフェイスにフラッド攻撃防御ポリシーを適用します。

フラッド攻撃検出は、内部サーバーへの接続が開始されるレートを監視します。デバイスは、次のフ

ラッド攻撃防御タイプをサポートしています。

- **Source-based flood attack prevention**: ソースIPベースのフラッド攻撃防止でのパケットの受信レートを監視します。
  IPアドレスから発信されたパケットの受信レートがしきい値以上になり続けると、デバイスは防御ステートに入り、指定された防御アクションを実行します。サポートされている防御アクションには、このIPアドレスから発信されたパケットのロギングおよびドロップが含まれます。レートが無音状態のしきい値(しきい値の4分の3)を下回ると、デバイスは攻撃検出ステートに戻ります。
- **Destination-based flood attack prev**: 宛先IPごとにパケットの受信レートを監視します。IPアドレス宛のパケットの受信レートが維持される場合
  しきい値に達するか、しきい値を超えると、デバイスは防御状態に入り、指定されたアクションを実行します。サポートされる防御アクションには、ロギング、このIPアドレスを宛先とする後続のパケットのドロップ、およびクライアント検証が含まれます。レートが無音状態のしきい値(しきい値の4分の3)を下回ると、デバイスは攻撃検出状態に戻ります。

適切なしきい値を設定すると、攻撃を効果的に防ぐことができます。フラッド攻撃防止をトリガーするグローバルなしきい値が低すぎると、フォールスポジティブが発生し、パフォーマンスの低下やパケット損失が発生する可能性があります。グローバルなしきい値が高すぎると、フォールスネガティブが発生し、ネットワークが無防備になる可能性があります。したがって、デバイスがグローバルなしきい値を自動的に学習するために、しきい値学習機能を有効にすることをお勧めします。この機能により、デバイスはネットワーク内のトラフィックフローに基づいて、次のようにグローバルなしきい値を学習できます。

1. ネットワーク内のパケット受信レートを監視します。
2. しきい値学習期間内に学習されたピークレートに基づいて、グローバルしきい値を計算します。

学習したしきい値を手動で適用するか、学習したしきい値を自動的に適用するようにデバイスを設定するかを選択できます。

しきい値学習機能には、次のモードがあります。

- **One-time learning**: デバイスは、しきい値学習を1回だけ実行します。
- **Periodic learning**: は一定の間隔でしきい値の学習を実行します。最近学習したしきい値が常に有効になります。

フラッド攻撃の検出と防止のための制限とガイドライン

デバイスに複数のサービスカードがある場合、設定したグローバルトリガーしきい値は各サービスカードで有効になります。デバイスのグローバルトリガーしきい値は、設定した値にサービスカードの数量を乗算した積です。

特定のIPアドレスに対してフラッド攻撃の検出および防御を設定できます。現在のソフトウェアバージョンでIPアドレスの指定をサポートしているのは、宛先ベースのフラッド攻撃防御だけです。非特定IPアドレスの場合、デバイスはグローバル攻撃防御設定を使用します。

loggingキーワードを指定すると、攻撃検出および防御モジュールは、フラッド攻撃イベントをログに記録し、ログメッセージを情報センターに送信できます。

インフォメーションセンターでは、ログメッセージのフィルタリングと出力先を含む出力ルールを設定できます。

情報センターは、コンソールおよび監視端末以外の宛先にフラッド攻撃ログを出力できますが、コンソールまたは監視端末を出力先として設定した場合、出力先の設定は有効になりません。

デバイスに保存されているフラッド攻撃ログを表示するには、**display logbuffer**コマンドを使用します。ログバッファへのログ出力(デフォルトではイネーブル)をディセーブルにしないようにしてください。

Information Centerの設定の詳細については、『Network Management and Monitoring Configuration Guide』を参照してください。

SYNフラッド攻撃防御ポリシーの設定

1. システムビューに入ります。

   system-view
2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*
3. グローバルSYNフラッド攻撃検出をイネーブルにします。

   syn-flood detect non-specific

   デフォルトでは、グローバルSYNフラッド攻撃検出はディセーブルです。
4. ソースベースのSYNフラッド攻撃防止をトリガーするためのグローバルしきい値を設定します。

   syn-flood source-threshold *threshold-value*

   デフォルト設定は10000です。
5. 宛先ベースのSYNフラッド攻撃防止をトリガーするためのグローバルしきい値を設定します。

   syn-flood threshold *threshold-v*

   デフォルト設定は10000です。
6. SYNフラッド攻撃に対するグローバルアクションを指定します。

   syn-flood action { **client-verify** | **drop** | **logging** } *

   デフォルトでは、SYNフラッド攻撃に対するグローバルアクションは指定されていません。
7. IPアドレス固有のSYNフラッド攻撃検出を設定します。

   syn-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } **vpn-instance** *vpn-instance-name* **threshold** *threshold-value* **action** { { **client-verify** | **drop** | **logging** } * | **none** }

   デフォルトでは、IPアドレス固有のSYNフラッド攻撃検出は設定されていません。

ACKフラッド攻撃防御ポリシーの設定

1. システムビューに入ります。

   system-view
2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*
3. グローバルACKフラッド攻撃検出をイネーブルにします。

   ack-flood detect non-specific

   デフォルトでは、グローバルACKフラッド攻撃検出はディセーブルになっています。
4. 送信元ベースのACKフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

   ack-flood source-threshold *threshold-value*

   デフォルト設定は40000です。
5. 宛先ベースのACKフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

   ack-flood threshold *threshold-value*

   デフォルト設定は40000です。
6. ACKフラッド攻撃に対するグローバルアクションを指定します。

ack-flood action **{** client-verify **|** drop **|** logging **} ***

デフォルトでは、ACKフラッド攻撃に対するグローバルアクションは指定されていません。

7. IPアドレス固有のACKフラッド攻撃検出を設定します。
ack-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } **vpn-instance** *vpn-instance-name* **threshold** *threshold-value* **action** { { **client-verify** | **drop** | **logging** } * | **none** }

デフォルトでは、IPアドレス固有のACKフラッド攻撃検出は設定されていません。

SYN-ACKフラッド攻撃防御ポリシーの設定

1. システムビューに入ります。

system-view

2. 攻撃防御ポリシービューに入ります。

attack-defense policy *policy-name*

3. グローバルSYN-ACKフラッド攻撃検出をイネーブルにします。

syn-ack-flood detect non-specific

デフォルトでは、グローバルSYN-ACKフラッド攻撃検出はディセーブルです。

4. 送信元ベースのSYN-ACKフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

syn-ack-flood source-threshold *threshold-value*

デフォルト設定は10000です。

5. 宛先ベースのSYN-ACKフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

syn-ack-flood threshold *threshold-value*

デフォルト設定は10000です。

6. SYN-ACKフラッド攻撃に対するグローバルアクションを指定します。

syn-ack-flood action **{** client-verify **|** drop **|** logging **}***

デフォルトでは、SYN-ACKフラッド攻撃に対するグローバルアクションは指定されていません。

7. IPアドレス固有のSYN-ACKフラッド攻撃検出を設定します。
syn-ack-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } **vpn-instance** *vpn-instance-name* **threshold** *threshold-value* **action** { { **client-verify** | **drop** | **logging** } * | **none** }

デフォルトでは、IPアドレス固有のSYN-ACKフラッド攻撃検出は設定されていません。

FINフラッド攻撃防御ポリシーの設定

1. システムビューに入ります。

system-view

2. 攻撃防御ポリシービューに入ります。

attack-defense policy *policy-name*

3. グローバルFINフラッド攻撃検出をイネーブルにします。

fin-flood detect non-specific

デフォルトでは、グローバルFINフラッド攻撃検出はディセーブルになっています。

4. ソースベースのFINフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

fin-flood source-threshold *threshold-v*

デフォルト設定は10000です。

5. 宛先ベースのFINフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

fin-flood threshold *threshold-value*

デフォルト設定は10000です。

6. FINフラッド攻撃に対するグローバルアクションを指定します。

   fin-flood action **{** client-verify **|** drop **|** logging **} ***

   デフォルトでは、FINフラッド攻撃に対するグローバルアクションは指定されていません。

7. IPアドレス固有のFINフラッド攻撃検出を設定します。

   fin-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } **vpn-instance** *vpn-instance-name* **threshold** *threshold-value* **action** { { **client-verify** | **drop** | **logging** } * | **none** }

   デフォルトでは、IPアドレス固有のFINフラッド攻撃検出は設定されていません。

RSTフラッド攻撃防御ポリシーの設定

1. システムビューに入ります。

   system-view

2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*

3. グローバルRSTフラッド攻撃検出をイネーブルにします。

   rst-flood detect non-specific

   デフォルトでは、グローバルRSTフラッド攻撃検出はディセーブルです。

4. ソースベースのRSTフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

   rst-flood source-threshold *threshold-value*

   デフォルト設定は10000です。

5. 宛先ベースのRSTフラッド攻撃防止をトリガーするためのグローバルしきい値を設定します。

   rst-flood threshold *threshold-value*

   デフォルト設定は10000です。

6. RSTフラッド攻撃に対するグローバルアクションを指定します。

   rst-flood action **{** client-verify **|** drop **|** logging **} ***

   デフォルトでは、RSTフラッド攻撃に対するグローバルアクションは指定されていません。

7. IPアドレス固有のRSTフラッド攻撃検出を設定します。

   rst-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } **vpn-instance** *vpn-instance-name* **threshold** *threshold-value* **action** { { **client-verify** | **drop** | **logging** } * | **none** }

   デフォルトでは、IPアドレス固有のRSTフラッド攻撃検出は設定されていません。

ICMPフラッド攻撃防御ポリシーの設定

1. システムビューに入ります。

   system-view

2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*

3. グローバルICMPフラッド攻撃検出をイネーブルにします。

   icmp-flood detect non-specific

   デフォルトでは、グローバルICMPフラッド攻撃検出はディセーブルになっています。

4. 送信元ベースのICMPフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

   icmp-flood source-threshold *threshold-value*

   デフォルト設定は10000です。

5. 宛先ベースのICMPフラッド攻撃防止をトリガーするためのグローバルしきい値を設定します。

   icmp-flood threshold *threshold-value*

   デフォルト設定は10000です。

6. ICMPフラッド攻撃に対するグローバルアクションを指定します。

   icmp-flood action **{** drop **|** logging **} ***

   デフォルトでは、ICMPフラッド攻撃に対するグローバルアクションは指定されていません。

7. IPアドレス固有のICMPフラッド攻撃検出を設定します。

   icmp-flood detect ip *ip-address* **vpn-instance** *vpn-instance-name* **threshold** *threshold-value* **action** { { **drop** | **logging** } * | **none** }

   デフォルトでは、IPアドレス固有のICMPフラッド攻撃検出は設定されていません。

ICMPv6フラッド攻撃防御ポリシーの設定

1. システムビューに入ります。

   system-view

2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*

3. グローバルなICMPv6フラッド攻撃検出を有効にします。

   icmpv6-flood detect non-specific

   デフォルトでは、グローバルICMPv6フラッド攻撃検出は無効になっています。

4. 送信元ベースのICMPv6フラッド攻撃防止をトリガーするグローバルしきい値を設定します。

   icmpv6-flood source-threshold *threshold-value*

   デフォルト設定は10000です。

5. 宛先ベースのICMPv6フラッド攻撃防止をトリガーするグローバルしきい値を設定します。

   icmpv6-flood threshold *threshold-value*

   デフォルト設定は10000です。

6. ICMPv6フラッド攻撃に対するグローバルアクションを指定します。

   icmpv6-flood action **{** drop **|** logging **} ***

   デフォルトでは、ICMPv6フラッド攻撃に対するグローバルアクションは指定されていません。

7. IPアドレス固有のICMPv6フラッド攻撃検出を設定します。

   icmpv6-flood detect ipv6 *ipv6-address* [**vpn-instance** *vpn-instance-name*] [**threshold** *threshold-value*] [**action** { { **drop** | **logging** } * | **none** }]

   デフォルトでは、IPアドレス固有のICMPv6フラッド攻撃検出は設定されていません。

UDPフラッド攻撃防御ポリシーの設定

1. システムビューに入ります。

   System-view

2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*

3. グローバルUDPフラッド攻撃検出をイネーブルにします。

   udp-flood detect non-specific

   デフォルトでは、グローバルUDPフラッド攻撃検出はディセーブルになっています。

4. 送信元ベースのUDPフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

   udp-flood source-threshold *threshold-value*

   デフォルト設定は10000です。

5. 宛先ベースのUDPフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

   udp-flood threshold *threshold-value*

   デフォルト設定は10000です。

6. UDPフラッド攻撃に対するグローバルアクションを指定します。

   udp-flood action **{** drop **|** logging **} ***

   デフォルトでは、UDPフラッド攻撃に対するグローバルアクションは指定されていません。

7. IPアドレス固有のUDPフラッド攻撃検出を設定します。

   udp-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } **vpn-instance** *vpn-instance-name* **threshold** *threshold-value* **action** { { **drop** | **logging** } * | **none** }

   デフォルトでは、IPアドレス固有のUDPフラッド攻撃検出は設定されていません。

DNSフラッド攻撃防御ポリシーの設定

1. システムビューに入ります。

   system-view

2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*

3. グローバルDNSフラッド攻撃検出をイネーブルにします。

   dns-flood detect non-specific

   デフォルトでは、グローバルDNSフラッド攻撃検出はディセーブルです。

4. 送信元ベースのDNSフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

   dns-flood source-threshold *threshold-value*

   デフォルト設定は10000です。

5. 宛先ベースのDNSフラッド攻撃防止をトリガーするためのグローバルしきい値を設定します。

   dns-flood threshold *threshold-value*

   デフォルト設定は10000です。

6. (任意)DNSフラッド攻撃から保護するグローバルポートを指定します。

   dns-flood port *port-list*

   デフォルトでは、DNSフラッド攻撃防止によってポート53が保護されます。

7. DNSフラッド攻撃に対するグローバルアクションを指定します。

   dns-flood action **{** client-verify **|** drop **|** logging **} ***

   デフォルトでは、DNSフラッド攻撃に対するグローバルアクションは指定されていません。

8. IPアドレス固有のDNSフラッド攻撃検出を設定します。

   dns-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } **vpn-instance** *vpn-instance-name* **port** *port-list* **threshold** *threshold-value* **action** { { **client-verify** | **drop** | **logging** } * | **none** }

   デフォルトでは、IPアドレス固有のDNSフラッド攻撃検出は設定されていません。

DNS応答フラッド攻撃防御ポリシーの設定

1. システムビューに入ります。

   system-view

2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*

3. グローバルDNS応答フラッド攻撃検出をイネーブルにします。

   dns-reply-flood detect non-specific

   デフォルトでは、グローバルDNS応答フラッド攻撃検出はディセーブルです。

4. 送信元ベースのDNS応答フラッド攻撃防止をトリガーするグローバルしきい値を設定します。

   dns-reply-flood source-threshold *threshold-value*

デフォルト設定は10000です。

5. 宛先ベースのDNS応答フラッド攻撃防止をトリガーするためのグローバルしきい値を設定します。

   dns-reply-flood threshold *threshold-value*

   デフォルト設定は10000です。

6. (任意)DNS応答フラッド攻撃から保護するグローバルポートを指定します。

   dns-reply-flood port *port-list*

   デフォルトでは、DNS応答フラッド攻撃防止によってポート53が保護されます。

7. DNS応答フラッド攻撃に対するグローバルアクションを指定します。

   dns-reply-flood action **{** client-verify **|** drop **|** logging **} ***

   デフォルトでは、DNS応答フラッド攻撃に対するグローバルアクションは指定されていません。

8. IPアドレス固有のDNS応答フラッド攻撃検出を設定します。
   dns-reply-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } **vpn-instance** *vpn-instance-name* **port** *port-list* **threshold** *threshold-value* **action** { { **client-verify** | **logging** } * | **none** }

   デフォルトでは、IPアドレス固有のDNS応答フラッド攻撃検出は設定されていません。

HTTPフラッド攻撃防御ポリシーの設定

1. システムビューに入ります。

   system-view

2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*

3. グローバルHTTPフラッド攻撃検出をイネーブルにします。

   http-flood detect non-specific

   デフォルトでは、グローバルHTTPフラッド攻撃の検出はディセーブルです。

4. 送信元ベースのHTTPフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

   http-flood source-threshold *threshold-value*

   デフォルト設定は10000です。

5. 宛先ベースのHTTPフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

   http-flood threshold *threshold-value*

   デフォルト設定は10000です。

6. (任意)HTTPフラッド攻撃から保護するグローバルポートを指定します。

   http-flood port *port-list*

   デフォルトでは、HTTPフラッド攻撃防御はポート80を保護します。

7. HTTPフラッド攻撃に対するグローバルアクションを指定します。

   http-flood action **{** client-verify **|** drop **|** logging **} ***

   デフォルトでは、HTTPフラッド攻撃に対するグローバルアクションは指定されていません。

8. IPアドレス固有のHTTPフラッド攻撃検出を設定します。
   http-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } **vpn-instance** *vpn-instance-name* **port** *port-list* **threshold** *threshold-value* **action** { { **client-verify** | **drop** | **logging** } * | **none** }

   デフォルトでは、IPアドレス固有のHTTPフラッド攻撃検出は設定されていません。

SIPフラッド攻撃防御ポリシーの設定

1. システムビューに入ります。

system-view

2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*

3. グローバルSIPフラッド攻撃検出をイネーブルにします。

   sip-flood detect non-specific

   デフォルトでは、グローバルSIPフラッド攻撃検出はディセーブルです。

4. 送信元ベースのSIPフラッド攻撃防止をトリガーするグローバルしきい値を設定します。

   sip-flood source-threshold *threshold-value*

   デフォルト設定は10000です。

5. 宛先ベースのSIPフラッド攻撃防止をトリガーするためのグローバルしきい値を設定します。

   sip-flood threshold *threshold-value*

   デフォルト設定は10000です。

6. (任意)SIPフラッド攻撃から保護するグローバルポートを指定します。

   sip-flood port *port-list*

   デフォルトでは、SIPフラッド攻撃防止によってポート5060が保護されます。

7. SIPフラッド攻撃に対するグローバルアクションを指定します。

   sip-flood action **{** client-verify **|** drop **|** logging **} ***

   デフォルトでは、SIPフラッド攻撃に対するグローバルアクションは指定されていません。

8. IPアドレス固有のSIPフラッド攻撃検出を設定します。
   sip-flood detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } **vpn-instance** *vpn-instance-name* **port** *port-list* **threshold** *threshold-value* **action** { { **client-verify** | **drop** | **logging** } * | **none** }

   デフォルトでは、IPアドレス固有のSIPフラッド攻撃検出は設定されていません。

フラッド攻撃防止のしきい値学習の設定

1. システムビューに入ります。

   system-view

2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*

3. フラッド攻撃防止のしきい値学習機能をイネーブルにします。

   threshold-learn enable

   デフォルトでは、フラッド攻撃防止のしきい値学習機能はディセーブルになっています。

4. (任意)しきい値学習モードを設定します。

   - **one-time learning**モードを設定するには:
     threshold-learn mode once
   - **periodic learning**モードを設定するには:
     threshold-learn mode periodic

   デフォルトでは、ワンタイム学習モードが使用されます。

5. (任意)しきい値学習期間を設定します。

   threshold-learn duration *duration*

   デフォルトでは、しきい値学習期間は1440分です。

6. (任意)しきい値学習間隔を設定します。

   threshold-learn interval *interval*

   デフォルトでは、しきい値学習間隔は1440分です。1回限りの学習モードの場合は、この手順を省

略します。

7. (任意)しきい値学習許容値を設定します。
   threshold-learn tolerance-value *tolerance-value*
   デフォルトでは、しきい値学習許容値は50%です。
   学習したしきい値の自動適用がディセーブルになっている場合は、この手順を省略します。
8. (任意)学習したしきい値の自動適用をイネーブルにします。
   threshold-learn auto-apply enable
   デフォルトでは、学習したしきい値の自動適用はディセーブルになっています。
9. デバイスが学習した最新のしきい値を適用します。
   threshold-learn apply
   このコマンドは、学習したしきい値の自動適用がイネーブルになっている場合は有効になりません。

HTTP低速攻撃防御ポリシーの設定

このタスクについて

HTTP同時接続数が検出トリガーしきい値に達すると、デバイスはHTTP低速攻撃検出状態になります。デバイスが後でHTTP低速攻撃パケットを受信すると、HTTP低速攻撃が発生します。検出期間内にHTTP低速攻撃パケット数がしきい値を超えると、デバイスは防御アクションを実行します。

HTTP低速攻撃防御アクションには、攻撃イベントのロギングと攻撃者のIPアドレスのブラックリスト化が含まれます。

制限事項およびガイドライン

防御アクションとしてブラックリストを使用するには、ブラックリスト機能をイネーブルにします。

ベストプラクティスとしては、HTTP低速攻撃から保護するグローバルポートとしてポート80を指定します。http-slow-attack portコマンドを使用して他のポートを指定する場合は、これらのポートがHTTP通信に使用されていることを確認します。指定されたポートがHTTP通信に使用されていない場合は、非HTTP低速攻撃パケットの検査にデバイスリソースが浪費されます。

手順

1. システムビューに入ります。

   system-view
2. 攻撃防御ポリシービューに入ります。

   attack-defense policy *policy-name*
3. グローバルなHTTP低速攻撃検出をイネーブルにします。

   http-slow-attack detect non-specific

   デフォルトでは、グローバルHTTP低速攻撃検出はディセーブルになっています。
4. HTTP低速攻撃防止をトリガーするグローバルしきい値を設定します。
   http-slow-attack threshold alert-number *alert-number* | **content-length** *content-length* | **payload-length** *payload-length* | **packet-number** *packet-number* *

   デフォルトでは、HTTP同時接続、Content-Lengthフィールド値、ペイロードサイズ、および異常パケットのしきい値は、それぞれ5000、10000、50、および10です。
5. グローバルなHTTP低速攻撃検出期間を設定します。

   http-slow-attack period ***period***

   デフォルトでは、グローバルHTTP低速攻撃検出期間は60秒です。
6. (任意)HTTP低速攻撃から保護するグローバルポートを指定します。

   http-slow-attack port *port-list*

   デフォルトでは、HTTP低速攻撃防止によってポート80が保護されます。
7. HTTP低速攻撃に対するグローバルアクションを指定します。
   http-slow-attack action { **block-source timeout** *minutes* | **logging** } *

   デフォルトでは、HTTP低速攻撃に対するグローバルアクションは指定されていません。
8. IPアドレス固有のHTTP低速攻撃検出を設定します。
   http-slow-attack detect { **ip** *ipv4-address* | **ipv6** *ipv6-address* } **vpn-instance** *vpn-instance-name* **port** *port-list* **threshold** { **alert-number** *alert-number* | **content-length** *content-length* | **payload-length** *payload-length* | **packet-number** *packet-number* }* **period** *period* **action** { **block-source timeout** *minutes* | **logging** }*

   デフォルトでは、IPアドレス固有のHTTP低速攻撃検出は設定されていません。

攻撃検出免除の設定

このタスクについて

攻撃防御ポリシーは、免除されたパケットを識別するためにACLを使用します。ポリシーは、ACLによって許可されたパケットをチェックしません。信頼できるサーバーからのパケットを識別するようにACLを設定できます。免除機能は、誤警報率を減らし、パケット処理効率を向上させます。たとえば、攻撃防御ポリシーは、スキャン攻撃パケット(たとえば、OSPFまたはPIMパケット)と同じ送信元アドレスと異なる宛先アドレスを持つマルチキャストパケットを識別します。このようなパケットを攻撃検出から免除するようにACLを設定できます。

制限事項およびガイドライン

ACLが攻撃検出免除に使用されている場合、ACL許可規則の次の一致基準だけが有効になります。

- 送信元IPアドレス。
- 宛先IPアドレス。
- 送信元ポート。
- 宛先ポート。
- プロトコル。
- L3VPNインスタンス。
- 先頭以外のフラグメントを照合するためのfragmentキーワード。

手順

1. システムビューに入ります。
   system-view
2. 攻撃防御ポリシービューに入ります。
   attack-defense policy *policy-name*
3. 攻撃検出免除を設定します。
   exempt acl ipv6 { *acl-number* | **name** *acl-name* }
   デフォルトでは、攻撃検出免除は設定されていません。

インターフェイスへの攻撃防御ポリシーの適用

1. システムビューに入ります。
   system-view
2. システムビューに入ります。
   interface *interface-type interface-number*
3. インターフェイスに攻撃防御ポリシーを適用します。
   attack-defense apply policy *policy-name*
   デフォルトでは、攻撃防御ポリシーはインターフェイスに適用されません。

デバイスへの攻撃防御ポリシーの適用

このタスクについて

インターフェイスではなくデバイス自体に適用される攻撃防御ポリシーは、デバイス宛てのパケットを検出し、デバイスを標的とした攻撃を防止します。

攻撃防御ポリシーを適用すると、デバイス宛ての攻撃パケットの処理効率を向上させることができます。

デバイスとそのインターフェイスに攻撃防御ポリシーが適用されている場合、デバイス宛てのパケットは次のように処理されます。

1. 受信インターフェイスに適用されるポリシーによって、パケットが処理されます。
2. パケットが受信インターフェイスによってドロップされない場合、デバイスに適用されるポリシーによってパケットが処理されます。

手順

1. システムビューに入ります。
   system-view
2. 攻撃防御ポリシーをデバイスに適用します。
   attack-defense local apply policy *policy-name*
   デフォルトでは、攻撃防御ポリシーはデバイスに適用されません。

単一パケット攻撃イベントのログ非集約の有効化

このタスクについて

ログ集計は、ある期間に生成された複数のログを集計し、1つのログを送信します。集計されるログには、次の共通属性が必要です。

- 攻撃は、同じインターフェイスで検出されるか、デバイスに向けられます。
- 攻撃タイプ。
- 攻撃防御行動。
- 送信元および宛先IPアドレス。
- 被害者のIPアドレスが属するVPNインスタンス。

制限事項およびガイドライン

ログ集約を無効にしないことをお薦めします。大量のログは、コンソールの表示リソースを消費します。

手順

1. システムビューに入ります。
   system-view
2. 単一パケット攻撃イベントのログ非集約をイネーブルにします。
   attack-defense signature log non-aggregate
   デフォルトでは、単一パケット攻撃イベントのログ非集約はディセーブルになっています。

トップ攻撃統計情報ランキング機能のイネーブル化

このタスクについて

この機能は、攻撃者、被害者、および攻撃タイプに基づいてドロップされた攻撃パケットに関する統計情報を収集し、攻撃者および被害者ごとに上位の攻撃統計情報をランク付けします。上位の攻撃統計情報ランキングを表示するには、display attack-defense top-attack-statisticsコマンドを使用します。

手順

1. システムビューに入ります。
   system-view

2. トップ攻撃統計情報ランキング機能をイネーブルにします。

   attack-defense top-attack-statistics enable

   デフォルトでは、トップ攻撃統計情報ランキング機能はディセーブルになっています。

TCPクライアント検証の設定

このタスクについて

外部ネットワークに接続されているインターフェイスでTCPクライアント検証を設定します。TCPクライアント検証は、次のフラッド攻撃を含むTCPフラッド攻撃から内部TCPサーバーを保護します。

- SYN
- SYN-ACK
- RST
- FIN
- ACK

TCPクライアント検証によって保護されるIPアドレスは、手動で追加することも、自動的に学習することもできます。

- 保護されたIPアドレスを手動で追加できます。デバイスは、保護されたIPアドレス宛ての最初のSYNパケットを受信したときにクライアント検証を実行します。
- TCPクライアント検証は、フラッド攻撃検出と連携するときに、被害者のIPアドレスを保護されたIPリストに自動的に追加できます。フラッド攻撃防止アクションとしてclient-verifyが指定されていることを確認してください。詳細については、「フラッド攻撃防御ポリシーの設定」を参照してください。

safe resetモードでTCPクライアントが正当であると確認された場合、デバイスはクライアントのIPアドレスを信頼できるIPリストに追加します。デバイスは、信頼できるIPアドレスからのTCPパケットを直接転送します。

手順

1. システムビューに入ります。

   system-view

2. (任意)TCPクライアント検証機能で保護するIPアドレスを指定します。

   client-verify tcp protected { **ip** *destination-ip-address* | **ipv6** *destination-ipv6-address* } **vpn-instance** *vpn-instance-name* **port** *port-number*

3. インターフェイスビューを入力します。

   interface *interface-type interface-number*

4. TCPクライアント検証をイネーブルにします。
   - **safe reset**モードを設定します。

     client-verify tcp enable mode safe-reset
   - **SYN cookie**モードを設定します。

     client-verify tcp enable mode syn-cookie

   デフォルトでは、TCPクライアント検証はディセーブルになっています。

DNSクライアント検証の構成

このタスクについて

外部ネットワークに接続されているインターフェイスでDNSクライアント検証を設定します。DNSクライアント検証は、DNSフラッド攻撃から内部DNSサーバーを保護します。

DNSクライアント検証によって保護されたIPアドレスは、手動で追加することも、自動的に学習することもできます。

- 保護されたIPアドレスを手動で追加できます。デバイスは、保護されたIPアドレス宛ての最初のDNSクエリーを受信したときに、クライアント検証を実行します。
- DNSクライアント検証は、DNSフラッド攻撃検出と連携するときに、被害者のIPアドレスを保護されたIPリストに自動的に追加できます。DNSフラッド攻撃防止アクションとしてclient-verifyが指定されていることを確認してください。詳細については、「DNSフラッド攻撃防御ポリシーの設定」を参照してください。

DNSクライアントが正当であると確認された場合、デバイスはクライアントのIPアドレスを信頼できるIPリストに追加します。デバイスは、信頼できるIPアドレスからのDNSパケットを直接転送します。

手順

1. システムビューに入ります。

   system-view
2. (任意)DNSクライアント検証機能で保護するIPアドレスを指定します。
   client-verify dns protected { **ip** *destination-ip-address* | **ipv6** *destination-ipv6-address* } **vpn-instance** *vpn-instance-name* **port** *port-number*
3. インターフェイスビューを入力します。

   interface *interface-type interface-number*
4. DNSクライアント検証をイネーブルにします。

   client-verify dns enable

   デフォルトでは、DNSクライアント検証はディセーブルになっています。

DNS応答の検証の設定

このタスクについて

外部ネットワークに接続されているインターフェイスでDNS応答検証を設定します。DNS応答検証は、DNS応答フラッド攻撃から内部DNSクライアントを保護します。

DNS応答検証によって保護されたIPアドレスは、手動で追加することも、自動的に学習することもできます。

- 保護されたIPアドレスを手動で追加できます。デバイスは、保護されたIPアドレス宛ての最初のDNS応答を受信したときに、応答検証を実行します。
- DNS応答検証では、DNS応答フラッド攻撃検出と連携するときに、被害者のIPアドレスを保護IPリストに自動的に追加できます。DNS応答フラッド攻撃防止アクションとしてclient-verifyが指定されていることを確認してください。詳細については、「DNS応答フラッド攻撃防御ポリシーの設定」を参照してください。

DNSサーバーが正当であると確認された場合、デバイスはクライアントのIPアドレスを信頼できるIPリストに追加します。デバイスは、信頼できるIPアドレスからのDNS応答を直接転送します。

制限事項およびガイドライン

DNS応答検証機能を使用するには、サーバーで標準のTCP/IPプロトコル群とDNSプロトコルを使用する必要があります。標準以外のプロトコルを使用する正当なサーバーは、DNS応答認証サーバーによって不正であると検証されます。

手順

1. システムビューに入ります。

   system-view
2. (任意)DNS応答検証機能で保護するIPアドレスを指定します。
   Client-verify dns-reply protected { **ip** *destination-ip-address* | **ipv6** *destination-ipv6-address* } **vpn-instance** *vpn-instance-name* **port** *port-number*
3. インターフェイスビューを入力します。

   interface *interface-type interface-number*
4. DNS応答検証をイネーブルにします。

   client-verify dns-reply enable

   デフォルトでは、DNS応答検証はディセーブルになっています。

HTTPクライアント検証の設定

このタスクについて

外部ネットワークに接続されているインターフェイスでHTTPクライアント検証を設定します。HTTPクライアント検証は、HTTPフラッド攻撃から内部HTTPサーバーを保護します。

HTTPクライアント検証によって保護されるIPアドレスは、手動で追加することも、自動的に学習することもできます。

- 保護されたIPアドレスを手動で追加できます。デバイスは、保護されたIPアドレス宛ての最初のHTTP GETまたはPOSTパケットを受信したときに、クライアント検証を実行します。
- HTTPクライアント検証では、HTTPフラッド攻撃検出と連携するときに、被害者のIPアドレスを保護IPリストに自動的に追加できます。HTTPフラッド攻撃防止アクションとしてclient-verifyが指定されていることを確認してください。詳細については、「HTTPフラッド攻撃防御ポリシーの設定」を参照してください。

HTTPクライアントが正当であると確認された場合、デバイスはクライアントのIPアドレスを信頼できるIPリストに追加します。デバイスは、信頼できるIPアドレスからのHTTPパケットを直接転送します。

手順

1. システムビューに入ります。

   system-view
2. (任意)HTTPクライアント検証機能で保護するIPアドレスを指定します。
   client-verify http protected { **ip** *destination-ip-address* | **ipv6** *destination-ipv6-address* } **vpn-instance** *vpn-instance-name* **port** *port-number*
3. インターフェイスビューを入力します。

   interface *interface-type interface-number*
4. HTTPクライアント検証をイネーブルにします。

   client-verify http enable

   デフォルトでは、HTTPクライアント検証はディセーブルになっています。

SIPクライアント検証の設定

このタスクについて

外部ネットワークに接続されているインターフェイスでSIPクライアント検証を設定します。SIPクライアント検証は、SIPフラッド攻撃から内部SIPサーバーを保護します。

SIPクライアント検証によって保護されたIPアドレスは、手動で追加することも、自動的に学習することもで

きます。

- 保護されたIPアドレスを手動で追加できます。デバイスは、保護されたIPアドレス宛ての最初のINVITEパケットを受信すると、クライアント検証を実行します。
- SIPクライアント検証では、SIPフラッド攻撃検出と連携するときに、被害者のIPアドレスを保護対象IPリストに自動的に追加できます。SIPフラッド攻撃防止アクションとしてclient-verifyが指定されていることを確認してください。詳細については、「SIPフラッド攻撃防御ポリシーの設定」を参照してください。

SIPクライアントが正当であることが確認された場合、デバイスはクライアントのIPアドレスを信頼できるIPリストに追加します。デバイスは、信頼できるIPアドレスからのSIPパケットを直接転送します。

制限事項およびガイドライン

SIPクライアントによって送信されたパケットに、フラグメンテーションが原因で完全なヘッダー情報が含まれていない場合、正規のSIPクライアントはクライアント検証に合格しないことがあります。

手順

1. システムビューに入ります。

   system-view
2. (任意)SIPクライアント検証機能で保護するIPアドレスを指定します。
   client-verify sip protected { **ip** *destination-ip-address* | **ipv6** *destination-ipv6-address* } **vpn-instance** *vpn-instance-name* **port** *port-number*
3. インターフェイスビューを入力します。

   interface *interface-type interface-number*
4. SIPクライアント検証をイネーブルにします。

   client-verify sip enable

   デフォルトでは、SIPクライアント検証はディセーブルになっています。

IPブラックリスト機能の設定

このタスクについて

IPブラックリスト機能は、ブラックリストエントリ内のIPアドレスを送信元または宛先とするパケットをフィルタリングします。グローバルブラックリスト機能がイネーブルの場合、ブラックリスト機能はすべてのインターフェイスでイネーブルになります。

送信元または宛先IPブラックリストエントリを手動で追加できます。このようなエントリを作成する場合は、エントリにエージングタイムを設定できます。エージングタイムのないエントリは、手動で削除しないかぎり、エージングアウトされません。

デバイスは、スキャン攻撃検出と連携するときに、送信元IPブラックリストエントリを自動的に追加できます。動的に学習された各送信元IPブラックリストエントリには、ユーザーが設定できるエージングタイムがあります。スキャン攻撃防止アクションとして、block-sourceキーワードが指定されていることを確認してください。スキャン攻撃の検出と防止の詳細については、「スキャン攻撃防御ポリシーの設定」を参照してください。

手順

1. システムビューに入ります。

   system-view
2. (任意)IPブラックリストエントリを追加します。
   - 送信元IPv4ブラックリストエントリを追加します。
     blacklist ip *source-ip-address* **vpn-instance** *vpn-instance-name* **ds-lite-peer** *ds-lite-peer-address* **timeout** *minutes*

- 送信元IPv6ブラックリストエントリを追加します。
  blacklist destination-ip *destination-ip-address* **vpn-instance** *vpn-instance-name* **timeout** *minutes*
- 宛先IPv4ブラックリストエントリを追加します。
  blacklist destination-ipv6 *destination -ipv6-address* **vpn-instance** *vpn-instance-name* **timeout** *minutes*
- 宛先IPv6ブラックリストエントリを追加します。

  blacklist destination-ipv6 destination-ipv6-address vpn-instance vpn-instance-name] [**timeout** *minutes*]

3. (任意)ブラックリスト機能のロギングをイネーブルにします。

   blacklist logging enable

   デフォルトでは、ブラックリスト機能のロギングはディセーブルになっています。
4. ブラックリスト機能を有効にします。必要に応じて、次のいずれかのオプションを選択します。
   - グローバルブラックリスト機能をイネーブルにします。

     blacklist global enable

     デフォルトでは、グローバルブラックリスト機能はディセーブルになっています。
   - 次のコマンドを順番に実行して、インターフェイス上でブラックリスト機能をイネーブルにします。
     interface *interface-type interface-number*
     blacklist enable

     デフォルトでは、ブラックリスト機能はインターフェイス上でディセーブルになっています。

アドレスオブジェクトグループのブラックリストの設定

このタスクについて

この機能は、ブラックリストアドレスオブジェクトグループで指定されたサブネットから送信されたパケットをフィルタリングします。

ハードウェアと機能の互換性

ハードウェア	**機能の互換性**
MSR610	はい
MSR810、MSR810-W、MSR810-W-DB、MSR810-LM、MSR810-W-LM、MSR810-10-PoE、MSR810-LM-HK、MSR810-W-LM-HK MSR810-LM-CNDE-SJK、MSR810-CNDE-SJK、MSR810-EI、MSR810-LM-EA、MSR810-LM-EI	はい
MSR810-LMS、MSR810-LUS	いいえ
MSR810-SI、MSR810-LM-SI	いいえ
MSR810-LMS-EA、MSR810-LME	はい
MSR1004S-5G	はい
MSR2600-6-X1、MSR2600-15-X1、MSR2600-15-X1-T	はい
MSR2600-10-X1	はい
MSR 2630	はい
MSR3600-28およびMSR3600-51	はい
MSR3600-28-SI、MSR3600-51-SI	はい

MSR3600-28-X1、MSR3600-28-X1-DP、MSR3600-51-X1、MSR3600-51-X1-DP	はい
MSR3600-28-G-DP, MSR3600-51-G-DP	はい
MSR3610-I-DP、MSR3610-IE-DP、MSR3610-IE-ES MSR3610-IE-EAD、MSR-EAD-AK770、MSR3610-I-IG、MSR3610-IE-IG	はい
MSR3610-X1、MSR3610-X1-DP、MSR3610-X1-DC、MSR3610-X1-DP-DC、MSR3620-X1、MSR3640-X1	はい
MSR 3610、MSR 3620、MSR 3620-DP、MSR 3640、MSR 3660	はい
MSR3610-G、MSR3620-G	はい
MSR 3640-X 1-HI	はい

ハードウェア	機能の互換性
MSR810-W-WiNet、MSR810-LM-WiNet	はい
MSR830-4LM-WiNet	はい
MSR830-5BEI-WiNet、MSR830-6EI-WiNet、MSR830-10BEI-WiNet	はい
MSR830-6BHI-WiNet、MSR830-10BHI-WiNet	はい
MSR 2600-6-WiNet	はい
MSR2600-10-X1-WiNet	はい
MSR2630-WiNet	はい
MSR 3600-28-WiNet	はい

ハードウェア	機能の互換性
MSR3610-X1-WiNet	はい
MSR3610-WiNet、MSR3620-10-WiNet、MSR3620-DP-WiNet、MSR3620-WiNet、MSR3660-WiNet	はい

ハードウェア	機能の互換性
MSR860-6EI-XS	はい
MSR860-6HI-XS	はい
MSR2630-XS	はい
MSR3600-28-XS	はい
MSR3610-XS	はい
MSR3620-XS	はい
MSR3610-I-XS	はい
MSR3610-IE-XS	はい
MSR3620-X1-XS	はい
MSR3640-XS	はい
MSR3660-XS	はい

ハードウェア	機能の互換性

MSR810-LM-GL	はい
MSR810-W-LM-GL	はい
MSR830-6EI-GL	はい
MSR830-10EI-GL	はい
MSR830-6HI-GL	はい
MSR830-10HI-GL	はい
MSR1004S-5G-GL	はい
MSR2600-6-X1-GL	はい
MSR3600-28-SI-GL	はい

制限事項およびガイドライン

アドレスオブジェクトグループは、手動でのみブラックリストに追加したり、ブラックリストから削除したりできます。

アドレスオブジェクトグループのブラックリスト機能は、アドレスオブジェクトグループ機能と一緒に使用する必要があります。アドレスオブジェクトグループの詳細については、「オブジェクトグループの設定」を参照してください。

手順

1. システムビューに入ります。

   system-view
2. アドレスオブジェクトグループをブラックリストに追加します。

   blacklist object-group *object-group-name*

   デフォルトでは、アドレスオブジェクトグループはブラックリストにありません。
3. ブラックリスト機能を有効にします。必要に応じて、次のいずれかのオプションを選択します。
   - グローバルブラックリスト機能をイネーブルにします。

     blacklist global enable

     デフォルトでは、グローバルブラックリスト機能はディセーブルになっています。
   - interface viewを入力し、インターフェイスでブラックリスト機能をイネーブルにします。

     interface *interface-type interface-number*
     blacklist enable
     デフォルトでは、ブラックリスト機能はインターフェイス上でディセーブルになっています。

アドレスオブジェクトグループホワイトリストの設定

このタスクについて

この機能は、ホワイトリストに記載されたアドレスオブジェクトグループで指定されたサブネットから送信されたパケットを攻撃検出から免除します。

ハードウェアと機能の互換性

ハードウェア	**機能の互換性**
MSR610	はい

MSR810、MSR810-W、MSR810-W-DB、MSR810-LM、MSR810-W-LM、MSR810-10-PoE、MSR810-LM-HK、MSR810-W-LM-HK MSR810-LM-CNDE-SJK、MSR810-CNDE-SJK、MSR810-EI、MSR810-LM-EA、MSR810-LM-EI	はい
MSR810-LMS、MSR810-LUS	いいえ
MSR810-SI、MSR810-LM-SI	いいえ
MSR810-LMS-EA、MSR810-LME	はい
MSR1004S-5G	はい
MSR2600-6-X1、MSR2600-15-X1、MSR2600-15-X1-T	はい
MSR2600-10-X1	はい
MSR 2630	はい
MSR3600-28, MSR3600-51	はい
MSR3600-28-SI、MSR3600-51-SI	はい
MSR3600-28-X1、MSR3600-28-X1-DP、MSR3600-51-X1、MSR3600-51-X1-DP	はい
MSR3600-28-G-DP, MSR3600-51-G-DP	はい
MSR3610-I-DP、MSR3610-IE-DP、MSR3610-IE-ES MSR3610-IE-EAD、MSR-EAD-AK770、MSR3610-I-IG、MSR3610-IE-IG	はい
MSR3610-X1、MSR3610-X1-DP、MSR3610-X1-DC、MSR3610-X1-DP-DC、MSR3620-X1、MSR3640-X1	はい
MSR 3610、MSR 3620、MSR 3620-DP、MSR 3640、MSR 3660	はい
MSR3610-G、MSR3620-G	はい
MSR 3640-X 1-HI	はい

ハードウェア	機能の互換性
MSR810-W-WiNet、MSR810-LM-WiNe	はい
MSR830-4LM-WiNet	はい

ハードウェア	機能の互換性
MSR830-5BEI-WiNet、MSR830-6EI-WiNet、MSR830-10BEI-WiNet	はい
MSR830-6BHI-WiNet、MSR830-10BHI-WiNet	はい
MSR 2600-6- WiNet	はい
MSR2600-10-X1-WiNet	はい
MSR2630-WiNet	はい
MSR 3600-28- WiNet	はい
MSR3610-X1-WiNet	はい
MSR3610-WiNet、MSR3620-10-WiNet、MSR3620-DP-WiNet、MSR3620-WiNet、MSR3660-WiNet	はい

ハードウェア	機能の互換性
MSR860-6EI-XS	はい

MSR860-6HI-XS	はい
MSR2630-XS	はい
MSR3600-28-XS	はい
MSR3610-XS	はい
MSR3620-XS	はい
MSR3610-I-XS	はい
MSR3610-IE-XS	はい
MSR3620-X1-XS	はい
MSR3640-XS	はい
MSR3660-XS	はい

ハードウェア	**機能の互換性**
MSR810-LM-GL	はい
MSR810-W-LM-GL	はい
MSR830-6EI-GL	はい
MSR830-10EI-GL	はい
MSR830-6HI-GL	はい
MSR830-10HI-GL	はい
MSR1004S-5G-GL	はい
MSR2600-6-X1-GL	はい
MSR3600-28-SI-GL	はい

制限事項およびガイドライン

アドレスオブジェクトグループは、ホワイトリストに手動で追加するか、ホワイトリストから手動で削除する必要があります。

アドレスオブジェクトグループホワイトリスト機能は、アドレスオブジェクトグループ機能と一緒に使用する必要があります。アドレスオブジェクトグループの詳細については、「オブジェクトグループの設定」を参照してください。

手順

1. システムビューに入ります。

   system-view
2. アドレスオブジェクトグループをホワイトリストに追加します。

   whitelist object-group *object-group-name*

   デフォルトでは、アドレスオブジェクトグループはホワイトリストに追加されません。
3. ホワイトリスト機能を有効にします。必要に応じて、次のいずれかのオプションを選択します。
   - グローバルホワイトリスト機能をイネーブルにします。

     whitelist global enable

     デフォルトでは、グローバルホワイトリスト機能はディセーブルになっています。
   - interface viewと入力し、インターフェイスでホワイトリスト機能をイネーブルにします。

     interface *interface-type interface-number*
     whitelist enable

デフォルトでは、インターフェイスのホワイトリスト機能はディセーブルになっています。

ログイン攻撃防止の設定

このタスクについて

ログイン攻撃防止機能は、ユーザーが最大連続ログイン試行回数に失敗した場合に、ログインDoS攻撃を検出します。この機能は、ブラックリスト機能をトリガーして、ユーザーのIPをブラックリストに追加します。ユーザーからの次のログイン試行は、ブロック期間中ブロックされます。ログイン攻撃防止を有効にするには、グローバルブラックリスト機能を有効にする必要があります。

この機能により、ログインDoS攻撃を効果的に防ぐことができます。

ハードウェアと機能の互換性

ハードウェア	機能の互換性
MSR610	いいえ
MSR810、MSR810-W、MSR810-W-DB、MSR810-LM、MSR810-W-LM、MSR810-10-PoE、MSR810-LM-HK MSR810-W-LM-HK、MSR810-LM-CNDE-SJK、MSR810-CNDE-SJK、MSR810-EI、MSR810-LM-EA、MSR810-LM-EI	いいえ
MSR810-LMS、MSR810-LUS	はい
MSR810-SI、MSR810-LM-SI	はい
MSR810-LMS-EA、MSR810-LME	いい
MSR1004S-5G	はい
MSR2600-6-X1、MSR2600-15-X1、MSR2600-15-X1-T	はい
MSR2600-10-X1	はい
MSR 2630	はい
MSR3600-28, MSR3600-51	はい
MSR3600-28-SI、MSR3600-51-SI	はい
MSR3600-28-X1、MSR3600-28-X1-DP、MSR3600-51-X1、MSR3600-51-X1-DP	はい
MSR3600-28-G-DP, MSR3600-51-G-DP	はい

ハードウェア	機能の互換性
試験MSR3610-I-DP、MSR3610-IE-DP、MSR3610-IE-ES、MSR3610-IE-EAD、MSR-EAD-AK770、 MSR3610-I-IG、MSR3610-IE-IG	はい
MSR3610-X1、MSR3610-X1-DP、MSR3610-X1-DC、MSR3610-X1-DP-DC、MSR3620-X1、 MSR3640-X1	はい
MSR 3610、MSR 3620、MSR 3620-DP、MSR 3640、MSR 3660	はい
MSR3610-G、MSR3620-G	はい
MSR 3640-X 1-HI	はい

ハードウェア	機能の互換性

MSR810-W-WiNet、MSR810-LM-WiNet	はい
MSR830-4LM-WiNet	はい
MSR830-5BEI-WiNet、MSR830-6EI-WiNet、MSR830-10BEI-WiNet	はい
MSR830-6BHI-WiNet、MSR830-10BHI-WiNet	はい
MSR 2600-6-WiNet	はい
MSR2600-10-X1-WiNet	はい
MSR2630-WiNet	はい
MSR 3600-28-WiNet	はい
MSR3610-X1-WiNet	はい
MSR3610-WiNet、MSR3620-10-WiNet、MSR3620-DP-WiNet、MSR3620-WiNet、MSR3660-WiNet	はい

ハードウェア	機能の互換性
MSR860-6EI-XS	はい
MSR860-6HI-XS	はい
MSR2630-XS	はい
MSR3600-28-XS	はい
MSR3610-XS	はい
MSR3620-XS	はい
MSR3610-I-XS	はい
MSR3610-IE-XS	はい
MSR3620-X1-XS	はい
MSR3640-XS	はい
MSR3660-XS	はい

ハードウェア	機能の互換性
MSR810-LM-GL	いいえ
MSR810-W-LM-GL	いいえ
MSR830-6EI-GL	はい
MSR830-10EI-GL	はい

ハードウェア	機能の互換性
MSR830-6HI-GL	はい
MSR830-10HI-GL	はい
MSR1004S-5G-GL	はい
MSR2600-6-X1-GL	はい
MSR3600-28-SI-G	はい

手順

1. システムビューに入ります。
   system-view
2. ログイン攻撃防止をイネーブルにします。
   attack-defense login enable
   デフォルトでは、ログイン攻撃防止はディセーブルになっています。
3. 連続するログイン失敗の最大数を設定します。
   attack-defense login max-attempt *max-attempt*
   デフォルト値は3です。
4. ログイン試行をブロックするブロック期間を設定します。
   attack-defense login block-timeout *minutes*
   デフォルト値は60分です。
5. グローバルブラックリスト機能をイネーブルにします。
   blacklist global enable
   デフォルトでは、グローバルブラックリスト機能はディセーブルになっています。

ログイン遅延の有効化

このタスクについて

ログイン遅延機能は、ユーザーがログイン試行に失敗した後、デバイスがユーザーからのログイン要求を受け入れるのを遅延させます。この機能は、ログインディクショナリ攻撃を遅らせることができます。

ログイン遅延機能は、ログイン攻撃防止機能とは独立しています。

手順

1. システムビューに入ります。
   system-view
2. ログイン遅延機能をイネーブルにします。
   attack-defense login reauthentication-delay ***seconds***
   デフォルトでは、ログイン遅延機能はディセーブルになっています。デバイスは、ログイン試行に失敗したユーザーからのログイン要求の受け入れを遅延させません。

攻撃の検出と防止のための表示およびメンテナンスコマンド

任意のビューでdisplayコマンドを使用し、ユーザービューでresetコマンドを使用します。攻撃の検出と防御を表示および維持するには、次の手順を実行します。

タスク	コマンド

IPv4アドレスのフラッド攻撃検出および防止の統計情報を表示します。	スタンドアロンモードの場合: **display attack-defense** { **ack-flood** | **dns-flood** | **dns-reply-flood** | **fin-flood** | **flood** | **http-flood** | **icmp-flood** | **rst-flood** | **sip-flood** | **syn-ack-flood** | **syn-flood** | **udp-flood** } **statistics ip** [*ip-address* [**vpn** *vpn-instance-name*]] [**interface** *interface-type interface-number* | **local**] [**count**] IRFモードの場合: **display attack-defense** { **ack-flood** | **dns-flood** | **dns-reply-flood** | **fin-flood** | **flood** | **http-flood** | **icmp-flood** | **rst-flood** | **sip-flood** | **syn-ack-flood** | **syn-flood** | **udp-flood** } **statistics ip** [*ip-address* [**vpn** *vpn-instance-name*]] [[**interface** *interface-type interface-number* | **local**] [**slot** *slot-number*]] [**count**]
IPv6アドレスのフラッド攻撃検出および防止の統計情報を表示します。	スタンドアロンモードの場合: **display attack-defense** { **ack-flood** | **dns-flood** | **dns-reply-flood** | **fin-flood** | **flood** | **http-flood** | **icmpv6-flood** | **rst-flood** | **sip-flood** | **syn-ack-flood** | **syn-flood** | **udp-flood** } **statistics ipv6** [*ipv6-address* [**vpn** *vpn-instance-name*]] [**interface** *interface-type interface-number* | **local**] [**count**] IRFモードの場合: **display attack-defense** { **ack-flood** | **dns-flood** | **dns-reply-flood** | **fin-flood** | **flood** | **http-flood** | **icmpv6-flood** | **rst-flood** | **sip-flood** | **syn-flood** | **syn-ack-flood** | **udp-flood** } **statistics ipv6** [*ipv6-address* [**vpn** *vpn-instance-name*]] [[**interface** *interface-type interface-number* | **local**] [**slot** *slot-number*]] [**count**]
IPv4 HTTP低速攻撃の検出および防止に関する統計情報を表示します。	スタンドアロンモードの場合: **display attack-defense http-slow-attack statistics ip** [*ip-address* [**vpn-instance** *vpn-instance-name*]] [**interface** { *interface-type interface-number* | *interface-name* } | **local**] [**count**] IRFモードの場合: **display attack-defense http-slow-attack statistics ip** [*ip-address* [**vpn-instance** *vpn-instance-name*]] [[**interface** { *interface-type interface-number* | *interface-name* } | **local**] [**slot** *slot-number*]] [**count**]
IPv6 HTTPに関する統計情報を表示します。	スタンドアロンモードの場合:

タスク	コマンド

遅い攻撃の検出と防止。	**display attack-defense http-slow-attack statistics ipv6** [*ipv6-address* [**vpn-instance** *vpn-instance-name*]] [**interface** { *interface-type interface-number* | *interface-name* } | **local**] [**count**] IRFモードの場合: **display attack-defense http-slow-attack statistics ipv6** [*ipv6-address* [**vpn-instance** *vpn-instance-name*]] [[**interface** { *interface-type interface-number* | *interface-name* } | **local**] [**slot** *slot-number*]] [**count**]
攻撃防御ポリシーの設定を表示します。	**display attack-defense policy** [*policy-name*]
フラッド攻撃の検出および防止によって保護されているIPv4アドレスに関する情報を表示します。	スタンドアロンモードの場合: **display attack-defense policy** *policy-name* { **ack-flood** | **dns-flood** | **dns-reply-flood** | **fin-flood** | **flood** | **http-flood** | **icmp-flood** | **rst-flood** | **sip-flood** | **syn-ack-flood** | **syn-flood** | **udp-flood** } **ip** [*ip-address* [**vpn** *vpn-instance-name*]] [**count**] IRFモードの場合: **display attack-defense policy** *policy-name* { **ack-flood** | **dns-flood** | **dns-reply-flood** | **fin-flood** | **flood** | **http-flood** | **icmp-flood** | **rst-flood** | **sip-flood** | **syn-ack-flood** | **syn-flood** | **udp-flood** } **ip** [*ip-address* [**vpn** *vpn-instance-name*]] [**slot** *slot-number*] [**count**]
フラッド攻撃の検出および防止によって保護されているIPv6アドレスに関する情報を表示します。	スタンドアロンモードの場合: **display attack-defense policy** *policy-name* { **ack-flood** | **dns-flood** | **dns-reply-flood** | **fin-flood** | **flood** | **http-flood** | **icmpv6-flood** | **rst-flood** | **sip-flood** | **syn-ack-flood** | **syn-flood** | **udp-flood** } **ipv6** [*ipv6-address* [**vpn** *vpn-instance-name*]] [**count**] IRFモードの場合: **display attack-defense policy** *policy-name* { **ack-flood** | **dns-flood** | **dns-reply-flood** | **fin-flood** | **flood** | **http-flood** | **icmpv6-flood** | **rst-flood** | **sip-flood** | **syn-ack-flood** | **syn-flood** | **udp-flood** } **ipv6** [*ipv6-address* [**vpn** *vpn-instance-name*]] [**slot** *slot-number*]] [**count**]
IPv4スキャン攻撃者に関する情報を表示します。	スタンドアロンモードの場合: **display attack-defense scan attacker ip** [**interface** *interface-type interface-number* | **local**] [**count**] IRFモードの場合: **display attack-defense scan attacker ip [[interface interface-type interface-number | local] [slot slot-number]] [count]**

タスク	コマンド
IPv6スキャン攻撃者に関する情報を表示します。	スタンドアロンモードの場合: **display attack-defense scan attacker ipv6** [**interface** *interface-type interface-number* | **local**] [**count**] IRFモードの場合: **display attack-defense scan attacker ipv6** [[**interface** *interface-type interface-number* | **local**] [**slot** *slot-number*]] [**count**]
インターフェイス上の攻撃検出および防御統計情報を表示します。	スタンドアロンモードの場合: **display attack-defense statistics interface** interface-type interface-number IRFモードの場合: **display attack-defense statistics interface** *interface-type interface-number* [**slot** *slot-number*]
デバイスの攻撃検出および防御の統計情報を表示します。	スタンドアロンモードの場合: display attack-defense statistics local IRFモードの場合: **display attack-defense statistics local** [**slot** *slot-number*]
上位10件の攻撃統計情報を表示します。	**display attack-defense top-attack-statistics** { **last-1-hour** | **last-24-hours** | **last-30-days** } [**by-attacker** | **by-type** | **by-victim**]
宛先IPv4ブラックリストエントリを表示します。	スタンドアロンモードの場合: **display blacklist destination-ip** [*destination-ip-address* [**vpn-instance** *vpn-instance-name*]] [**count**] IRFモードの場合: **display blacklist destination-ip** [*destination-ip-address* [**vpn-instance** *vpn-instance-name*]] [**slot** *slot-number*] [**count**]
宛先IPv6ブラックリストエントリを表示します。	スタンドアロンモードの場合: **display blacklist destination-ipv6** [*destination-ipv6-address* [**vpn-instance** *vpn-instance-name*]] [**count**] **display blacklist destination-ipv6** [*destination-ipv6-address* [**vpn-instance** *vpn-instance-name*]] [**count**] IRFモードの場合: **display blacklist destination-ipv6** [*destination-ipv6-address* [**vpn-instance** *vpn-instance-name*]] [**slot** *slot-number*] [**count**]
送信元IPv4ブラックリストエントリを表示します。	スタンドアロンモードの場合: **display blacklist ip** [*source-ip-address* [**vpn-instance** *vpn-instance-name*] [**ds-lite-peer** *ds-lite-peer-address*] | **count**]

タスク	コマンド
	IRFモードの場合: **display blacklist ip** [*source-ip-address* [**vpn-instance** *vpn-instance-name*] [**ds-lite-peer** *ds-lite-peer-address*]] [**slot** *slot-number*] [**count**]
送信元IPv6ブラックリストエントリを表示します。	スタンドアロンモードの場合: **display blacklist ipv6** [*source-ipv6-address* [**vpn-instance** *vpn-instance-name*] | **count**] IRFモードの場合: **display blacklist ipv6** [*source-ipv6-address* [**vpn-instance** *vpn-instance-name*]] [**slot** *slot-number*] [**count**]
クライアントの確認用に保護されたIPv4アドレスを表示します。	スタンドアロンモードの場合: **display client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **protected ip** [*ip-address* [**vpn** *vpn-instance-name*]] [**port** *port-number*] [**count**] IRFモードの場合: **display client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **protected ip** [*ip-address* [**vpn** *vpn-instance-name*]] [**port** *port-number*] [**slot** *slot-number*] [**count**]
クライアントの確認用に保護されたIPv6アドレスを表示します。	スタンドアロンモードの場合: **display client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **protected ipv6** [*ipv6-address* [**vpn** *vpn-instance-name*]] [**port** *port-number*] [**count**] IRFモードの場合: **display client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **protected ipv6** [*ipv6-address* [**vpn** *vpn-instance-name*]] [**port** *port-number*] [**slot** *slot-number*] [**count**]
クライアント検証用の信頼できるIPv4アドレスを表示します。	スタンドアロンモードの場合: **display client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **trusted ip** [*ip-address* [**vpn** *vpn-instance-name*]] [**count**] IRFモードの場合: **display client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **trusted ip** [*ip-address* [**vpn** *vpn-instance-name*]] [**slot** *slot-number*] [**count**]
クライアント検証用の信頼されたIPv6アドレスを表示します。	スタンドアロンモードの場合: **display client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **trusted ipv6** [*ipv6-address* [**vpn** *vpn-instance-name*]] [**count**] IRFモードの場合: **display client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **trusted ipv6** [*ipv6-address* [**vpn** *vpn-instance-name*]] [**slot** *slot-number*]] [**count**]

タスク	コマンド
ホワイトリストのアドレスオブジェクトグループに一致するパケットに関する統計情報を表示します。	スタンドアロンモードの場合: **display whitelist object-group** [*object-group-name*] IRFモードの場合: **display whitelist object-group** [*object-group-name*] [**slot** *slot-number*]
フラッド攻撃の検出と防止の統計情報をクリアします。	**reset attack-defense policy** *policy-name* **flood protected** { **ip** | **ipv6** } **statistics**
インターフェイスの攻撃検出および防御統計情報をクリアします。	**reset attack-defense statistics interface** interface-type interface-number
デバイスの攻撃検出および防御統計情報をクリアします。	reset attack-defense statistics local
上位10件の攻撃統計情報をクリアします。	reset attack-defense top-attack-statistics
ダイナミック宛先IPv4ブラックリストエントリを削除します。	**reset blacklist destination-ip** { *destination-ip-address* [**vpn-instance** *vpn-instance-name*] | **all** }
ダイナミック宛先IPv6ブラックリストエントリを削除します。	**reset blacklist** **destination-ipv6**{ *destination-ipv6-address* [**vpn-instance** *vpn-instance-name*] | **all** }
ダイナミック送信元IPv4ブラックリストエントリを削除します。	**reset blacklist ip** { *source-ip-address* [**vpn-instance** *vpn-instance-name*] [**ds-lite-peer** *ds-lite-peer-address*] | **all** }
ダイナミック送信元IPv6ブラックリストエントリを削除します。	**reset blacklist ipv6** { *source-ipv6-address* [**vpn-instance** *vpn-instance-name*] | **all** }
ブラックリスト統計情報をクリアします。	reset blacklist statistics
クライアント検証用の保護されたIP統計情報をクリアします。	**reset client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **protected** { **ip** | **ipv6** } **statistics**
クライアント検証のために、信頼できるIPリストをクリアします。	**reset client-verify** { **dns** | **dns-reply** | **http** | **sip** | **tcp** } **trusted** { **ip** | **ipv6** }
ホワイトリストのアドレスオブジェクトグループに一致するパケットに関する統計情報をクリアします。	reset whitelist statistics

攻撃の検出と防御の設定例

例:インターフェイスベースの攻撃検出および防御の設定

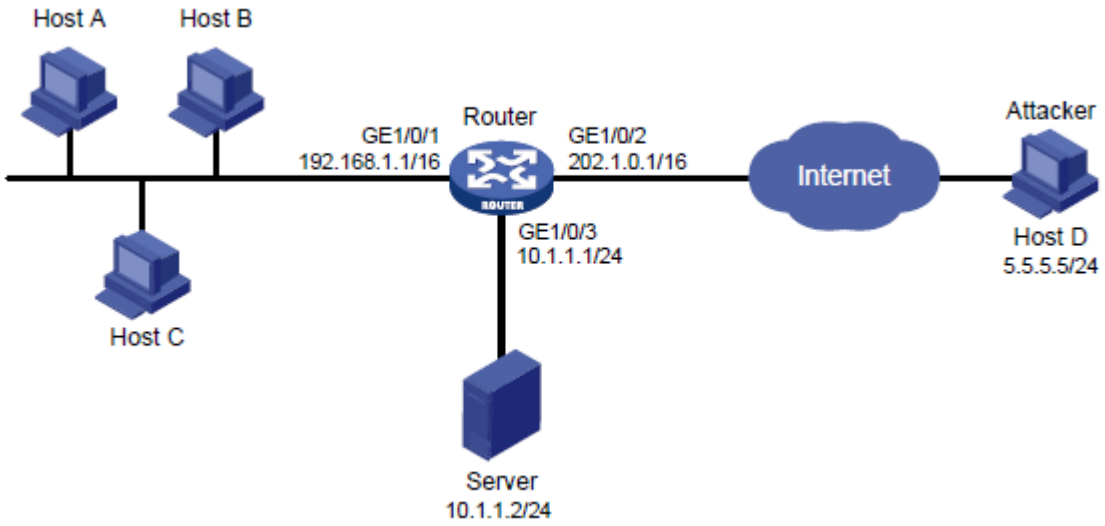
ネットワーク構成

図10に示すように、ルーターは内部ネットワークのゲートウェイです。

次の要件を満たすように、攻撃防御ポリシーを設定し、ポリシーをGigabitEthernet 1/0/2に適用します。

- 内部ホストおよびサーバーに低レベルのスキャン攻撃検出を提供します。スキャン攻撃が検出された場合は、攻撃をログに記録し、攻撃者を10分間ブラックリストに載せます。
- smurf攻撃から内部ホストとサーバーを保護します。smurf攻撃が検出された場合は、攻撃をログに記録します。
- SYNフラッド攻撃から内部サーバーを保護します。1秒間にサーバーに送信されるSYNパケットの数が5000以上の場合は、攻撃をログに記録し、後続のパケットをドロップします。
- 内部サーバーをHTTP低速攻撃から保護します。ポート80および8080へのHTTP同時接続数が3000を超えると、ルーターはHTTP低速攻撃検出状態になります。ルーターが60秒以内に10を超えるHTTP低速攻撃パケットを受信すると、デバイスはログを生成し、攻撃元をIPブラックリストに追加します。

図10 ネットワークダイアグラム



手順

#ルーターのインターフェイスにIPアドレスを設定します(詳細は省略)。
#グローバルブラックリスト機能をイネーブルにします。
<Router> system-view
[Router] blacklist global enable

#攻撃防御ポリシーa1を作成します。

[Router] attack-defense policy a1

#smurf攻撃のシグニチャ検出を設定し、防止アクションとしてロギングを指定します。

[Router-attack-defense-policy-a1] signature detect smurf action logging

#低レベルのスキャン攻撃検出を設定し、防止アクションとしてloggingとblock-sourceを指定し、ブラックリストエントリのエージングタイムを10分に設定します。
[Router-attack-defense-policy-a1] scan detect level low action logging block-source timeout 10

#SYNフラッド攻撃検出を10.1.1.2に設定し、攻撃防止トリガーしきい値を5000に設定し、防止アクションとしてロギングとドロップを指定します。
[Router-attack-defense-policy-a1] syn-flood detect ip 10.1.1.2 threshold 5000 action logging drop
[Router-attack-defense-policy-a1] quit

#10.1.1.2に対してHTTP低速攻撃検出を設定します。保護ポートを80および8080に指定し、HTTP同時接続しきい値を3000に、Content-Lengthフィールド値しきい値を10000に、ペイロードサイズしきい値を20に、異常パケットしきい値を10に設定します。防止アクションとしてloggingおよびblack-sourceを指定します。
[Router-attack-defense-policy-a1] http-slow-attack detect ip 10.1.1.2 port 80 8080 threshold alert-number 3000 content-length 10000 payload-length 20 packet-number 10 action logging block-source

#攻撃防御ポリシーa1をGigabitEthernet 1/0/2に適用します。
[Router] interface gigabitethernet 1/0/2
[Router-GigabitEthernet1/0/2] attack-defense apply policy a1
[Router-GigabitEthernet1/0/2] quit

設定の確認

#攻撃防御ポリシーa1が正常に設定されていることを確認します。

```
[Router] display attack-defense policy a1
Attack-defense Policy Information
-------------------------------------------------------------------------
Policy name                          : a1
Applied list                         : GE1/0/2
-------------------------------------------------------------------------
Exempt IPv4 ACL :                    Not configured
Exempt IPv6 ACL :                    Not configured
-------------------------------------------------------------------------
Actions: CV-Client verify BS-Block source L-Logging D-Drop N-None

Signature attack defense configuration:
Signature name                       Defense    Level          Actions
Fragment                             Disabled   low            L
Impossible                           Disabled   medium         L,D
Teardrop                             Disabled   medium         L,D
Tiny fragment                        Disabled   low            L
IP option abnormal                   Disabled   medium         L,D
Smurf                                Enabled    medium         L
Traceroute                           Disabled   low            L
Ping of death                        Disabled   medium         L,D
Large ICMP                           Disabled   info           L
 Max length 4000 bytes
Large ICMPv6                         Disabled   info           L
    Max length 4000 bytes
TCP invalid flags                    Disabled   medium         L,D
TCP null flag                        Disabled   medium         L,D
TCP all flags                        Disabled   medium         L,D
TCP SYN-FIN flags                    Disabled   medium         L,D
TCP FIN only flag                    Disabled   medium         L,D
TCP Land                             Disabled   medium         L,D
Winnuke                              Disabled   medium         L,D
UDP Bomb                             Disabled   medium         L,D
UDP Snork                            Disabled   medium         L,D
UDP Fraggle                          Disabled   medium         L,D
IP option record route               Disabled   info           L
IP option internet timestamp         Disabled   info           L
IP option security                   Disabled   info           L
```

IP option loose source routing	Disabled	info	L
IP option stream ID	Disabled	info	L
IP option strict source routing	Disabled	info	L
IP option route alert	Disabled	info	L
ICMP echo request	Disabled	info	L
ICMP echo reply	Disabled	info	L
ICMP source quench	Disabled	info	L
ICMP destination unreachable	Disabled	info	L
ICMP redirect	Disabled	info	L
ICMP time exceeded	Disabled	info	L
ICMP parameter problem	Disabled	info	L
ICMP timestamp request	Disabled	info	L
ICMP timestamp reply	Disabled	info	L
ICMP information request	Disabled	info	L
ICMP information reply	Disabled	info	L
ICMP address mask request	Disabled	info	L
ICMP address mask reply	Disabled	info	L
ICMPv6 echo request	Disabled	info	L
ICMPv6 echo reply	Disabled	info	L
ICMPv6 group membership query	Disabled	info	L
ICMPv6 group membership report	Disabled	info	L
ICMPv6 group membership reduction	Disabled	info	L
ICMPv6 destination unreachable	Disabled	info	L
ICMPv6 time exceeded	Disabled	info	L
ICMPv6 parameter problem	Disabled	info	L
ICMPv6 packet too big	Disabled	info	L
IPv6 extension header abnormal	Disabled	Info	L
IPv6 extension header exceeded	Disabled	Info	L
Limit	7		

Scan attack defense configuration:
Defense : Enabled
Level : low
Actions : L,BS(10)

Flood attack defense configuration:

Flood type	Global dest/src thres(pps)	Global actions	Service ports	Non-specific
DNS flood	1000/1000	-	53	Disabled
DNS reply flood	1000/1000	-	-	Disabled
HTTP flood	1000/1000	-	80	Disabled
SIP flood	1000/1000	-	5060	Disabled
SYN flood	1000/1000	80	-	Enabled
ACK flood	1000/1000	-	-	Disabled
SYN-ACK flood	1000/1000	-	-	Disabled
RST flood	1000/1000	-	-	Disabled
FIN flood	1000/1000	-	-	Disabled
UDP flood	1000/1000	-	-	Disabled
ICMP flood	1000/1000	-	-	Disabled
ICMPv6 flood	1000/1000	-	-	Disabled

Flood attack defense for protected IP addresses:

Address	VPN instance	Flood type	Thres(pps)	Actions	Ports
10.1.1.2	--	SYN-FLOOD	5000	L,D	-

HTTP slow attack defense configuration:
Non-specific: Disabled
Global threshold:
Alert-number: 5000
Content-length: 10000
Payload-length: 50
Packet-number: 10
Global period: 60 seconds
Global actions: -

Ports: 80
Threshold: AN-Alert number,CL-Content length,PL-Payload length,PN-Packet number
HTTP slow attack defense configuration for protected IP addresses:

Address	VPN instance	Threshold(AN/CL/PL/PN)	Period	Actions	Ports
10.1.1.2	--	3000,10000,20,10	60	L,BS(10)	80,8080

#攻撃の検出と防御がGigabitEthernet 1/0/2で有効になることを確認します。

[Router] display attack-defense statistics interface gigabitethernet 1/0/2
Attack policy name: a1
Scan attack defense statistics:

AttackType	AttackTimes	Dropped
Port scan	2	0

IP sweep 3 0
Distribute port scan 1 0
Flood attack defense statistics:
AttackType AttackTimes Dropped
SYN flood 1 5000
Signature attack defense statistics:
AttackType AttackTimes Dropped

Smurf	1	0

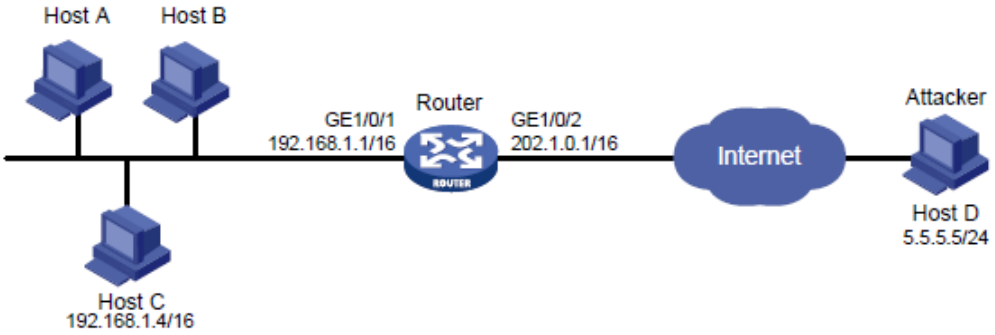
HTTP slow attack defense statistics:
AttackType AttackTimes
HTTP slow attack 2
Verify dynamic IPv4 blacklist entries.
[Router] display blacklist ip
IP address VPN instance DS-Lite tunnel peer Type TTL(sec) Dropped
5.5.5.5 -- -- Dynamic 600 353452

例:送信元IPブラックリストの設定

ネットワーク構成

図11に示すように、攻撃者のホストDからのパケットを永続的にブロックし、ホストCからのパケットを50分間ブロックするように、ルーター上で送信元IPブラックリストエントリを設定します。

図11 ネットワークダイアグラム



手順

#ルーターのインターフェイスにIPアドレスを設定します(詳細は省略)。
#グローバルブラックリスト機能をイネーブルにします。
<Router> system-view
[Router] blacklist global enable
#ホストDの送信元IPv4ブラックリストエントリを追加します。

[Router] blacklist ip 5.5.5.5

#ホストCの送信元IPv4ブラックリストエントリを追加し、ブラックリストエントリのエージングタイムを50分に設定します。

[Router] blacklist ip 192.168.1.4 timeout 50

設定の確認

#送信元IPv4ブラックリストエントリが正常に追加されたことを確認します。
<Router> display blacklist ip
IP address VPN instance DS-Lite tunnel peer Type TTL(sec) Dropped
5.5.5.5 -- -- Manual Never 0

192.168.1.4 -- -- Manual 2989 0

#ルーターがホストDからのパケットをドロップすることを確認します(詳細は省略)。

#undo blacklist ip 5.5.5.5コマンドを実行し、ルーターがホストDからパケットを転送することを確認します(詳細は省略)。

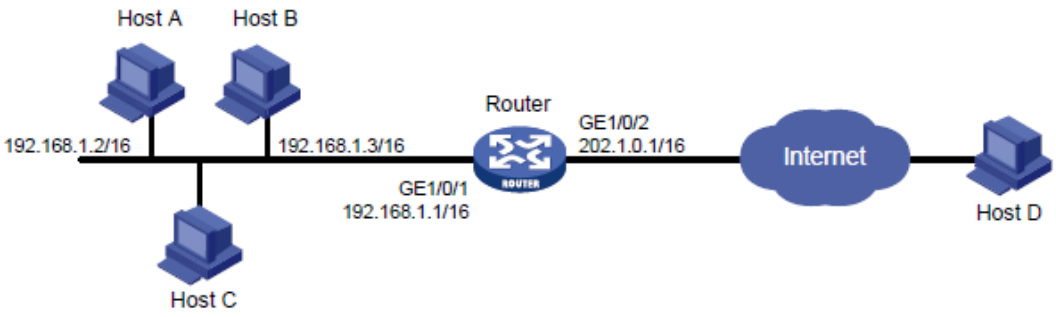
#ルーターがホストCからのパケットを50分間ドロップし、50分後にホストCからのパケットを転送することを確認します(詳細は省略)。

例:宛先IPブラックリストの設定

ネットワーク構成

図12に示すように、ホストA宛てのパケットを永続的にブロックし、ホストB宛てのパケットを50分間ブロックするように、ルーター上で宛先IPブラックリストエントリを設定します。

図12 ネットワークダイアグラム



手順

#インターフェイスにIPアドレスを割り当てます(詳細は省略)。
#グローバルブラックリスト機能をイネーブルにします。
<Router> system-view
[Router] blacklist global enable
#ホストAの宛先IPv4ブラックリストエントリを追加します。

[Router] blacklist destination-ip 192.168.1.2

#ホストBの宛先IPv4ブラックリストエントリを追加し、ブラックリストエントリのエージングタイムを50分に設定します。

[Router] blacklist destination-ip 192.168.1.3 timeout 50

設定の確認

#宛先IPv4ブラックリストエントリが正常に追加されたことを確認します。
[Router] display blacklist destination-ip

IP address	VPN instance	Type	TTL(sec)	Dropped
192.168.1.2	--	Manual	Never	0
192.168.1.3	--	Manual	2989	0

#ルーターがホストA宛てのパケットをドロップすることを確認します(詳細は省略)。

#undo blacklist destination-ip 192.168.1.2コマンドを実行し、ルーターがパケットをホストAに転送することを確認します(詳細は省略)。

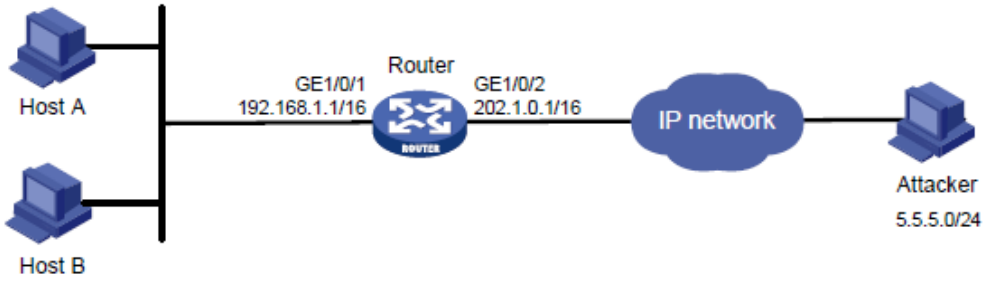
#ルーターがホストB宛てのパケットを50分間ドロップし、50分後にパケットをホストBに転送することを確認します(詳細は省略)。

例:アドレスオブジェクトグループのブラックリストの設定

ネットワーク構成

図13に示すように、サブネットからの攻撃を防ぐために、サブネット5.5.5.0/24からのすべてのパケットをブロックするように、ルーター上でアドレスオブジェクトグループブラックリスト機能を設定します。

図13 ネットワークダイアグラム



手順

#ルーターのインターフェイスにIPアドレスを設定します(詳細は省略)。
#グローバルブラックリスト機能をイネーブルにします。
<Router> system-view
[Router] blacklist global enable
#IPv4アドレスオブジェクトグループobj1を作成します。サブネット5.5.5.0/24でIPv4アドレスオブジェクトを設定します。
[Router] object-group ip address obj1
[Router-obj-grp-ip-obj1] network subnet 5.5.5.0 24
[Router] quit
#IPv4アドレスオブジェクトグループobj1をブラックリストに追加します。
[Router] blacklist object-group obj1

設定の確認

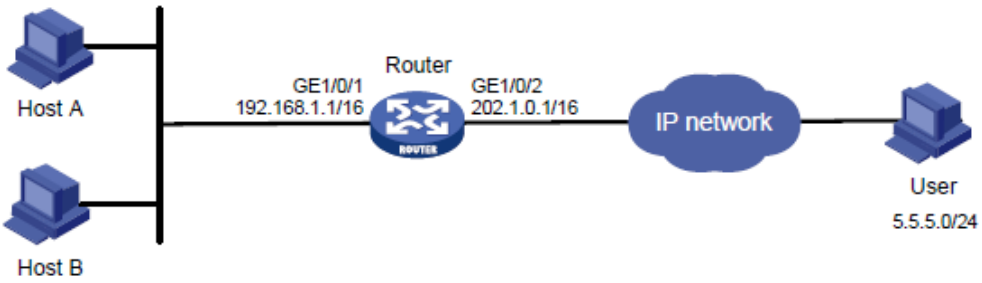
#ルーターでundo blacklist object-groupコマンドを実行しない限り、ルーターがサブネット5.5.5.0/24からのすべてのパケットをドロップすることを確認します(詳細は省略)。

例:アドレスオブジェクトグループのホワイトリストの設定

ネットワーク構成

図14に示すように、サブネット5.5.5.0/24からのすべてのパケットが通過できるように、ルーター上でアドレスオブジェクトグループホワイトリスト機能を設定します。

図14 ネットワークダイアグラム



手順

#ルーターのインターフェイスにIPアドレスを設定します。(詳細は省略)
#グローバルホワイトリスト機能を有効にします。
<Router> system-view
[Router] whitelist global enable
#IPv4アドレスオブジェクトグループobj1を作成します。サブネット5.5.5.0/24でIPv4アドレスオブジェクトを設定します。
[Router] object-group ip address obj1
[Router-obj-grp-ip-obj1] network subnet 5.5.5.0 24
[Router] quit
#IPv4アドレスオブジェクトグループobj1をホワイトリストに追加します。
[Router] whitelist object-group obj1

設定の確認

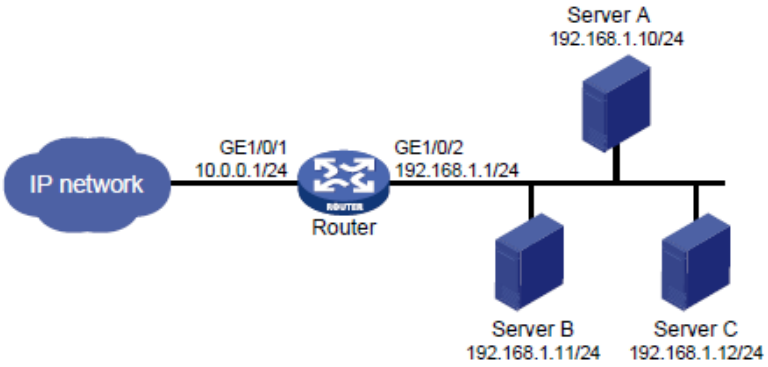
#ルーターでundo whitelist object-groupコマンドを実行しない限り、サブネット5.5.5.0/24からのすべてのパケットがルーターを通過できることを確認します(詳細は省略)。

例:インターフェイスベースのTCPクライアント検証の設定

ネットワーク構成

図15に示すように、SYNフラッド攻撃から内部サーバーを保護するために、ルーター上でSYNクッキーモードでTCPクライアント検証を設定します。

図15 ネットワークダイアグラム



手順

#ルーターのインターフェイスにIPアドレスを設定します。(詳細は省略)
#攻撃防御ポリシーa1を作成します。
<Router> system-view
[Router] attack-defense policy a1

#グローバルSYNフラッド攻撃検出を有効にします。

[Router-attack-defense-policy-a1] dns-flood detect non-specific

#SYNフラッド攻撃防止をトリガーするグローバルしきい値を10000に設定します。

[Router-attack-defense-policy-a1] dns-flood threshold 10000

#SYNフラッド攻撃に対するグローバルアクションとして、loggingとclient-verifyを指定します。

[Router-attack-defense-policy-a1] dns-flood action logging client-verify

[Router-attack-defense-policy-a1] quit

#攻撃防御ポリシーa1をGigabitEthernet 1/0/1に適用します。
[Router] interface gigabitethernet 1/0/1
[Router-GigabitEthernet1/0/1] attack-defense apply policy a1

[Router-GigabitEthernet1/0/1] quit

#GigabitEthernet 1/0/1でSYNクッキーモードのTCPクライアント検証をイネーブルにします。
[Router] interface gigabitethernet 1/0/1
[Router-GigabitEthernet1/0/1] client-verify dns enable

[Router-GigabitEthernet1/0/1] quit

設定の確認

#SYNフラッド攻撃を開始します(詳細は省略)。

#被害者のIPアドレスが、TCPクライアントの検証用の保護されたIPリストに追加されていることを確認します。
[Router] display client-verify dns protected ip

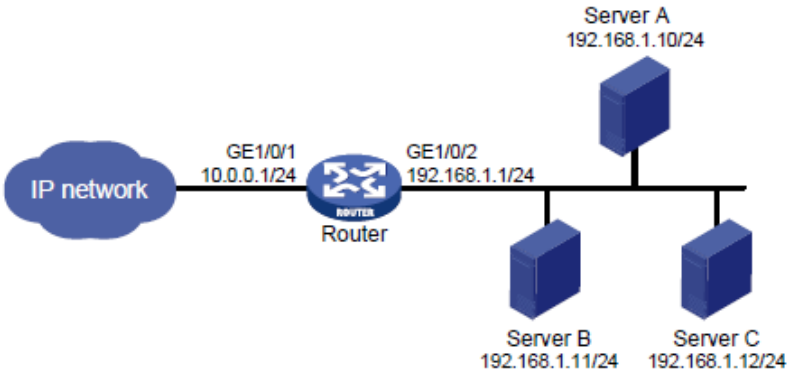
IP address	VPN instance	Port	Type	Requested	Trusted
192.168.1.10	--	53	Dynamic	20	12

例:インターフェイスベースのDNSクライアント検証の設定

ネットワーク構成

図16に示すように、DNSフラッド攻撃から内部サーバーを保護するために、ルーターでDNSクライアント検証を設定します。

図16 ネットワークダイアグラム



手順

#ルーターのインターフェイスにIPアドレスを設定します。(詳細は省略)
#攻撃防御ポリシーa1を作成します。
<Router> system-view
[Router] attack-defense policy a1
#グローバルDNSフラッド攻撃検出を有効にします。
[Router-attack-defense-policy-a1] http-flood detect non-specific
#DNSフラッド攻撃防止をトリガーするグローバルしきい値を10000に設定します。
[Router-attack-defense-policy-a1] http-flood threshold 10000
#DNSフラッド攻撃に対するグローバルアクションとして、loggingとclient-verifyを指定します。
[Router-attack-defense-policy-a1] http-flood action logging client-verify
[Router-attack-defense-policy-a1] quit
#攻撃防御ポリシーa1をGigabitEthernet 1/0/1に適用します。
[Router] interface gigabitethernet 1/0/1
[Router-GigabitEthernet1/0/1] attack-defense apply policy a1
[Router-GigabitEthernet1/0/1] quit
#GigabitEthernet 1/0/1でDNSクライアント検証をイネーブルにします。
[Router] interface gigabitethernet 1/0/1
[Router-GigabitEthernet1/0/1] client-verify http enable
[Router-GigabitEthernet1/0/1] quit
#DNSフラッド攻撃を開始します(詳細は省略)。
#被害者のIPアドレスが、DNSクライアントの検証のために保護されたIPリストに追加されていることを確認します。
[Router] display client-verify http protected ip

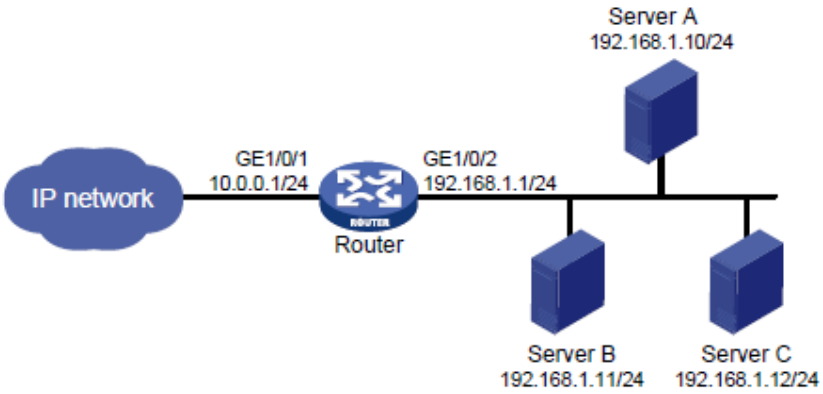
IP address	VPN instance	Port	Type	Requested	Trusted
192.168.1.10	--	8080	Dynamic	20	12

例:インターフェイスベースのHTTPクライアント検証の設定

ネットワーク構成

図17に示すように、HTTPフラッド攻撃から内部サーバーを保護するために、ルーターでHTTPクライアント検証を設定します。

図17 ネットワークダイアグラム



手順

#ルーターのインターフェイスにIPアドレスを設定します。(詳細は省略)
#攻撃防御ポリシーa1を作成します。
<Router> system-view
[Router] attack-defense policy a1

#グローバルHTTPフラッド攻撃検出をイネーブルにします。

[Router-attack-defense-policy-a1] http-flood detect non-specific

#HTTPフラッド攻撃防御をトリガーするためのグローバルしきい値を10000に設定します。

[Router-attack-defense-policy-a1] http-flood threshold 10000

#HTTPフラッド攻撃に対するグローバルアクションとして、loggingとclient-verifyを指定します。

[Router-attack-defense-policy-a1] http-flood action logging client-verify

[Router-attack-defense-policy-a1] quit

#攻撃防御ポリシーa1をGigabitEthernet 1/0/1に適用します。
[Router] interface gigabitethernet 1/0/1
[Router-GigabitEthernet1/0/1] attack-defense apply policy a1

[Router-GigabitEthernet1/0/1] quit

#GigabitEthernet 1/0/1でHTTPクライアント検証をイネーブルにします。
[Router] interface gigabitethernet 1/0/1
[Router-GigabitEthernet1/0/1] client-verify http enable

[Router-GigabitEthernet1/0/1] quit

#HTTPフラッド攻撃を開始します(詳細は省略)。

#被害者のIPアドレスが、HTTPクライアント検証用の保護されたIPリストに追加されていることを確認します。
[Router] display client-verify http protected ip

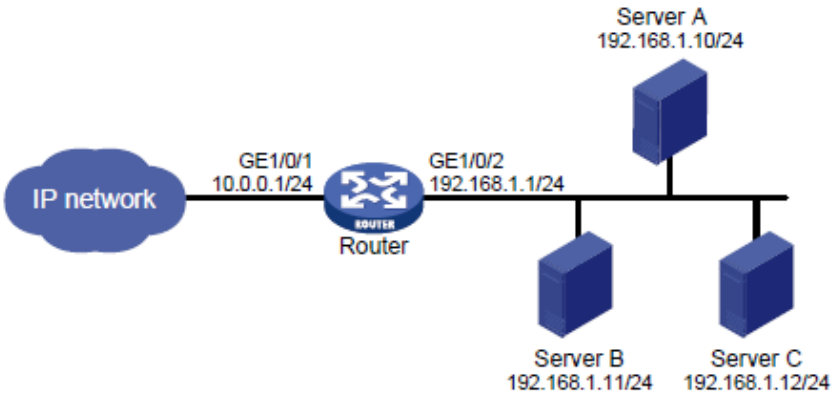
IP address	VPN instance	Port Type	Requested	Trusted
192.168.1.10	--	8080 Dynamic	20	12

例:インターフェイスベースのSIPクライアント検証の設定

ネットワーク構成

図18に示すように、SIPフラッド攻撃から内部サーバーを保護するために、ルーターでSIPクライアント検証を設定します。

図18 ネットワークダイアグラム



手順

#ルーターのインターフェイスにIPアドレスを設定します。(詳細は省略)
#攻撃防御ポリシーa1を作成します。
<Router> system-view
[Router] attack-defense policy a1
#グローバルSIPフラッド攻撃検出をイネーブルにします。
[Router-attack-defense-policy-a1] sip-flood detect non-specific
#SIPフラッド攻撃防御をトリガーするためのグローバルしきい値を10000に設定します。
[Router-attack-defense-policy-a1] sip-flood threshold 10000
#SIPフラッド攻撃に対するグローバルアクションとして、loggingとclient-verifyを指定します。
[Router-attack-defense-policy-a1] sip-flood action logging client-verify
[Router-attack-defense-policy-a1] quit
#攻撃防御ポリシーa1をGigabitEthernet 1/0/1に適用します。
[Router] interface gigabitethernet 1/0/1
[Router-GigabitEthernet1/0/1] attack-defense apply policy a1
#GigabitEthernet 1/0/1でSIPクライアント検証をイネーブルにします。
[Router-GigabitEthernet1/0/1] client-verify sip enable
[Router-GigabitEthernet1/0/1] quit

設定の確認

#SIPフラッド攻撃を開始します(詳細は省略)。

#被害者のIPアドレスが、SIPクライアント検証用の保護されたIPリストに追加されていることを確認します。
[Router] display client-verify sip protected ip
IP address VPN instance Port Type Requested Trusted

192.168.1.10 -- 5060 Dynamic 20 12

IPソースガードの設定

IPSGについて

IP Source Guard(IPSG;IPソースガード)は、IPSGバインディングテーブルを使用して不正なパケットをフィルタリングすることで、スプーフィング攻撃を防止します。この機能は通常、ユーザー側のインターフェイスで設定されます。

IPSG動作メカニズム

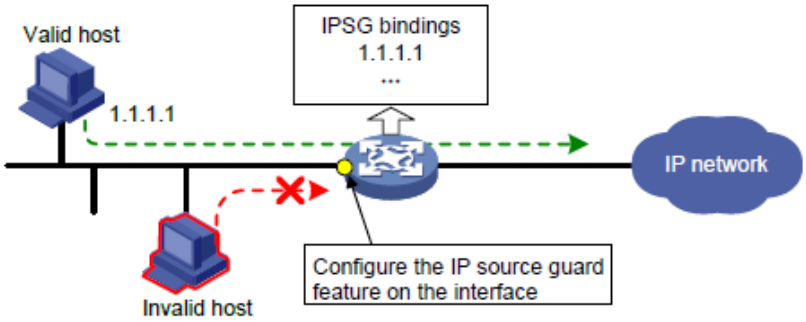
IPSGバインディングテーブルには、IPアドレス、MACアドレス、または任意の組み合わせをバインドするバインディングが含まれています。IPSGはバインディングを使用して着信パケットを照合します。一致が見つかった場合、パケットは転送されます。一致が見つからなかった場合、パケットは廃棄されます。

IPSGは、インターフェイス単位のパケットフィルターです。あるインターフェイスでこの機能を設定しても、別のインターフェイスでのパケット転送には影響しません。

IPSGバインディングには、スタティックバインディングとダイナミックバインディングがあります。

図19に示すように、IPSGはIPSGバインディングに一致するパケットだけを転送します。

図19 IPSGアプリケーション



スタティックIPSGバインディング

スタティックIPSGバインディングは、手動で設定します。LAN上に存在するホストが少なく、そのIPアドレスが手動で設定されているシナリオに適しています。たとえば、サーバーに接続するインターフェイスにスタティックIPSGバインディングを設定できます。このバインディングにより、インターフェイスはサーバーからのパケットだけを受信できます。

インターフェイスのスタティックIPSGバインディングには、次の機能が実装されています。

- インターフェイス上の着信IPv4またはIPv6パケットをフィルタリングします。
- IPv4のARP攻撃検出と連携して、ユーザーの有効性をチェックします。ARP攻撃検出の詳細については、「ARP攻撃保護の構成」を参照してください。

スタティックIPSGバインディングは、IPアドレス、MACアドレス、またはインターフェイスビュー内の項目の任意の組み合わせをバインドします。このバインディングは、インターフェイスにアクセスしようとしているユーザーの有効性を確認するために、インターフェイス上でのみ有効になります。

ダイナミックIPSGバインディング

IPSGは、他のモジュールからユーザー情報を自動的に取得して、ダイナミックバインディングを生成します。ダイナミックIPSGバインディングには、MACアドレス、IPv4またはIPv6アドレス、入力インターフェイス、およびバインディングタイプがあります。バインディングタイプは、DHCPスヌーピングやDHCPv6スヌーピングなど、バインディングの送信元モジュールを識別します。

たとえば、DHCPベースのIPSGバインディングは、LAN上のホストがDHCPを介してIPアドレスを取得するシナリオに適しています。ホストがDHCPを介してIPアドレスを取得すると、DHCPスヌーピングはDHCPスヌーピングエントリを生成します。IPSGは、上記のエントリに基づいてダイナミックIPSGバインディングを生成します。IPSGは、DHCPクライアントからのパケットだけを通過させます。

ダイナミックIPv4SG

さまざまなソースモジュールに基づいて生成された動的バインディングは、さまざまな用途に使用されます。

インターフェイスの種類	ソースモジュール	バインドの使用方法
レイヤ2イーサネットインターフェイス	DHCPスヌーピング802.1X	パケットフィルタリング。

WLANネットワークでは、IPSGはモジュール(ARP攻撃検出モジュールなど)のWLANスヌーピングに基づいてバインディングを生成し、セキュリティサービスを提供できます。

802.1Xの詳細については、「802.1Xの設定」を参照してください。DHCPスヌーピングの詳細については、『Layer 3 IP Services Configuration Guide』の「DHCP snooping configuration」を参照してください。

ダイナミックIPv6SG

さまざまなソースモジュールに基づいて生成されたダイナミックIPv6SGバインディングは、さまざまな用途に使用されます。

インターフェイスの種類	ソースモジュール	バインドの使用方法
レイヤ2イーサネットインターフェイス	DHCPv6スヌーピング802.1X	パケットフィルタリング。

WLANネットワークでは、IPv6SGはモジュール(ND攻撃検出モジュールなど)のWLANスヌーピングに基づいてバインディングを生成し、セキュリティサービスを提供できます。

DHCPv6スヌーピングの詳細については、『Layer 3 IP Services Configuration Guide』を参照してください。

制約事項:IPソースガードとのハードウェア互換性

次のマトリックスは、ハードウェアとスタティックIPv4SGの互換性を示しています。

ハードウェア	備考
• 次のルーターの固定レイヤ2イーサネットポート: ○ MSR610、MSR810、MSR810-W、MSR810-W-DB、MSR810-LM、MSR810-W-LM、MSR810-10-PoE、MSR810-LM-HK MSR810-W-LM-HK、MSR810-LM-CNDE-SJK、MSR810-EI、MSR810-LM-EA、MSR810-LM-EI、MSR810-LMS、MSR810-LUS、MSR810-SI、MSR810-LM-SI、MSR810-LMS-EA、MSR810-LME ○ MSR2600-6-X1、MSR2600-15-X1、MSR2600-15-X1-T、MSR2600-10-X1 ○ MSR3600-28、MSR3600-51、MSR3600-28-SI、MSR3600-51-SI、MSR3600-28-X1、MSR3600-28-X1-DP、MSR3600-51-X1、MSR3600-51-X1-DP、MSR3600-28-G-DP、MSR3600-51-G-DP ○ MSR3610-I-DP、MSR3610-IE-DP、MSR3610-IE-ES、MSR3610-IE-EAD、MSR3610-I-IG、MSR3610-IE-IG 試験 ○ MSR810-W-WiNet、MSR810-LM-WiNet、MSR830-4LM-WiNet、MSR830-5BEI-WiNet、MSR830-6EI-WiNet	レイヤ2インターフェイスモジュールに対するルーターのサポートについては、『H3C MSR Router Series Comware 7 Interface Module Guide』を参照してください。

ハードウェア	備考

MSR830-10BEI-WiNet、MSR830-6BHI-WiNet、MSR830-10BHI-WiNet、MSR2600-6-WiNet、MSR2600-10-X1-WiNet、MSR3600-28-WiNet ○ MSR2630-XS、MSR3600-28-XS、MSR3610-I-XS、MSR3610-IE-XS ○ MSR810-LM-GL、MSR810-W-LM-GL、MSR830-6EI-GL、MSR830-10EI-GL、MSR830-6HI-GL、MSR830-10HI-GL、MSR2600-6-X1-GL、MSR3600-28-SI-GL ○ MSR-EAD-AK770 • ルーターにインストールされている次のレイヤ2インターフェイスモジュール。 ○ HMIM-24GSW ○ HMIM-24GSWP ○ HMIM-8GSW ○ HMIM-8GSWF ○ SIC-4GSW ○ SIC-4GSWF ○ SIC-4GSWP	

次のマトリックスは、ハードウェアとスタティックIPv6SGの互換性を示しています。

ハードウェア	備考
• 次のルーターの固定レイヤ2イーサネットポート: ○ MSR610、MSR810、MSR810-W、MSR810-W-DB、MSR810-LM、MSR810-W-LM、MSR810-10-PoE、MSR810-LM-HK MSR810-W-LM-HK、MSR810-LM-CNDE-SJK、MSR810-EI、MSR810-LM-EA、MSR810-LM-EI、MSR810-LMS、MSR810-LUS、MSR810-SI、MSR810-LM-SI、MSR810-LMS-EA、MSR810-LME ○ MSR2600-6-X1、MSR2600-15-X1、MSR2600-15-X1-T、MSR2600-10-X1 ○ MSR3600-28、MSR3600-51、MSR3600-28-X1 MSR3600-28-X1-DP、MSR3600-51-X1、MSR3600-51-X1-DP、MSR3600-28-G-DP、MSR3600-51-G-DP ○ MSR3610-I-DP、MSR3610-IE-DP、MSR3610-IE-ES、MSR3610-IE-EAD、MSR3610-I-IG、MSR3610-IE-IG ○ MSR810-W-WiNet、MSR810-LM-WiNet、MSR830-4LM-WiNet、MSR830-5BEI-WiNet、MSR830-6EI-WiNet MSR830-10BEI-WiNet、MSR830-6BHI-WiNet、MSR830-10BHI-WiNet、MSR2600-6-WiNet MSR2600-10-X1-WiNet、MSR2630-WiNet、MSR3600-28-WiNet ○ MSR2630-XS、MSR3600-28-XS、MSR3610-I-XS、MSR3610-IE-XS ○ MSR810-LM-GL、MSR810-W-LM-GL、MSR830-6EI-GL、MSR830-10EI-GL、MSR830-6HI-GL、MSR830-10HI-GL、MSR2600-6-X1-GL ○ MSR-EAD-AK770 • ルーターにインストールされている次のレイヤ2インターフェイスモジュール。 ○ HMIM-24GSW ○ HMIM-24GSWP ○ HMIM-8GSW ○ HMIM-8GSWF ○ SIC-4GSW ○ SIC-4GSWF ○ SIC-4GSWP	レイヤ2インターフェイスモジュールに対するルーターのサポートについては、『H3C MSR Router Series Comware 7 Interface Module Guide』を参照してください。

次のマトリックスは、ハードウェアとダイナミックIPv4SGの互換性を示しています。

ハードウェア	備考

• 次のルーターの固定レイヤ2イーサネットポート:
  ○ MSR3600-28、MSR3600-51、MSR3600-28-SI、MSR3600-51-SI、MSR3600-28-X1、MSR3600-28-X1-DP、MSR3600-51-X1、MSR3600-51-X1-DP、MSR3600-28-G-DP、MSR3600-51-G-DP、MSR3600-28-WiNet、MSR3600-28-XS、MSR3600-28-SI-GL
• ルーターにインストールされている次のレイヤ2インターフェイスモジュール。
  ○ HMIM-24GSW
  ○ HMIM-24GSWP
  ○ HMIM-8GSW
  ○ HMIM-8GSWF
	レイヤ2インターフェイスモジュールに対するルーターのサポートについては、『H3C MSR Router Series Comware 7 Interface Module Guide』を参照してください。

次のマトリックスは、ハードウェアとダイナミックIPv6SGの互換性を示しています。

ハードウェア	備考

• 次のルーターの固定レイヤ2イーサネットポート:
  ○ MSR3600-28、MSR3600-51、MSR3600-28-X1 MSR3600-28-X1-DP、MSR3600-51-X1、MSR3600-51-X1-DP、MSR3600-28-G-DP、MSR3600-51-G-DP、MSR3600-28-WiNet、MSR3600-28-XS
• ルーターにインストールされている次のレイヤ2インターフェイスモジュール。
  ○ HMIM-24GSW
  ○ HMIM-24GSWP
  ○ HMIM-8GSW
  ○ HMIM-8GSWF
	レイヤ2インターフェイスモジュールに対するルーターのサポートについては、『H3C MSR Router Series Comware 7 Interface Module Guide』を参照してください。

IPSGタスクの概要

IPv4SGを設定するには、次の作業を実行します。

1. インターフェイスでのIPv4SGのイネーブル化
2. (省略可能)静的IPv4SGバインディングの構成

IPv6SGを構成するには、次の作業を行います。

1. インターフェイスでのIPv6SGのイネーブル化
2. (オプション)スタティックIPv6SGバインディングの設定

IPv4SG機能の設定

インターフェイスでのIPv4SGのイネーブル化

このタスクについて

インターフェイスでIPSGをイネーブルにすると、スタティックIPSGとダイナミックIPSGの両方がイネーブルになります。

- スタティックIPv4SGは、ip source bindingコマンドを使用して設定されたスタティックバインディングを使用します。詳細については、「スタティックIPv4SGバインディングの設定」を参照してください。
- ダイナミックIPv4SGは、関連するソースモジュールからダイナミックバインディングを生成します。IPv4SGは、バインディングを使用して、ip verify sourceコマンドで指定された一致基準に

基づいて着信IPv4パケットをフィルタリングします。

ダイナミックIPv4SGを実装するには、802.1X、DHCPスヌーピング、またはWLANスヌーピングがネットワーク上で正常に動作することを確認します。

手順

1. システムビューに入ります。
   system-view
2. インターフェイスビューを入力します。
   interface *interface-type interface-number*
   レイヤ2イーサネットインターフェイスだけがサポートされています。
3. IPv4SG機能をイネーブルにします。
   ip verify source **{** ip-address **|** ip-address mac-address **|** mac-address
   デフォルトでは、IPv4SG機能はインターフェイスでディセーブルになっています。

静的IPv4SGバインディングの設定

制限事項およびガイドライン

ARP 攻撃検出機能の静的 IPv4SG バインディングを設定するには、**ip-address** *ip-address* オプションと **mac-address** *mac-address* オプションを指定する必要があります。

インターフェイスでのスタティックIPv4SGバインディングの設定

1. システムビューに入ります。
   system-view
2. インターフェイスビューを入力します。
   interface *interface-type interface-number*
   レイヤ2イーサネットインターフェイスだけがサポートされています。
3. スタティックIPv4SGバインディングを設定します。
   ip source binding { **ip-address** *ip-address* | **ip-address** *ip-address* **mac-address** *mac-address* | **mac-address** *mac-address* }
   異なるインターフェイスに同じスタティックIPv4SGバインディングを設定できます。
   ip-addressおよびmac-addressパラメータのサポートは、デバイスモデルによって異なります。詳細については、コマンドリファレンスを参照してください。

IPv6SG機能の設定

インターフェイスでのIPv6SGのイネーブル化

このタスクについて

インターフェイスでIPv6SGをイネーブルにすると、スタティックIPv6SGとダイナミックIPv6SGの両方がイネーブルになります。

- スタティックIPv6SGは、ipv6ソースバインディングを使用して設定されたスタティックバインディングを使用します。
  コマンドを使用します。詳細については、「スタティックIPv6SGバインディングの設定」を参照してください。
- ダイナミックIPv6SGは、関連する送信元モジュールからダイナミックバインディングを生成しま

す。IPv6SGはバインディングを使用して、ipv6 verify sourceコマンドで指定された一致基準に基づいて着信ipv6パケットをフィルタリングします。

ダイナミックIPv6SGを実装するには、DHCPv6スヌーピングまたはWLANスヌーピングがネットワーク上で正しく動作することを確認します。

手順

1. システムビューに入ります。

   system-view
2. インターフェイスビューを入力します。

   interface *interface-type interface-number*

   レイヤ2イーサネットインターフェイスだけがサポートされています。
3. IPv6SG機能をイネーブルにします。
   ipv6 verify source { **ip-address** | **ip-address mac-address** | mac-address **}**

   デフォルトでは、IPv6SG機能はインターフェイスでディセーブルになっています。

静的IPv6SGバインディングの設定

インターフェイスでのスタティックIPv6SGバインディングの設定

1. システムビューに入ります。

   system-view
2. インターフェイスビューを入力します。

   interface *interface-type interface-number*

   レイヤ2イーサネットインターフェイスだけがサポートされています。
3. スタティックIPv6SGバインディングを設定します。
   ipv6 source binding { **ip-address** *ipv6-address* | **ip-address** *ipv6-address* **mac-address** *mac-address* | **mac-address** *mac-address* }

   異なるインターフェイスに同じスタティックIPv6SGバインディングを設定できます。

   ip-addressおよびmac-addressパラメータのサポートは、デバイスモデルによって異なります。詳細については、コマンドリファレンスを参照してください。

IPSGの表示コマンドおよびメンテナンスコマンド

任意のビューでdisplayコマンドを実行し、ユーザービューでコマンドをリセットします。

タスク	コマンド

IPv4SGバインディングを表示します。	スタンドアロンモードの場合: **display ip source binding** [**static** | [**vpn-instance** *vpn-instance-name*] [**dhcp-relay** | **dhcp-server** | **dhcp-snooping** | **dot1x** | **wlan-snooping**]] [**ip-address** *ip-address*] [**mac-address** *mac-address*] [**vlan** *vlan-id*] [**interface** *interface-type interface-number*] IRFモードの場合: **display ip source binding** [**static** | [**vpn-instance** *vpn-instance-name*] [**dhcp-relay** | **dhcp-server** | **dhcp-snooping** | **dot1x** | **wlan-snooping**]] [**ip-address** *ip-address*] [**mac-address** *mac-address*] [**vlan** *vlan-id*] [**interface** *interface-type interface-number*] [**slot** *slot-number*]

タスク	コマンド
IPv6SGバインディングを表示します。	スタンドアロンモードの場合: **display ipv6 source binding** [**static** | [**vpn-instance** *vpn-instance-name*] [**dhcpv6-snooping** | **wlan-snooping**]] [**ip-address** *ipv6-address*] [**mac-address** *mac-address*] [**vlan** *vlan-id*] [**interface** *interface-type interface-number*] IRFモードの場合: **display ipv6 source binding** [**static** | [**vpn-instance** *vpn-instance-name*] [**dhcpv6-snooping** | **wlan-snooping**]] [**ip-address** *ipv6-address*] [**mac-address** *mac-address*] [**vlan** *vlan-id*] [**interface** *interface-type interface-number*] [**slot** *slot-number*]

IPSGの設定例

例:スタティックIPv4SGの設定

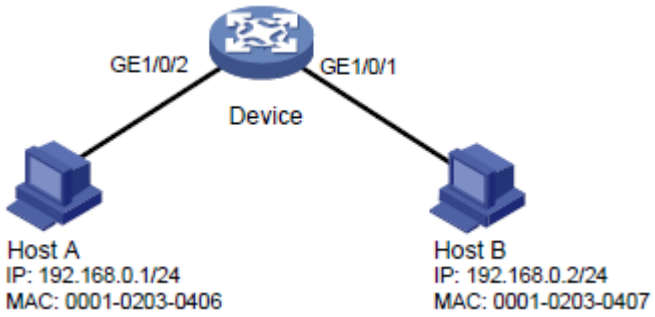
ネットワーク構成

図20に示すように、すべてのホストは静的IPアドレスを使用します。

次の要件を満たすように、デバイスでスタティックIPv4SGバインディングを設定します。

- デバイスのすべてのインターフェイスは、ホストAからのIPパケットの通過を許可します。
- デバイスのGigabitEthernet 1/0/1は、ホストBからのIPパケットの通過を許可します。

図20 ネットワークダイアグラム



手順

#インターフェイスのIPアドレスを設定します(詳細は省略)。

#GigabitEthernet 1/0/2でIPv4SGをイネーブルにします。

```
<Device> system-view
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] ip verify source ip-address mac-address
[Device-GigabitEthernet1/0/2] quit
```

#ホストAの静的IPv4SGバインディングを設定します。

```
[Device] ip source binding ip-address 192.168.0.1 mac-address 0001-0203-0406
```

#GigabitEthernet 1/0/1でIPv4SGをイネーブルにします。

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip verify source ip-address mac-address
```

#GigabitEthernet 1/0/1で、ホストBのスタティックIPv4SGバインディングを設定します。

```
[Device-GigabitEthernet1/0/1] ip source binding mac-address 0001-0203-0407
[Device-GigabitEthernet1/0/1] quit
```

設定の確認

#スタティックIPv4SGバインディングがデバイスで正常に設定されていることを確認します。

```
<Device> display ip source binding static
Total entries found: 2
IP Address   MAC Address        Interface      VLAN   Type
192.168.0.1  0001-0203-0406     N/A            N/A    Static
N/A          0001-0203-0407     GE1/0/1        N/A    Static
```

例:DHCPスヌーピングベースのダイナミックIPv4SGの設定

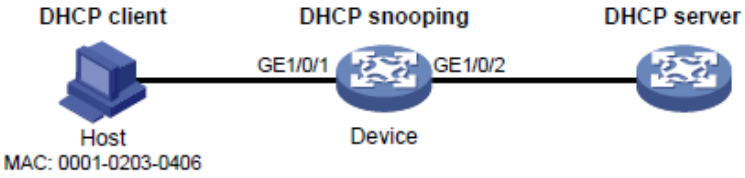
ネットワーク構成

図21に示すように、ホスト(DHCPクライアント)はDHCPサーバーからIPアドレスを取得します。次のタスクを実行します。

- デバイスでDHCPスヌーピングをイネーブルにして、DHCPクライアントが認可されたDHCPサーバーからIPアドレスを取得するようにします。DHCPクライアントのDHCPスヌーピングエントリを生成するには、DHCPスヌーピングエントリでクライアント情報の記録をイネーブルにします。
- DHCPスヌーピングエントリに基づいて生成されたIPv4SGバインディングを使用して着信パケットをフィルタリングするには、GigabitEthernet 1/0/1でダイナミックIPv4SGをイネーブルに

します。DHCPクライアントからのパケットだけが通過を許可されます。

図21 ネットワークダイアグラム



手順

1. DHCPサーバーを設定します。

   DHCPサーバー設定の詳細については、『Layer 3 IP Services Configuration Guide』を参照してください。

2. デバイスを設定します。

   #インターフェイスのIPアドレスを設定します(詳細は省略)。

   #DHCPスヌーピングを有効にします。

   <Device> system-view

   [Device] dhcp snooping enable

   #GigabitEthernet 1/0/2を信頼できるインターフェイスとして設定します。

   [Device] interface gigabitethernet 1/0/2

   [Device-GigabitEthernet1/0/2] dhcp snooping trust

   [Device-GigabitEthernet1/0/2] quit

   #GigabitEthernet 1/0/1でIPv4SGをイネーブルにし、ダイナミックIPSGの送信元IPアドレスとMACアドレスを確認します。

   [Device] interface gigabitethernet 1/0/1

   [Device-GigabitEthernet1/0/1] ip verify source ip-address mac-address

   #GigabitEthernet 1/0/1のDHCPスヌーピングエントリで、クライアント情報の記録をイネーブルにします。

   [Device-GigabitEthernet1/0/1] dhcp snooping binding record

   [Device-GigabitEthernet1/0/1] quit

設定の確認

#DHCPスヌーピングエントリに基づいて生成されたダイナミックIPv4SGバインディングを表示します。

[Device] display ip source binding dhcp-snooping

Total entries found: 1

IP Address	MAC Address	Interface	VLAN	Type
192.168.0.1	0001-0203-0406	GE1/0/1	1	DHCP snooping

この出力は、GigabitEthernet 1/0/1がIPv4SGバインディングに基づいてパケットをフィルタリングすることを示しています。

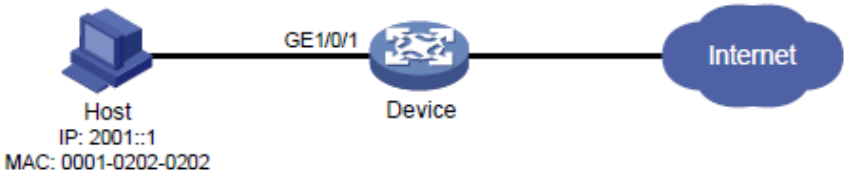
例:スタティックIPv6SGの設定

ネットワーク構成

図22に示すように、デバイスのGigabitEthernet 1/0/1にスタティックIPv6SGバインディングを設定し

て、ホストからのIPv6パケットだけが通過できるようにします。

図22 ネットワークダイアグラム



手順

#GigabitEthernet 1/0/1でIPv6SGをイネーブルにします。
<Device> system-view
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ipv6 verify source ip-address mac-address
#GigabitEthernet 1/0/1で、ホストのスタティックIPv6SGバインディングを設定します。
[Device-GigabitEthernet1/0/1] ipv6 source binding ip-address 2001::1 mac-address 0001-0202-0202
[Device-GigabitEthernet1/0/1] quit

設定の確認

#スタティックIPv6SGバインディングがデバイスで正常に設定されていることを確認します。
[Device] display ipv6 source binding static
Total entries found: 1

IPv6 Address	MAC Address	Interface	VLAN	Type
2001::1	0001-0202-0202	GE1/0/1	N/A	Static

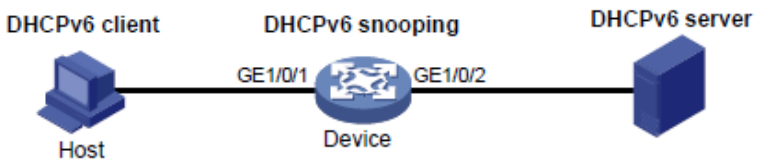
例:DHCPv6スヌーピングベースのダイナミックIPv6SGバインディングの設定

ネットワーク構成

図23に示すように、ホスト(DHCPv6クライアント)はDHCPv6サーバーからIPアドレスを取得します。次のタスクを実行します。

- デバイス上でDHCPv6スヌーピングをイネーブルにして、DHCPv6クライアントが認可されたDHCPv6サーバーからIPv6アドレスを取得するようにします。DHCPv6クライアントのDHCPv6スヌーピングエントリを生成するには、DHCPv6スヌーピングエントリ内のクライアント情報の記録をイネーブルにします。
- DHCPv6スヌーピングエントリに基づいて生成されたIPv6SGバインディングを使用して着信パケットをフィルタリングするには、GigabitEthernet 1/0/1でダイナミックIPv6SGをイネーブルにします。DHCPv6クライアントからのパケットだけが通過を許可されます。

図23 ネットワークダイアグラム



手順

1. DHCPv6スヌーピングを設定します。

#DHCPv6スヌーピングをグローバルに有効にします。
<Device> system-view
[Device] ipv6 dhcp snooping enable
#GigabitEthernet 1/0/2を信頼できるインターフェイスとして設定します。
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] ipv6 dhcp snooping trust
[Device-GigabitEthernet1/0/2] quit

2. IPv6SGを有効にします。
#GigabitEthernet 1/0/1でIPv6SGをイネーブルにし、ダイナミックIPv6SGの送信元IPアドレスとMACアドレスを確認します。
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ipv6 verify source ip-address mac-address
#GigabitEthernet 1/0/1のDHCPv6スヌーピングエントリで、クライアント情報の記録をイネーブルにします。
[Device-GigabitEthernet1/0/1] ipv6 dhcp snooping binding record
[Device-GigabitEthernet1/0/1] quit

設定の確認

#DHCPv6スヌーピングエントリに基づいて生成されたダイナミックIPv6SGバインディングを表示します。
[Device] display ipv6 source binding dhcpv6-snooping
Total entries found: 1

IPv6 Address	MAC Address	Interface	VLAN	Type
2001::1	040a-0000-0001	GE1/0/1	1	DHCPv6 snooping

この出力は、GigabitEthernet 1/0/1がIPv6SGバインディングに基づいてパケットをフィルタリングすることを示しています

ARP攻撃からの保護の設定

ARP攻撃からの保護について

デバイスは、LAN内のARP攻撃およびウイルスを検出して防止するための複数の機能を提供できます。攻撃者は、ARPの脆弱性を不正利用して、次の方法でネットワークデバイスを攻撃できます。

- CPUが過負荷になるまで、IPアドレスの解決で受信デバイスをビジー状態にするために、解決不可能なIPパケットを大量に送信します。解決不可能なIPパケットとは、ARPが対応するMACアドレスを検出できないIPパケットのことです。
- 大量のARPパケットを送信して、受信デバイスのCPUを過負荷にします。
- 信頼されたユーザーまたはゲートウェイとしてARPパケットを送信し、受信側デバイスが不正なARPエントリを取得できるようにします。

ARP攻撃からの保護タスクの概要

すべてのARP攻撃保護タスクはオプションです。

- 洪水攻撃の防止
  - 解決不可能なIP攻撃からの保護の設定
  - 送信元MACベースARP攻撃検出の設定
- ユーザーおよびゲートウェイのスプーフィング攻撃の防止
  - ARPパケットの送信元MAC整合性チェックの設定

- ARPアクティブ確認応答の設定
- 許可ARPの設定
- ARP攻撃検出の設定
- ARPスキャンおよび固定ARPの設定
- ARPゲートウェイ保護の設定
- ARPフィルタリングの設定

解決不可能なIP攻撃からの保護の設定

解決不可能なIP攻撃からの保護について

デバイスがホストから解決できないIPパケットを大量に受信した場合、次の状況が発生する可能性があります。

- デバイスは大量のARP要求を送信し、ターゲットサブネットに過負荷をかけます。
- デバイスは宛先IPアドレスを解決しようとし続け、CPUに過負荷をかけます。このよう

なIP攻撃からデバイスを保護するために、次の機能を設定できます。

- **ARP source suppression**: IPアドレスからのパケットの解決を停止します。
  IPアドレスからの解決不可能なIPパケットが、5秒以内に上限を超えました。デバイスは、間隔が経過するとARP解決を継続します。この機能は、攻撃パケットの送信元アドレスが同じ場合に適用されます。
- **ARP blackhole routing**: 未解決のIPアドレスを宛先とするブラックホールルートを作成します。デバイスは、ブラックホールルートが削除されるまで、一致するすべてのパケットをドロップします。ブラックホールルートは、エージングタイマーに達したとき、またはルートが到達可能になったときに削除されます。

  未解決のIPアドレスに対してブラックホールルートが作成された後、デバイスはARP要求を送信して最初のARPブラックホールルートプローブをただちに開始します。解決に失敗した場合、デバイスはプローブ設定に従ってプローブを続行します。IPアドレス解決がプローブで成功した場合、デバイスはブラックホールルートを通常のルートに変換します。デバイスがすべてのプローブを終了する前にARPブラックホールルートが期限切れになった場合、デバイスはブラックホールルートを削除し、残りのプローブは実行しません。

  この機能は、攻撃パケットの送信元アドレスが同じかどうかに関係なく適用できます。

ARP送信元抑制の設定

1. システムビューに入ります。

   system-view
2. ARP送信元抑制をイネーブルにします。

   arp source-suppression enable

   デフォルトでは、ARP送信元抑制はディセーブルです。
3. デバイスが送信元IPアドレスごとに5秒以内に処理できる解決不可能なパケットの最大数を設定します。

   arp source-suppression limit *limit-value*

   デフォルトでは、最大数は10です。

ARPブラックホールルーティングの設定

制限事項およびガイドライン

ARPブラックホールルートプローブカウントを25などの大きな値に設定します。デバイスが一時的に宛先IPアドレスに到達できず、プローブカウントが小さすぎる場合、問題が解決される前にすべてのプローブが終了する可能性があります。その結果、攻撃以外のパケットがドロップされます。この設定により、このような状況を回避できます。

手順

1. システムビューに入ります。
   system-view
2. ARPブラックホールルーティングをイネーブルにします。
   arp resolving-route enable
   デフォルトでは、ARPブラックホールルーティングはイネーブルです。
3. (任意)未解決のIPアドレスごとにARPブラックホールルートプローブの数を設定します。
   arp resolving-route probe-count ***count***
   デフォルトの設定は3つのプローブです。
4. (任意)デバイスがARPブラックホールルートをプローブする間隔を設定します。
   arp resolving-route probe-interval ***interval***
   デフォルト設定は1秒です。

解決不可能なIP攻撃から保護するための表示コマンドとメンテナンスコマンド

任意のビューで表示コマンドを実行します。

タスク	コマンド
ARP送信元抑制の設定情報を表示します。	display arp source-suppression

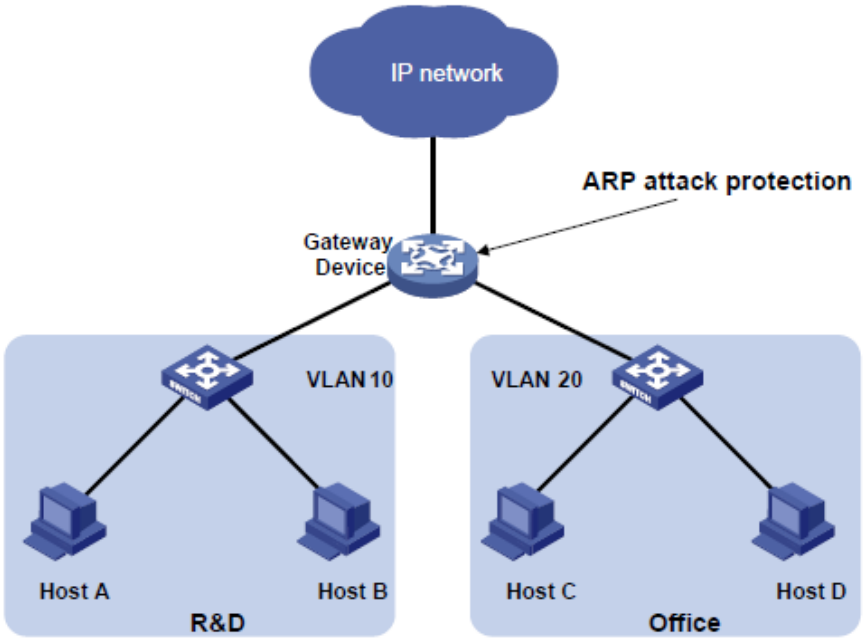
例:解決不可能なIP攻撃からの保護の設定

ネットワーク構成

図24に示すように、LANにはVLAN 10の研究開発エリアとVLAN 20のオフィスエリアの2つのエリアがあり、各エリアはアクセススイッチを介してゲートウェイ(デバイス)に接続されています。

多数のARP要求がオフィスエリアで検出され、解決不可能なIPパケットによって引き起こされた攻撃と見なされます。攻撃を防止するには、ARP送信元抑制またはARPブラックホールルーティングを設定します。

図24 ネットワークダイアグラム



手順

- 攻撃パケットの送信元アドレスが同じ場合は、ARP送信元抑制を設定します。
  #ARP送信元抑制を有効にします。
  <Device> system-view
  [Device] arp source-suppression enable
  #5秒以内に送信元IPアドレスごとに最大100個の解決できないパケットを処理するようにデバイスを設定します。
  [Device] arp source-suppression limit 100
- 攻撃パケットの送信元アドレスが異なる場合は、ARPブラックホールルーティングを設定します。
  #ARPブラックホールルーティングを有効にします。
  [Device] arp resolving-route enable

送信元MACベースARP攻撃検出の設定

送信元MACベースのARP攻撃検出について

この機能は、CPUに配信されたARPパケットの数をチェックします。5秒以内に同じMACアドレスからのパケット数がしきい値を超えた場合、デバイスはMACアドレスのARP攻撃エントリを生成します。ARPロギング機能がイネーブルの場合、デバイスは、ARP攻撃エントリが期限切れになる前に、次のいずれかの方法を使用して攻撃を処理します。

- **Monitor** :ログメッセージだけを生成します。
- **Filter** :ログメッセージを生成し、MACアドレスからの後続のARPパケットをフィルタリングします。

ARPロギング機能をイネーブルにするには、arp check log enableコマンドを使用します。ARPロギング機能の詳細については、『Layer 3 IP Services Configuration Guide』の「ARP configuration」を参照してください。

ARP攻撃エントリが期限切れになると、エントリ内のMACアドレスを送信元とするARPパケットが正しく処理されます。

制限事項およびガイドライン

処理方法をmonitorからfilterに変更すると、設定はただちに有効になります。処理方法をfilterからmonitorに変更すると、デバイスは既存の攻撃エントリに一致するパケットのフィルタリングを続行します。

一部のゲートウェイおよびサーバーのMACアドレスをこの検出から除外できます。この機能は、これらのデバイスが攻撃者であっても、これらのデバイスからのARPパケットを検査しません。

手順

1. システムビューに入ります。
   system-view
2. 送信元MACベースのARP攻撃検出をイネーブルにし、処理方法を指定します。
   arp source-mac **{** filter **|** monitor **}**
   デフォルトでは、この機能はディセーブルになっています。
3. しきい値を設定します。
   arp source-mac threshold *threshold-value*
   デフォルトでは、送信元MACベースARP攻撃検出のしきい値は30です。
4. ARP攻撃エントリのエージングタイマーを設定します。
   arp source-mac aging-time *time*
   デフォルトでは、タイムタイムは300秒です。
5. (任意)この検出から特定のMACアドレスを除外します。
   arp source-mac exclude-mac *mac-address*&<1-10>
   デフォルトでは、MACアドレスは除外されません。

送信元MACベースARP攻撃検出用の表示およびメンテナンスコマンド

任意のビューで表示コマンドを実行します。

タスク	コマンド
送信元MACベースARP攻撃検出によって検出されたARP攻撃エントリを表示します。	スタンドアロンモードの場合: **display arp source-mac** [**interface** *interface-type interface-number*] IRFモードの場合: **display arp source-mac** { **interface** *interface-type interface-number* | **slot** *slot-number* }

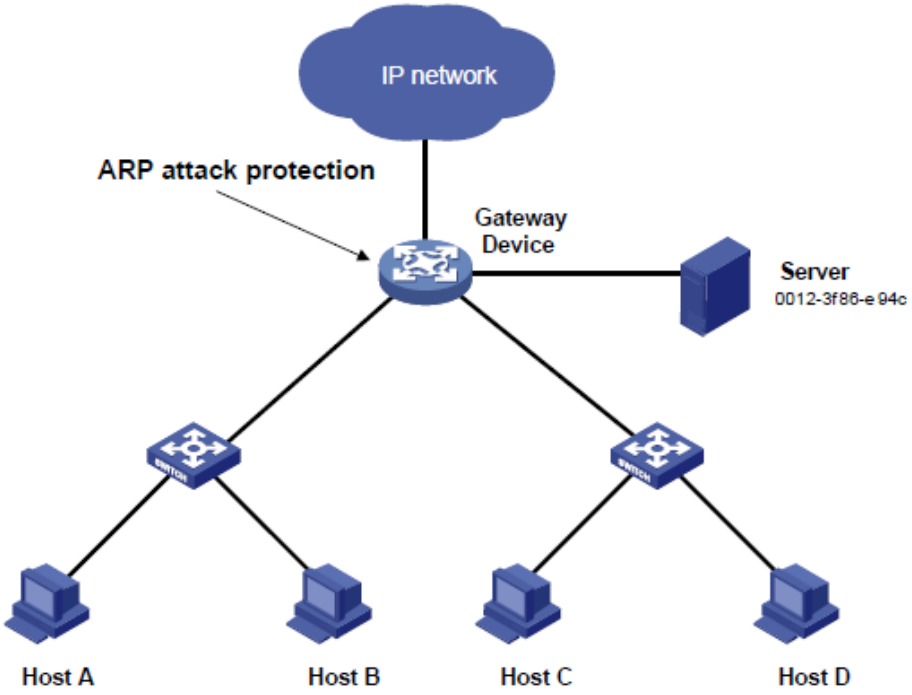
例:送信元MACベースARP攻撃検出の設定

ネットワーク構成

図25に示すように、ホストはゲートウェイ(デバイス)を介してインターネットにアクセスします。悪意の

あるユーザーが大量のARP要求をゲートウェイに送信すると、ゲートウェイがクラッシュしてクライアントからの要求を処理できなくなる可能性があります。この問題を解決するには、ゲートウェイで送信元MACベースのARP攻撃検出を設定します。

図25 ネットワークダイアグラム



手順

#送信元MACベースのARP攻撃検出を有効にし、フィルターとして処理方法を指定します。
<Device> system-view
[Device] arp source-mac filter
#しきい値を30に設定します。
[Device] arp source-mac threshold 30
#ARP攻撃エントリのライフタイムを60秒に設定します。
[Device] arp source-mac aging-time 60
#この検出からMACアドレス0012-3f86-e94cを除外します。
[Device] arp source-mac exclude-mac 0012-3f86-e94c

ARPパケットの送信元MAC整合性チェックの設定

このタスクについて

この機能により、ゲートウェイは、イーサネットヘッダー内の送信元MACアドレスがメッセージ本文内の送信元MACアドレスと異なるARPパケットをフィルタリングできます。この機能により、ゲートウェイは正しいARPエントリを学習できます。

手順

1. システムビューに入ります。

   system-view
2. ARPパケットの送信元MACアドレスの整合性チェックをイネーブルにします。

   arp valid-check enable

   デフォルトでは、ARPパケットの送信元MACアドレスの整合性チェックはディセーブルです。

ARPアクティブ確認応答の設定

このタスクについて

ゲートウェイでARPアクティブ確認応答機能を使用して、ユーザースプーフィングを防止します。

この機能により、デバイスはARPエントリを作成する前にアクティブな確認応答を実行できます。

- デバイスは、デバイスのMACアドレスを要求するARP要求を受信すると、ARP応答を送信します。次に、受信したARP要求内の送信元IPアドレスに対してARP要求を送信し、送信元IPアドレスのARPエントリを作成するかどうかを決定します。
  - デバイスは、プローブ間隔内にARP応答を受信すると、ARPエントリを作成します。
  - デバイスがプローブ間隔内にARP応答を受信しない場合、ARPエントリは作成されません。
- ARP応答を受信すると、デバイスは、それがデバイスが送信した要求に対する応答であるかどうかを調べます。
  - 一致した場合、デバイスはARP応答に送信元IPアドレスのARPエントリを作成します。
  - そうでない場合、デバイスは送信元IPアドレスのARP要求を送信して、送信元IPアドレスのARPエントリを作成するかどうかを決定します。
    - デバイスは、プローブ間隔内にARP応答を受信すると、ARPエントリを作成します。
    - デバイスがプローブ間隔内にARP応答を受信しない場合、ARPエントリは作成されません。

ARPエントリの有効性と信頼性を向上させるために、strictモードでARPアクティブ確認応答をイネーブルにできます。このモードでは、デバイスは、ARP解決をアクティブに開始するIPアドレスに対してだけARPエントリを作成します。

手順

1. システムビューに入ります。

   system-view
2. ARPアクティブ確認応答機能をイネーブルにします。

   `arp active-ack [ strict ] enable`

   デフォルトでは、この機能はディセーブルになっています。

   ARPアクティブ確認応答をstrictモードで有効にするには、ARPブラックホールルーティングがイネーブルになっていることを確認します。

許可ARPの設定

認可ARPについて

認可されたARPエントリは、DHCPサーバー上のDHCPクライアントのアドレスリースまたはDHCPリレーエージェント上のダイナミッククライアントエントリに基づいて生成されます。DHCPサーバーおよびDHCPリレーエージェントの詳細については、『Layer 3 IP Services Configuration Guide』を参照してください。

この機能を使用して、ユーザースプーフィングを防止し、認可されたクライアントだけにネットワークリソースへのアクセスを許可します。

手順

1. システムビューに入ります。

   system-view

2. インターフェイスビューを入力します。

   interface *interface-type interface-number*

   サポートされるインターフェイスタイプには、レイヤ3イーサネットインターフェイス、レイヤ3イーサネットサブインターフェイス、レイヤ3集約インターフェイス、レイヤ3集約サブインターフェイス、およびVLANインターフェイスがあります。

3. インターフェイス上で許可ARPをイネーブルにします。

   arp authorized enable

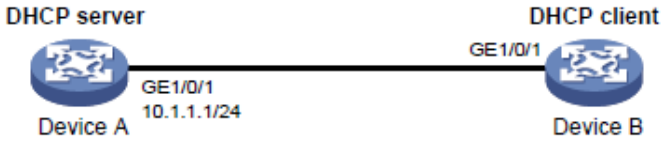
   デフォルトでは、許可ARPはディセーブルになっています。

例:DHCPサーバーでの許可ARPの設定

ネットワーク構成

図26に示すように、デバイスA(DHCPサーバー)のGigabitEthernet 1/0/1に認可ARPを設定して、ユーザーの有効性を確認します。

図26ネットワークダイアグラム



手順

1. デバイスAを構成します。

   #GigabitEthernet 1/0/1のIPアドレスを指定します。

   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.1 24

   [DeviceA-GigabitEthernet1/0/1] quit

   #DHCPを設定します。
   [DeviceA] dhcp enable
   [DeviceA] dhcp server ip-pool 1
   [DeviceA-dhcp-pool-1] network 10.1.1.0 mask 255.255.255.0
   [DeviceA-dhcp-pool-1] qui

   #レイヤ3イーサネットインターフェイスビューを開始します。

   [DeviceA] interface gigabitethernet 1/0/1

   #許可されたARPを有効にします。

   [DeviceA-GigabitEthernet1/0/1] arp authorized enable

   [DeviceA-GigabitEthernet1/0/1] quit

2. デバイスBを構成します。
   <DeviceB> system-view
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] ip address dhcp-alloc

[DeviceB-GigabitEthernet1/0/1] quit

設定の確認

#デバイスAの許可されたARPエントリ情報を表示します。
[DeviceA] display arp all
Type: S-Static D-Dynamic O-Openflow R-Rule M-Multiport I-Invalid
IP address MAC address VLAN/VSI name Interface/Link ID Aging Type

10.1.1.2 0012-3f86-e94c -- GE1/0/1 20 D

この出力は、IPアドレス10.1.1.2がデバイスBに割り当てられていることを示しています。

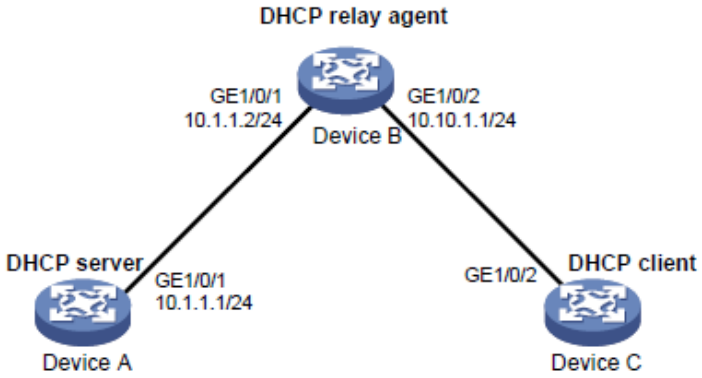
デバイスBは、デバイスAと通信するために、認可されたARPエントリ内のIPアドレスとMACアドレスを使用する必要があります。そうしないと、通信は失敗します。したがって、ユーザーの有効性が保証されます。

例:DHCPリレーエージェントでの許可ARPの設定

ネットワーク構成

図27に示すように、デバイスB(DHCPリレーエージェント)のGigabitEthernet 1/0/2に認可ARPを設定して、ユーザーの有効性を確認します。

図27 ネットワークダイアグラム



手順

1. デバイスAを構成します。
   #GigabitEthernet 1/0/1のIPアドレスを指定します。
   <DeviceA> system-view
   [DeviceA] interface gigabitethernet 1/0/1
   [DeviceA-GigabitEthernet1/0/1] ip address 10.1.1.1 24
   [DeviceA-GigabitEthernet1/0/1] quit
   #DHCPを設定します。
   [DeviceA] dhcp enable
   [DeviceA] dhcp server ip-pool 1
   [DeviceA-dhcp-pool-1] network 10.10.1.0 mask 255.255.255.0
   [DeviceA-dhcp-pool-1] gateway-list 10.10.1.1
   [DeviceA-dhcp-pool-1] quit
   [DeviceA] ip route-static 10.10.1.0 24 10.1.1.2
2. デバイスBの設定:
   #DHCPを有効にします。
   <DeviceB> system-view
   [DeviceB] dhcp enable
   #GigabitEthernet 1/0/1およびGigabitEthernet 1/0/2のIPアドレスを指定します。

[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ip address 10.1.1.2 24
[DeviceB-GigabitEthernet1/0/1] quit
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ip address 10.10.1.1 24 24
#GigabitEthernet 1/0/2でDHCPリレーエージェントをイネーブルにします。
[DeviceB-GigabitEthernet1/0/2] dhcp select relay
#DHCPサーバー10.1.1.1をDHCPサーバーグループ1に追加します。
[DeviceB-GigabitEthernet1/0/2] dhcp relay server-address 10.1.1.1
#許可されたARPを有効にします。
[DeviceB-GigabitEthernet1/0/2] arp authorized enable
[DeviceB-GigabitEthernet1/0/2] quit
#リレーエージェントでリレーエントリの記録をイネーブルにします。
[DeviceB] dhcp relay client-information record

3. デバイスCを構成します。
<DeviceC> system-view
[DeviceC] ip route-static 10.1.1.0 24 10.10.1.1
[DeviceC] interface gigabitethernet 1/0/2
[DeviceC-GigabitEthernet1/0/2] ip address dhcp-alloc
[DeviceC-GigabitEthernet1/0/2] quit

設定の確認

#デバイスBの許可されたARP情報を表示します。
[DeviceB] display arp all
Type: S-Static D-Dynamic O-Openflow R-Rule M-Multiport I-Invalid
IP address MAC address VLAN/VSI name Interface/Link ID Aging Type

10.10.1.2 0012-3f86-e94c -- GE1/0/2 20 D

出力は、デバイスAがデバイスCにIPアドレス10.10.1.2を割り当てたことを示しています。

デバイスCは、デバイスBと通信するために、許可されたARPエントリ内のIPアドレスとMACアドレスを使用する必要があります。そうしないと、通信は失敗します。これにより、ユーザーの有効性が保証されます。

ARP攻撃検出の設定

ARP攻撃の検出について

ARP攻撃検出を使用すると、アクセスデバイスは、許可されていないクライアントからのARPパケットをブロックして、ユーザースプーフィング攻撃およびゲートウェイスプーフィング攻撃を防止できます。

ARP攻撃検出には、次の機能があります。

- ユーザー妥当性検査。
- ARPパケットの有効性チェック。
- ARP制限付き転送。
- ARP攻撃検出ロギング。

ARPパケットの有効性チェックとユーザーの有効性チェックの両方がイネーブルになっている場合、前者が最初に適用され、次に後者が適用されます。

ARP攻撃検出とARPスヌーピングを同時に設定しないでください。同時に設定すると、ARPスヌーピングエントリを生成できません。

ARP攻撃検出とのハードウェア互換性

ハードウェア	備考
• 次のルーターの固定レイヤ2イーサネットポート: o MSR2600-6-X1、MSR2600-15-X1、MSR2600-15-X1-T o MSR2600-10-X1 o MSR3600-28, MSR3600-51 o MSR3600-28-X1、MSR3600-28-X1-DP、MSR3600-51-X1、MSR3600-51-X1-DP o MSR3600-28-G-DP, MSR3600-51-G-DP o MSR 2600-6-WiNet o MSR 3600-28- WiNet o MSR2630-XS o MSR3600-28-XS o MSR2600-6-X1-GL • ルーターにインストールされている次のレイヤ2インターフェイスモジュール。 o HMIM-24GSW o HMIM-24GSW-PoE o HMIM-8GSW o HMIM-8GSWF o SIC-4GSW	レイヤ2インターフェイスモジュールに対するルーターのサポートについては、『H3C MSR Router Series Comware 7 Interface Module Guide』を参照してください。

ユーザー妥当性検査の構成

このタスクについて

ユーザー有効性チェックでは、ARP信頼インターフェイスで受信されたARPパケットはチェックされません。この機能は、ARP信頼できないインターフェイスで受信されたARPパケットの送信元IPおよび送信元MACを、次の順序で一致基準と比較します。

1. ユーザー妥当性検査ルール。
   - 一致が見つかった場合、デバイスはルールに従ってARPパケットを処理します。
   - 一致が見つからない場合、またはユーザー有効性チェック規則が設定されていない場合は、ステップ2に進みます。
2. スタティックIPSGバインディング、802.1Xセキュリティエントリ、およびDHCPスヌーピングエントリ。
   - 一致が検出された場合、デバイスはARPパケットが有効であると判断します。次に、デバイスはターゲットIPアドレスを含むエントリを検索してパケットを転送します。
     - 一致が見つかり、受信インターフェイスが、一致する送信元IPアドレスを持つエントリのインターフェイスと異なる場合、デバイスはレイヤ3転送を実行します。
     - 一致が検出されたが、受信インターフェイスが、一致する送信元IPアドレスを持つエントリのインターフェイスと同じである場合、デバイスはレイヤ2転送を実行します。
     - 一致するものが見つからない場合、デバイスはレイヤ2転送を実行します。
   - 一致するものが見つからない場合、デバイスはARPパケットを廃棄します。

スタティックIPソースガードバインディングは、ip source bindingコマンドを使用して作成します。詳細については、「IPソースガードの設定」を参照してください。

DHCPスヌーピングエントリは、DHCPスヌーピングによって自動的に生成されます。詳細については、を参照してください。
『Layer 3-IP Services Configuration Guide』を参照してください。

802.1 Xセキュリティエントリは、802.1XクライアントのIPからMACへのマッピングを記録します。クライアントが802.1X認証を通過し、ARP攻撃検出が有効なデバイスにそのIPアドレスをアップロードす

ると、デバイスは自動的に802.1Xセキュリティエントリを生成します。デバイスにIPアドレスをアップロードするには、802.1Xクライアントを有効にする必要があります。詳細は、「802.1Xの構成」を参照してください。

制限事項およびガイドライン

ユーザー有効性チェックを設定する場合は、次の項目の1つ以上が設定されていることを確認します。

- ユーザー妥当性検査ルール。
- スタティックIPソースガードバインディング。
- DHCPスヌーピング。
- 802.1X

いずれの項目も設定されていない場合、ARP untrustedインターフェイス上の着信ARPパケットは正常に転送できません。

IPソースガードバインディングに対してARP攻撃検出をイネーブルにするIPアドレス、MACアドレス、およびVLANを指定します。イネーブルにしない場合、IPソースガードバインディングに一致するARPパケットはありません。

手順

1. システムビューに入ります。

   system-view
2. (任意)ユーザー有効性チェック規則を設定します。
   arp detection rule *rule-id* { **deny** | **permit** } **ip** { *ip-address* [*mask*] | **any** } **mac** { *mac-address* [*mask*] | **any** } [**vlan** *vlan-id*]

   デフォルトでは、ユーザー有効性チェック規則は設定されていません。
3. VLANビューを開始します。

   vlan *vlan-id*
4. ARP攻撃検出をイネーブルにします。

   arp detection enable

   デフォルトでは、ARP攻撃検出はディセーブルになっています。デバイスは、ユーザーの有効性チェックを実行しません。
5. (任意)ARPユーザー有効性チェックを必要としないインターフェイスを、信頼できるインターフェイスとして設定します。
   a. システムビューに戻ります。

      quit
   b. インターフェイスビューを入力します。

      interface *interface-type interface-number*

      サポートされているインターフェイスタイプには、レイヤ2イーサネットインターフェイスおよびレイヤ2集約インターフェイスがあります。
   c. インターフェイスを、ARP攻撃検出から除外される信頼できるインターフェイスとして設定します。

      arp detection trust

      デフォルトでは、インターフェイスは信頼できません。

ARPパケットの有効性チェックの設定

このタスクについて

ARPパケットの有効性チェックでは、ARP信頼インターフェイスで受信されたARPパケットはチェックされません。信頼できないインターフェイスで受信されたARPパケットをチェックするには、次のチェック対象オブジェクトを指定できます。

- **src-mac**: メッセージ本文の送信元MACアドレスが、イーサネットヘッダーの送信元MACアドレスと同一であるかどうかをチェックします。同一である場合、パケットは転送されます。同一でない場合、パケットは廃棄されます。
- **dst-mac**: ARP応答のターゲットMACアドレスをチェックします。ターゲットMACアドレスがすべて0、すべて1、またはイーサネットヘッダーの宛先MACアドレスと一致しない場合、パケットは無効と見なされ、廃棄されます。
- **ip**: ARP応答の送信元と宛先のIPアドレス、およびARP要求の送信元IPアドレスをチェックします。すべてが1つのIPアドレスまたはマルチキャストIPアドレスは無効と見なされ、対応するパケットは廃棄されます。

前提条件

ARPパケットの有効性チェックを設定する前に、まずユーザーの有効性チェックを設定する必要があります。ユーザーの有効性チェックの設定の詳細については、「ユーザーの有効性チェックの設定」を参照してください。

手順

1. システムビューに入ります。

   system-view
2. VLANビューを開始します。

   vlan *vlan-id*
3. ARP攻撃検出をイネーブルにします。

   arp detection enable

   デフォルトでは、ARP攻撃検出はディセーブルになっています。デバイスはARPパケットの有効性チェックを実行しません。
4. ARPパケットの有効性チェックをイネーブルにします。
   a. システムビューに戻ります。

      quit
   b. ARPパケットの有効性チェックを有効にし、チェックするオブジェクトを指定します。

      arp detection validate { **dst-mac** | **ip** | **src-mac** } *

      デフォルトでは、ARPパケットの有効性チェックは無効になっています。
5. (任意)ARPパケットの有効性チェックを必要としないインターフェイスを、信頼できるインターフェイスとして設定します。
   a. インターフェイスビューを入力します。

      interface *interface-type interface-number*

      サポートされているインターフェイスタイプには、レイヤ2イーサネットインターフェイスおよびレイヤ2集約インターフェイスがあります。
   b. インターフェイスを、ARP攻撃検出から除外される信頼できるインターフェイスとして設定します。

      arp detection trust

      デフォルトでは、インターフェイスは信頼できません。

ARP制限付き転送の設定

このタスクについて

ARP制限付き転送は、ARP信頼インターフェイスで受信されたARPパケットには影響せず、ARPパケットを正しく転送します。この機能は、信頼できないインターフェイスで受信され、ユーザー有効性チェックに合格したARPパケットの転送を次のように制御します。

- パケットがARP要求の場合は、信頼できるインターフェイスを介して転送されます。
- パケットがARP応答の場合は、宛先MACアドレスに従って転送されます。MACアドレステーブルで一致が見つからない場合は、信頼できるインターフェイスを介して転送されます。

制限事項およびガイドライン

ARP制限付き転送は、マルチポート宛先MACアドレスを使用するARPパケットには適用されません。

前提条件

ARP制限付き転送を構成する前に、ユーザー妥当性チェックを構成します。ユーザー妥当性チェックの構成の詳細は、「ユーザー妥当性チェックの構成」を参照してください。

手順

1. システムビューに入ります。
   system-view
2. VLANビューを開始します。
   vlan *vlan-id*
3. ARP制限転送をイネーブルにします。
   arp restricted-forwarding enable
   デフォルトでは、ARP制限転送はディセーブルになっています。

ARP攻撃検出用の表示およびメンテナンスコマンド

任意のビューでdisplayコマンドを実行し、ユーザービューでコマンドをリセットします。

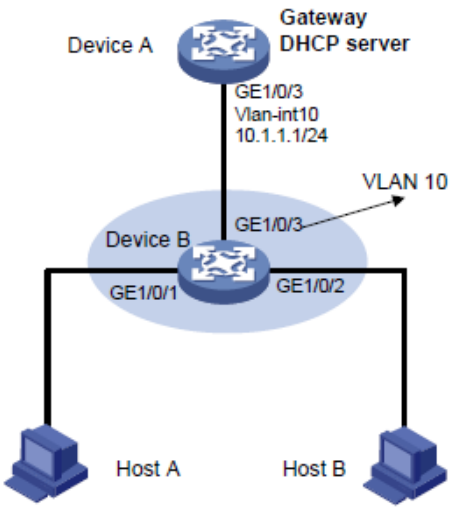
タスク	コマンド
ARP攻撃検出がイネーブルになっているVLANを表示します。	display arp detection
ARP攻撃検出によってドロップされたパケットの統計情報を表示します。	display arp detection statistics [interface *interface-type interface-number*]
ARP攻撃検出によってドロップされたパケットの統計情報をクリアします。	reset arp detection statistics [interface *interface-type interface-number*]

例:ユーザー妥当性検査の構成

ネットワーク構成

図28に示すように、接続されたホストの802.1Xセキュリティエントリに基づいてユーザーの有効性チェックを実行するようにデバイスBを設定します。

図28 ネットワークダイアグラム



手順

1. デバイスBのすべてのインターフェイスをVLAN 10に追加し、デバイスAのVLANインターフェイス10のIPアドレスを指定します(詳細は省略)。
2. デバイスAでDHCPサーバーを設定し、DHCPアドレスプール0を設定します。
   <DeviceA> system-view
   [DeviceA] dhcp enable
   [DeviceA] dhcp server ip-pool 0
   [DeviceA-dhcp-pool-0] network 10.1.1.0 mask 255.255.255.0
3. ホストAとホストBを802.1Xクライアントとして設定し、ARP攻撃検出用のIPアドレスをアップロードするように設定します(詳細は省略)。
4. デバイスBを設定します。

   #802.1Xを有効にします。

   <DeviceB> system-view
   [DeviceB] dot1x
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] dot1x
   [DeviceB-GigabitEthernet1/0/1] quit
   [DeviceB] interface gigabitethernet 1/0/2
   [DeviceB-GigabitEthernet1/0/2] dot1x
   [DeviceB-GigabitEthernet1/0/2] quit
   #ローカルユーザーテストを追加します。
   [DeviceB] local-user test class network
   [DeviceB-luser-network-test] service-type lan-access
   [DeviceB-luser-network-test] password simple test
   [DeviceB-luser-network-test] quit
   #VLAN 10のARP攻撃検出をイネーブルにして、802.1Xエントリに基づいてユーザーの有効性をチェックします。
   [DeviceB] vlan 10
   [DeviceB-vlan10] arp detection enable
   #アップストリームインターフェイスをARP信頼インターフェイスとして設定します。デフォルトでは、インターフェイスは信頼できないインターフェイスです。
   [DeviceB-vlan10] interface gigabitethernet 1/0/3
   [DeviceB-GigabitEthernet1/0/3] arp detection trust
   [DeviceB-GigabitEthernet1/0/3] quit

設定の確認

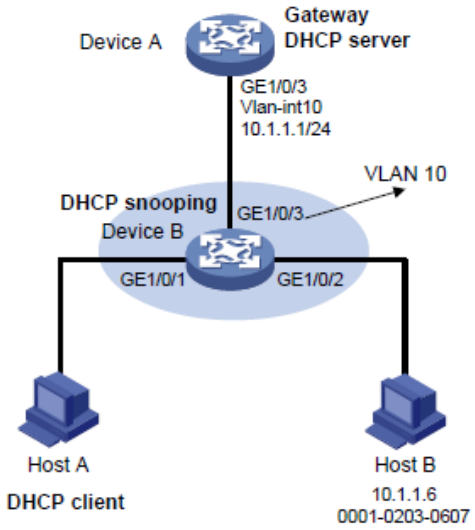
#インターフェイスGigabitEthernet 1/0/1およびGigabitEthernet 1/0/2で受信されたARPパケットが、802.1Xエントリに対してチェックされることを確認します。

例:ユーザー有効性チェックおよびARPパケット有効性チェックの設定

ネットワーク構成

図29に示すように、接続されたホストのスタティックIPソースガードバインディングとDHCPスヌーピングエントリに基づいて、ARPパケットの有効性チェックとユーザーの有効性チェックを実行するようにデバイスBを設定します。

図29 ネットワークダイアグラム



手順

1. デバイスBのすべてのインターフェイスをVLAN 10に追加し、デバイスAのVLANインターフェイス10のIPアドレスを指定します(詳細は省略)。
2. デバイスAでDHCPサーバーを設定し、DHCPアドレスプール0を設定します。
   <DeviceA> system-view
   [DeviceA] dhcp enable
   [DeviceA] dhcp server ip-pool 0
   [DeviceA-dhcp-pool-0] network 10.1.1.0 mask 255.255.255.0
3. ホストA(DHCPクライアント)とホストBを構成します(詳細は省略)。
4. デバイスBを構成します。
   #DHCPスヌーピングを有効にします。
   <DeviceB> system-view
   [DeviceB] dhcp snooping enable
   [DeviceB] interface gigabitethernet 1/0/3
   [DeviceB-GigabitEthernet1/0/3] dhcp snooping trust
   [DeviceB-GigabitEthernet1/0/3] quit
   #GigabitEthernet 1/0/1のDHCPスヌーピングエントリで、クライアント情報の記録をイネーブルにします。
   [DeviceB] interface gigabitethernet 1/0/1
   [DeviceB-GigabitEthernet1/0/1] dhcp snooping binding record
   [DeviceB-GigabitEthernet1/0/1] quit
   #VLAN 10のARP攻撃検出を有効にします。

[DeviceB] vlan 10
[DeviceB-vlan10] arp detection enable
#アップストリームインターフェイスを信頼できるインターフェイスとして設定します。デフォルトでは、インターフェイスは信頼できないインターフェイスです。
[DeviceB-vlan10] interface gigabitethernet 1/0/3
[DeviceB-GigabitEthernet1/0/3] arp detection trust
[DeviceB-GigabitEthernet1/0/3] quit
#ユーザーの有効性チェックのために、インターフェイスGigabitEthernet 1/0/2にスタティックIPソースガードバインディングエントリを設定します。
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] ip source binding ip-address 10.1.1.6 mac-address 0001-0203-0607 vlan 10
[DeviceB-GigabitEthernet1/0/2] quit
#ARPパケットのMACアドレスとIPアドレスをチェックすることで、ARPパケットの有効性チェックを有効にします。
[DeviceB] arp detection validate dst-mac ip src-mac

設定の確認

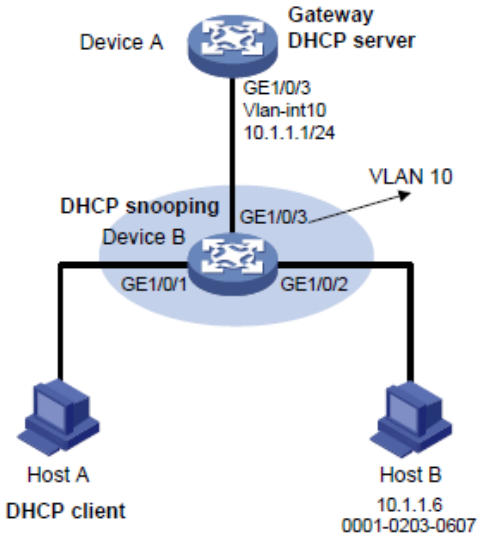
#デバイスBが最初にGigabitEthernet 1/0/1およびGigabitEthernet 1/0/2で受信したARPパケットの有効性をチェックすることを確認します。ARPパケットが有効であることが確認された場合、デバイスBはスタティックIPソースガードバインディングを使用してユーザーの有効性チェックを実行し、最後にDHCPスヌーピングエントリを使用します。

例:ARP制限付きフォワーディングの設定

ネットワーク構成

図30に示すように、ARP攻撃検出が設定されているデバイスBにARP制限付き転送を設定します。デバイスBに設定されたポート分離は、ブロードキャストARP要求に対して有効になります。

図30 ネットワークダイアグラム



手順

1. VLAN 10を設定し、VLAN 10にインターフェイスを追加し、デバイスAのVLANインターフェイス10のIPアドレスを指定します(詳細は省略)。
2. デバイスAでDHCPサーバーを設定し、DHCPアドレスプール0を設定します。
<DeviceA> system-view
[DeviceA] dhcp enable
[DeviceA] dhcp server ip-pool 0

[DeviceA-dhcp-pool-0] network 10.1.1.0 mask 255.255.255.0

3. ホストA(DHCPクライアント)とホストBを構成します(詳細は省略)。

4. デバイスBを構成します。

   #DHCPスヌーピングをイネーブルにし、GigabitEthernet 1/0/3をDHCP信頼インターフェイスとして設定します。

   <DeviceB> system-view

   [DeviceB] dhcp snooping enable

   [DeviceB] interface gigabitethernet 1/0/3

   [DeviceB-GigabitEthernet1/0/3] dhcp snooping trust
   [DeviceB-GigabitEthernet1/0/3] quit

   #ユーザーの有効性チェックのためにARP攻撃検出を有効にします。
   [DeviceB] vlan 10

   [DeviceB-vlan10] arp detection enable

   #GigabitEthernet 1/0/3をARP信頼インターフェイスとして設定します。

   [DeviceB-vlan10] interface gigabitethernet 1/0/3
   [DeviceB-GigabitEthernet1/0/3] arp detection trust

   [DeviceB-GigabitEthernet1/0/3] quit

   #インターフェイスGigabitEthernet 1/0/2にスタティックIPソースガードエントリを設定します。
   [DeviceB] interface gigabitethernet 1/0/2
   [DeviceB-GigabitEthernet1/0/2] ip source binding ip-address 10.1.1.6 mac-address 0001-0203-0607 vlan 10

   [DeviceB-GigabitEthernet1/0/2] quit

   #ARPパケットのMACアドレスとIPアドレスをチェックすることで、ARPパケットの有効性チェックを有効にします。

   [DeviceB] arp detection validate dst-mac ip src-mac

   #隔離ポートを設定します。

   [DeviceB] port-isolate group 1

   [DeviceB] interface gigabitethernet 1/0/1

   [DeviceB-GigabitEthernet1/0/1] port-isolate enable group 1

   [DeviceB-GigabitEthernet1/0/1] quit

   [DeviceB] interface gigabitethernet 1/0/2

   [DeviceB-GigabitEthernet1/0/2] port-isolate enable group 1

   [DeviceB-GigabitEthernet1/0/2] quit

   設定が完了すると、デバイスBは最初にGigabitEthernet 1/0/1およびGigabitEthernet 1/0/2で受信したARPパケットの有効性を確認します。ARPパケットが有効であることが確認された場合、デバイスBはスタティックIPソースガードバインディングを使用してユーザーの有効性チェックを実行し、最後にDHCPスヌーピングエントリを使用します。ただし、ホストAから送信されたARPブロードキャスト要求は、デバイスBのチェックを通過してホストBに到達できます。ポートの分離は失敗します。

   #ARP制限付き転送を有効にします。

   [DeviceB] vlan 10

   [DeviceB-vlan10] arp restricted-forwarding enable

   [DeviceB-vlan10] quit

設定の確認

#デバイスBが、信頼できるインターフェイスGigabitEthernet 1/0/3を介して、ホストAからデバイ

スAにARPブロードキャスト要求を転送することを確認します。ホストBは、このようなパケットを受信できません。ポート分離は正常に動作します。

ARPスキャンおよび固定ARPの設定

このタスクについて

ARPスキャンは、通常、小規模で安定したネットワークで固定ARP機能とともに使用されます。

ARPスキャンは、アドレス範囲内のデバイスのARPエントリを自動的に作成します。デバイスは、次の手順でARPスキャンを実行します。

1. アドレス範囲内の各IPアドレスに対してARP要求を送信します。
2. 受信したARP応答を通じてMACアドレスを取得します。
3. ダイナミックARPエントリを作成します。

固定ARPは、既存のダイナミックARPエントリ(ARPスキャンによって生成されたものを含む)をスタティックARPエントリに変換します。これらのスタティックARPエントリは、手動で設定されたARPエントリと同じ属性を持ちます。この機能は、攻撃者によるARPエントリの変更を防止します。

制限事項およびガイドライン

既存のARPエントリ内のIPアドレスはスキャンされません。

スタティックARPエントリの合計数の制限により、一部のダイナミックARPエントリが変換に失敗する場合があります。

arp fixupコマンドは1回限りの操作です。このコマンドを再度使用して、後で学習したダイナミックARPエントリをスタティックに変換できます。

ダイナミックARPエントリから変換されたスタティックARPエントリを削除するには、undo arp ip-addressvpn-instance-nameコマンドを使用します。また、reset arp allコマンドを使用してすべてのARPエントリを削除したり、reset arp staticコマンドを使用してすべてのスタティックARPエントリを削除することもできます。

手順

1. システムビューに入ります。

   system-view
2. インターフェイスビューを入力します。

   interface *interface-type interface-number*
3. ARPスキャンをトリガーします。

   arp scan [*start-ip-address* **to** *end-ip-address*]

△注意:

ARPスキャンには時間がかかります。進行中のスキャンを停止するには、Ctrl+Cを押します。ダイナミックARPエントリは、スキャンが終了する前に受信されたARP応答に基づいて作成されます。

4. システムビューに戻ります。

   quit
5. 既存のダイナミックARPエントリをスタティックARPエントリに変換します。

   arp fixup

ARPゲートウェイ保護の設定

ARPゲートウェイ保護について

ゲートウェイスプーフィング攻撃を防止するには、ゲートウェイに接続されていないインターフェイスでこの機能を設定します。

このようなインターフェイスは、ARPパケットを受信すると、パケット内の送信元IPアドレスが保護されたゲートウェイのIPアドレスと一致しているかどうかをチェックします。一致している場合は、パケットを破棄します。一致していない場合は、パケットを正しく処理します。

制限事項およびガイドライン

ARPゲートウェイ保護は、インターフェイス上で最大8つのゲートウェイに対してイネーブルにできます。

インターフェイス上でarp filter sourceコマンドとarp filter bindingコマンドの両方を設定しないでください。

ARPゲートウェイ保護がARP攻撃検出、ARPスヌーピング、およびARPファーストリプライと連動する場合、ARPゲートウェイ保護が最初に適用されます。

手順

1. システムビューに入ります。
   system-view
2. インターフェイスビューを入力します。
   interface *interface-type interface-number*
3. 指定されたゲートウェイのARPゲートウェイ保護をイネーブルにします。
   arp filter source *ip-address*
   デフォルトでは、ARPゲートウェイ保護はディセーブルになっています。

ARPフィルタリングの設定

ARPフィルタリング

ARPフィルタリング機能は、ゲートウェイスプーフィング攻撃とユーザースプーフィング攻撃を防止できます。

この機能を有効にしたインターフェイスは、受信したARPパケットの送信元IPアドレスとMACアドレスを、許可されたエントリと照合します。一致するものが見つかった場合、パケットは正しく処理されます。一致しない場合、パケットは破棄されます。

制限事項およびガイドライン

インターフェイスには、最大8つの許可エントリを設定できます。

インターフェイス上でarp filter sourceコマンドとarp filter bindingコマンドの両方を設定しないでください。

ARPフィルタリングがARP攻撃検出、ARPスヌーピング、およびARP高速応答とともに動作する場合、ARPフィルタリングが最初に適用されます。

手順

1. システムビューに入ります。

   system-view
2. インターフェイスビューを入力します。

   interface *interface-type interface-number*
3. ARPフィルタリングを有効にし、許可されたエントリを設定します。

   arp filter binding *ip-address mac-address*

   デフォルトでは、ARPフィルタリングは無効になっています。

ND攻撃防御の設定

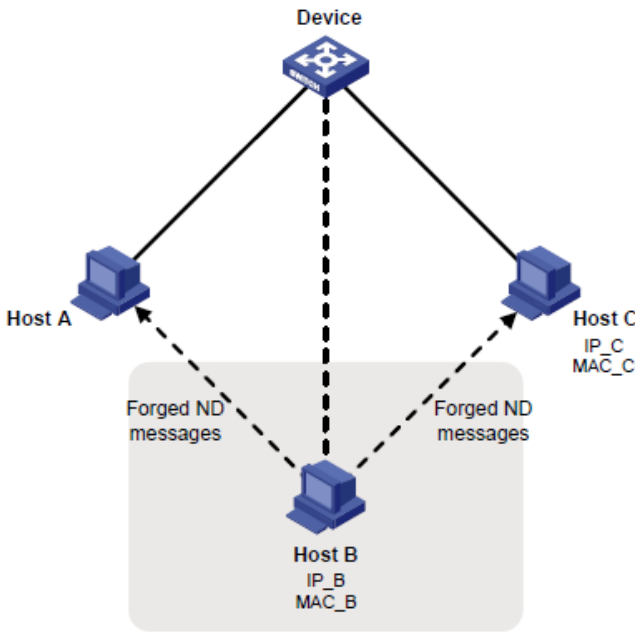
ND攻撃防御について

IPv6近隣探索(ND)攻撃防御では、偽造されたNDメッセージを識別して、ND攻撃を防ぐことができる。

IPv6 NDプロトコルはセキュリティメカニズムを提供せず、ネットワーク攻撃に対して脆弱です。図31に示すように、攻撃者は次の偽造ICMPv6メッセージを送信してND攻撃を実行できます。

- 被害ホストのIPv6アドレスを持つ偽造されたNS/NA/RSメッセージ。ゲートウェイと他のホストは、不正なアドレス情報で被害のNDエントリを更新します。その結果、被害を目的としたすべてのパケットが攻撃端末に送信されます。
- ビクティムゲートウェイのIPv6アドレスを使用した偽造RAメッセージ。その結果、ビクティムゲートウェイに接続されているすべてのホストは、不正なIPv6設定パラメータとNDエントリを保持します。

図31 NDアタックダイヤグラム



送信元MACベースND攻撃検出の設定

送信元MACベースのND攻撃検出について

送信元MACベースのND攻撃検出では、CPUに配信されたNDメッセージの数が送信元MACごとにチェックされます。同じMACアドレスからのメッセージの数が5秒以内にしきい値を超えると、デバイスはそのMACアドレスのND攻撃エントリを生成します。このエントリに一致するNDメッセージの処理は、検出モードによって異なります。NDロギングがイネーブルになっている場合(ipv6 nd check log enableコマンドを使用)、送信元MACベースのND攻撃検出では、メッセージが次のように処理されます。

- **Filter mode**: MACアドレスから送信された後続のNDメッセージをフィルタリングし、ログメッセージを生成します。
- **Monitor mode**: ログメッセージだけを生成します。

デバイスは、エントリのエージングタイム(300秒に固定)としきい値を使用して値を計算します。計算された値=(しきい値/5)×300

デバイスは、エントリに対してドロップされたパケットの数を監視します。エントリのエージングタイムに達すると、その数を計算値と比較し、それに応じたアクションを実行します。

- ドロップされたパケットの数が計算値以上の場合、デバイスはエントリのエージングタイムをリセットします。
- ドロップされたパケットの数が計算された値より少ない場合、システムはエントリを削除し、エントリ内のMACアドレスを共通MACアドレスとしてマークします。

制限事項およびガイドライン

検出モードをモニタからフィルターに変更すると、フィルターモードはすぐに有効になります。

検出モードをfilterからmonitorに変更すると、デバイスは既存の攻撃エントリに一致するメッセージのフィルタリングを続行します。

手順

1. システムビューに入ります。
   system-view
2. 送信元MACベースのND攻撃検出をイネーブルにし、検出モードを設定します。
   ipv6 nd source-mac **{** filter **|** monitor **}**
   デフォルトでは、送信元MACベースのND攻撃検出はディセーブルです。
3. 送信元MACベースのND攻撃検出のしきい値を設定します。
   ipv6 nd source-mac threshold *threshold-value*
   デフォルト設定は30です。
4. NDロギング機能をイネーブルにします。
   ipv6 nd check log enable

   デフォルトでは、NDロギング機能はディセーブルになっています。

送信元MACベースのND攻撃検出用の表示およびメンテナンスコマンド

任意のビューでdisplayコマンドを実行し、ユーザービューでコマンドをリセットします。

タスク	コマンド
送信元MACベースND攻撃検出の設定を表示します。	display ipv6 nd source-mac configuration
送信元MACベースND攻撃検出エントリを表示します。	スタンドアロンモードの場合: **display ipv6 nd source-mac interface** *interface-type interface-number* **[** verbose **]** **display ipv6 nd source-mac** { **mac** ***mac-address*** **|** vlan ***vlan-id*** **} [** verbose **]** **display ipv6 nd source-mac** [**count** | verbose **]**

タスク	コマンド
	IRFモードの場合: **display ipv6 nd source-mac interface** *interface-type interface-number* [**slot** *slot-number*] [**verbose**] **display ipv6 nd source-mac**{ **mac** *mac-address* | **vlan** *vlan-id* } **slot** *slot-number* [**verbose**] **display ipv6 nd source-mac slot** ***slot-number*** **[** count **|** verbose **]**
送信元MACベースND攻撃検出エントリを削除します。	スタンドアロンモードの場合: **reset ipv6 nd source-mac** [**interface** *interface-type interface-number* | **mac** ***mac-address*** **|** vlan ***vlan-id*** **]** IRFモードの場合: **reset ipv6 nd source-mac** [**interface** *interface-type interface-number* | **mac** *mac-address* | **vlan** *vlan-id*] [**slot** ***slot-number*** **]**

インターフェイスベースのND攻撃抑制の設定

インターフェイスベースのND攻撃抑制について

この機能は、NDスプーフィング攻撃を防止するために、各レイヤ3インターフェイス上のND要求をレ

ート制限します。各レイヤ3インターフェイスが5秒以内に受信したND要求の数を監視します。インターフェイス上の数がしきい値を超えると、デバイスはそのインターフェイスのND攻撃抑制エントリを作成します。抑制期間中(300秒に固定)、デバイスはこのインターフェイスで受信したNDメッセージをドロップします。

抑え込みの時間が期限切れになると、システムは抑え込みの時間内のインターフェイスでドロップされたNDメッセージの数を調べます。

- 数値が計算値以上の場合、デバイスはエントリの抑え込みの時間をリセットし、インターフェイス上でND抑制を継続します。

  計算値=(閾値/5)×300
- 数値が計算値よりも小さい場合、デバイスは抑制エントリを削除します。

制限事項およびガイドライン

ベストプラクティスとして、ゲートウェイでこの機能をイネーブルにします。

手順

1. システムビューに入ります。

   system-view
2. インターフェイスベースのND攻撃抑制をイネーブルにします。

   ipv6 nd attack-suppression enable per-interface

   デフォルトでは、インターフェイスベースのND攻撃抑制はディセーブルになっています。
3. ND攻撃抑制をトリガーするしきい値を設定します。

   ipv6 nd attack-suppression threshold *threshold-value*

   デフォルトでは、ND攻撃抑制をトリガーするしきい値は1000です。

インターフェイスベースのND攻撃抑制の表示コマンドおよびメンテナンスコマンド

任意のビューで表示コマンドを実行します。

タスク	コマンド
の設定を表示します。 インターフェイスベースのND攻撃抑制。	**display ipv6 nd attack-suppression** configuration
インターフェイスベースのND攻撃抑制エントリを表示します。	スタンドアロンモードの場合: **display ipv6 nd attack-suppression** per-interface**[** count **|** verbose **]** IRFモードの場合: **display ipv6 nd attack-suppression per-interface slot** *slot-number* [**count**	verbose **]**
インターフェイス上のインターフェイスベースのND攻撃抑制エントリを表示します。	**display ipv6 nd attack-suppression per-interface interface** *interface-type* ***interface-number*** **[** verbose **]**

インターフェイスベースのND攻撃抑制エントリを削除します。	スタンドアロンモードの場合: **reset ipv6 nd attack-suppression per-interface** [**interface** *interface-type* ***interface-number*** **]** IRFモードの場合: **per-interface** [**interface** *interface-type* ***interface-number*** **] [** slot ***slot-number*** **]**
インターフェイスベースのND攻撃抑制によってドロップされたNDメッセージの統計情報をクリアします。	スタンドアロンモードの場合: **reset ipv6 nd attack-suppression per-interface statistics** [**interface** *interface-type interface-number*] IRFモードの場合: **reset ipv6 nd attack-suppression per-interface statistics** [**interface** *interface-type interface-number*] [**slot** ***slot-number*** **]**

NDメッセージの送信元MAC整合性チェックのイネーブル化

このタスクについて

通常、送信元MAC整合性チェック機能は、ND攻撃を防止するためにゲートウェイで設定されます。

この機能は、各着信NDメッセージの一貫性について、送信元MACアドレスと送信元リンクレイヤアドレスをチェックします。

- 送信元MACアドレスと送信元リンクレイヤアドレスが同じでない場合、デバイスはパケットをドロップします。
- アドレスが同じ場合、デバイスはNDエントリの学習を継続します。

NDロギング機能は、送信元MAC不一致イベントを記録し、ログメッセージをインフォメーションセンターに送信します。インフォメーションセンターは、さまざまな送信元モジュールからさまざまな宛先にログメッセージを出力できます。インフォメーションセンターの詳細については、『Network Management and Monitoring Configuration Guide』を参照してください。

手順

1. システムビューに入ります。

   system-view
2. NDメッセージの送信元MAC整合性チェックをイネーブルにします。

   ipv6 nd mac-check enable

   デフォルトでは、NDメッセージの送信元MAC整合性チェックはディセーブルになっています。
3. (任意)NDロギング機能をイネーブルにします。

   ipv6 nd check log enable

   デフォルトでは、NDロギング機能はディセーブルになっています。

   過剰なNDログを回避するには、NDロギング機能をディセーブルにすることを推奨します。

uRPFの設定

uRPFについて

ユニキャストReverse Path Forwarding(uRPF)は、DoS攻撃やDDoS攻撃などの送信元アドレススプーフィング攻撃からネットワークを保護します。

uRPFアプリケーションのシナリオ

攻撃者は、許可されたユーザーまたは管理者の名前で、IPv4ベースの認証を使用するシステムにアクセスするために、偽造された送信元アドレスを持つパケットを送信します。攻撃者または他のホストが応答パケットを受信できない場合でも、攻撃は攻撃されたターゲットを混乱させます。

図32 送信元アドレススプーフィング攻撃

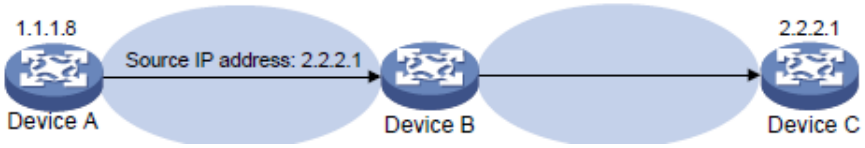


図32に示すように、デバイスAの攻撃者は、偽造された送信元IPアドレス2.2.2.1を含むサーバー(デバイスB)要求を高いレートで送信します。デバイスBは、IPアドレス2.2.2.1(デバイスC)に応答パケットを送信します。その結果、デバイスBとデバイスCの両方が攻撃されます。管理者が誤ってデバイスCを切断すると、ネットワークサービスが中断されます。

攻撃者は、異なる偽造された送信元アドレスを持つパケットを送信したり、複数のサーバーを同時に攻撃して接続をブロックしたり、ネットワークを破壊したりすることもできます。

uRPFは、これらの送信元アドレススプーフィング攻撃を防ぐことができます。uRPFは、パケットを受信するインターフェイスが、パケットの送信元アドレスと一致するFIBエントリの出力インターフェイスであるかどうかを確認します。一致しない場合、uRPFはスプーフィング攻撃と見なし、パケットを廃棄します。

uRPFチェックモード

uRPFは、strictモードとlooseモードをサポートします。

Strict uRPFチェック

厳密なuRPFチェックに合格するには、パケットの送信元アドレスと受信インターフェイスが、FIBエントリの宛先アドレスと出力インターフェイスに一致する必要があります。一部のシナリオ(非対称ルーティングなど)では、厳密なuRPFによって有効なパケットが廃棄される場合があります。

Strict uRPFは、多くの場合、PEとCEの間に配置されます。

Loose uRPFチェック

Loose uRPFチェックを通過するには、パケットの送信元アドレスがFIBエントリの宛先アドレスと一致する必要があります。Loose uRPFでは、有効なパケットの廃棄を回避できますが、攻撃パケットが廃棄される可能性があります。

Loose uRPFは、特に非対称ルーティングでは、ISP間で展開されることがよくあります。

uRPF拡張関数

リンク層チェック

厳密なuRPFチェックでは、さらにパケットのリンク層チェックを実行できます。一致するFIBエントリ内のネクストホップアドレスを使用して、一致するエントリのARPテーブルを検索します。パケットの送信元MACアドレスが一致するARPエントリ内のMACアドレスと一致する場合、パケットは厳密なuRPFチェックに合格します。リンク層チェックは、レイヤ3イーサネットインターフェイスが多数のPCを接続するISPデバイスに適用できます。

Loose uRPFは、リンク層チェックをサポートしません。

uRPFチェックでのデフォルトルートの使用

デフォルトルートが存在する場合、特定のFIBエントリに一致しないすべてのパケットは、uRPFチェック中にデフォルトルートに一致するため、通過が許可されます。この状況を回避するには、uRPFがデフォルトルートを使用してこのようなパケットを廃棄することをディセーブルにします。デフォルトルートの使用を許可する場合(allow-default-routeを使用して設定)、uRPFはデフォルトルートに一致するパケットだけを許可します。

デフォルトでは、uRPFはデフォルトルートにしか一致しないパケットを廃棄します。

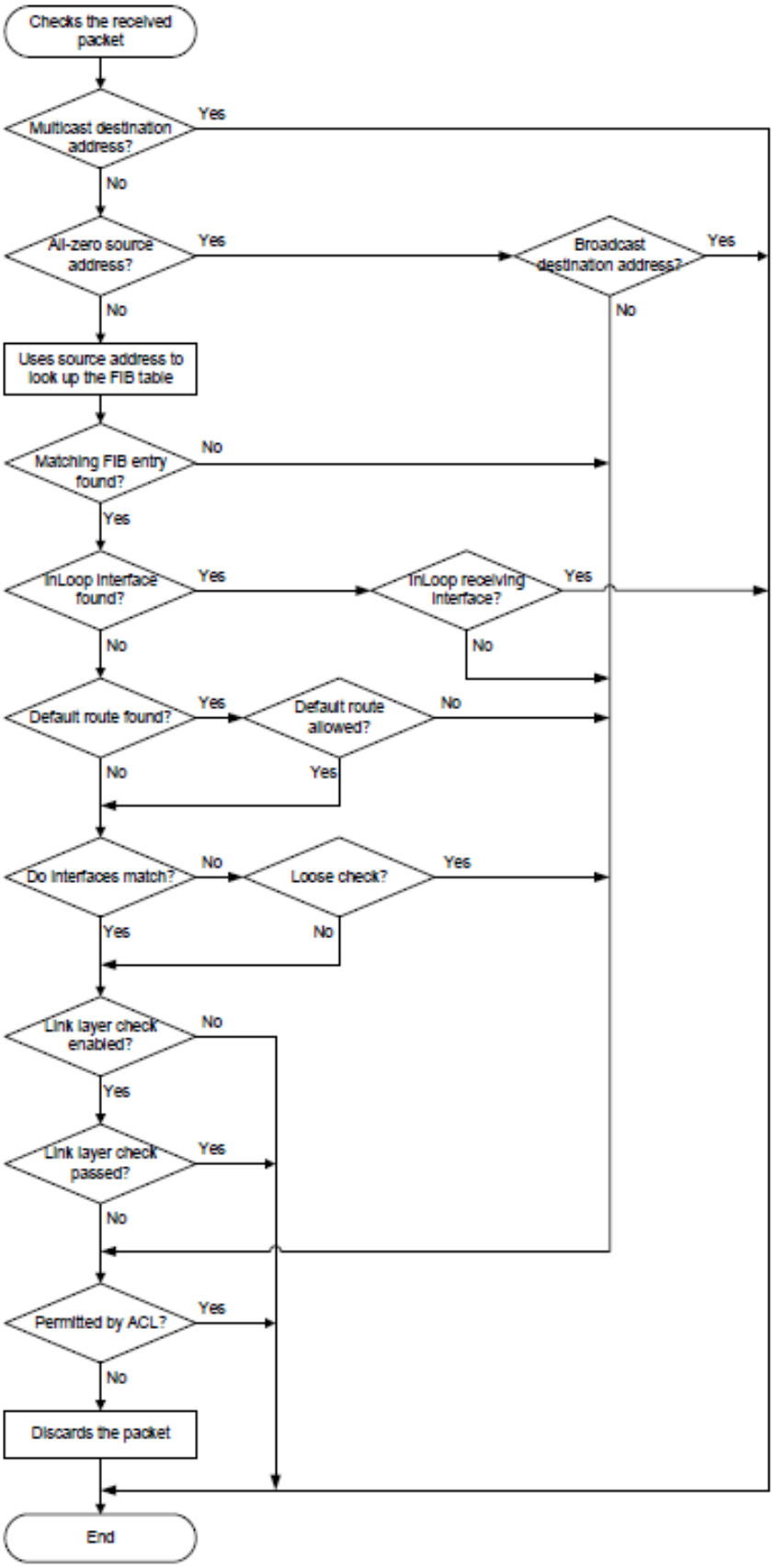
通常、PEデバイスにはCEを指すデフォルトルートがないため、PEデバイスでallow-default-routeキーワードを設定する必要はありません。CEインターフェイスでuRPFをイネーブルにし、CEインターフェイスにPEを指すデフォルトルートがある場合は、allow-default-routeキーワードを指定します。

uRPFチェック免除のためのACLの使用

特定のパケットを有効なパケットとして識別するには、ACLを使用してこれらのパケットを照合します。パケットがuRPFチェックに合格しない場合でも、転送されます。

uRPF操作

図33 uRPFのワークフロー



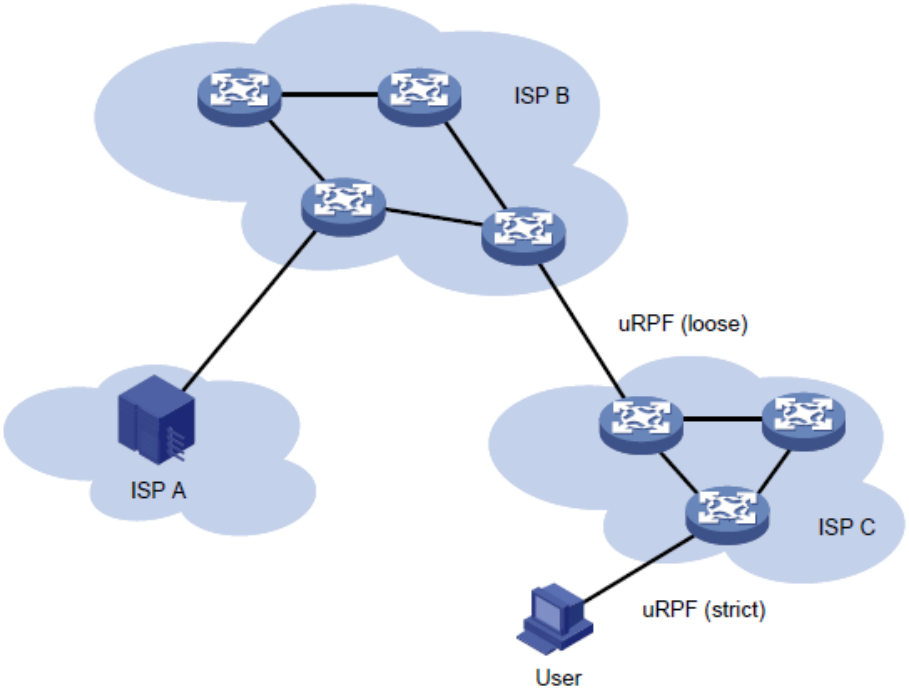
1. uRPFはアドレスの有効性をチェックします。
   - uRPFは、マルチキャスト宛先アドレスを持つパケットを許可します。
   - すべてゼロの送信元アドレスを持つパケットの場合、uRPFは、ブロードキャスト宛先アドレスを持つパケットを許可します(送信元アドレス0.0.0.0と宛先アドレス255.255.255.255を持つパケットは、DHCPまたはBOOTPパケットである可能性があり、廃棄できません)。uRPFは、パケットに非ブロードキャスト宛先アドレスがある場合、ステップ7に進みます。
   - uRPFは、他のパケットについてはステップ2に進みます。
2. uRPFは、送信元アドレスがユニキャストルートと一致するかどうかを確認します。
   - 「はい」の場合、uRPFはステップ3に進みます。
   - そうでない場合、uRPFはステップ7に進みます。非ユニキャスト送信元アドレスは、非ユニキャストルートと一致します。
3. uRPFは、一致するルートがホスト自体に対するものであるかどうかを確認します。
   - Yesの場合、一致するルートの出力インターフェイスはInLoopインターフェイスです。uRPFは、パケットの受信インターフェイスがInLoopインターフェイスであるかどうかをチェックします。Yesの場合、パケットをチェックしません。Noの場合、ステップ7に進みます。
   - 「いいえ」の場合、uRPFはステップ4に進みます。
4. uRPFは、一致するルートがデフォルトルートであるかどうかを確認します。
   - yesの場合、uRPFはallow-default-routeキーワードがデフォルトルートの使用を許可するように設定されているかどうかを確認します。yesの場合は、ステップ5に進みます。noの場合は、ステップ7に進みます。
   - そうでない場合、uRPFはステップ5に進みます。
5. uRPFは、受信インターフェイスが、一致するFIBエントリの出力インターフェイスと一致するかどうかを確認します。
   - 「はい」の場合、uRPFはステップ6に進みます。
   - Noの場合、uRPFはチェックモードが緩いかどうかをチェックします。Yesの場合、ステップ7に進みます。Noの場合、ステップ6に進みます。
6. uRPFは、リンクレイヤチェックにlink-checkキーワードが設定されているかどうかを確認します。
   - noの場合、パケットはチェックを通過します。
   - 「はい」の場合、uRPFはFIBエントリのネクストホップアドレスを使用して、一致するエントリのARPテーブルを検索します。次に、一致するARPエントリのMACアドレスがパケットの送信元MACアドレスと同じかどうかをチェックします。「はい」の場合、パケットはチェックに合格します。「いいえ」の場合、uRPFはステップ7に進みます。
7. uRPFは、パケットがACLで許可されているかどうかを確認します。
   - yesの場合、パケットは転送されます(このようなパケットは、uRPF情報に「suppressed drop」として表示されます)。
   - noの場合、パケットは破棄されます。

ネットワークアプリケーション

図34に示すように、ISPネットワークとカスタマーネットワークの間には、厳密なuRPFチェックが設定されています。ISP間には、緩やかなuRPFチェックが設定されています。

特殊なパケットまたはユーザーに対しては、ACLを設定できます。

図34 ネットワークダイアグラム



制約事項およびガイドライン:uRPFの設定

インターフェイスでuRPFをイネーブルにする場合は、display ip interfaceコマンドを使用して、uRPFによって廃棄されたパケットに関する統計情報を表示できます(「Drops」および「Suppressed drops」として表示されます)。

Loose uRPFチェックには、allow-default-routeキーワードを設定しないでください。設定すると、uRPFが動作しなくなる可能性があります。

ネットワークでECMPルーティングが使用可能な場合は、厳密なuRPFを使用しないでください。ECMPルートに沿って移動するサービスパケットは、厳密なuRPFチェックを通過できず、ドロップされます。

インターフェイスでのuRPFのイネーブル化

1. システムビューに入ります。

   system-view
2. インターフェイスビューを入力します。

   interface *interface-type interface-number*
3. uRPFをイネーブルにします。

   ip urpf { **loose** [**allow-default-route**] [**acl** *acl-number*] | **strict** [**allow-default-route**] [**acl** *acl-number*] [**link-check**] }

   デフォルトでは、uRPFはディセーブルです。

uRPFの表示およびメンテナンスコマンド

任意のビューで表示コマンドを実行します。

タスク	コマンド
uRPF設定を表示します。	スタンドアロンモードの場合: **display ip urpf** [**interface** ***interface-type interface-number***] IRFモードの場合: **display ip urpf** [**interface** *interface-type interface-number*] [**slot** *slot-number*]

uRPFの設定例

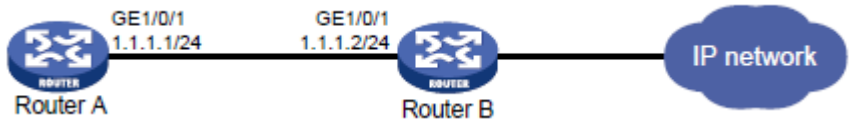
例:インターフェイスのuRPFの設定

ネットワーク構成

図35に示すように、次のタスクを実行します。

- ルーターBのGigabitEthernet 1/0/1で厳密なuRPFチェックを設定し、ネットワーク10.1.1.0/24からのパケットを許可します。
- ルーターAのGigabitEthernet 1/0/1に厳密なuRPFチェックを設定し、uRPFチェックにデフォルトルートを使用できるようにします。

図35ネットワークダイアグラム



手順

1. ルーターBを設定します。

   #ネットワーク10.1.1.0/24からのトラフィックを許可するようにACL 2010を設定します。

   <RouterB> system-view

   [RouterB] acl basic 2010

   [RouterB-acl-ipv4-basic-2010] rule permit source 10.1.1.0 0.0.0.255
   [RouterB-acl-ipv4-basic-2010] quit

   #GigabitEthernet 1/0/1のIPアドレスを指定します。
   [RouterB] interface gigabitethernet 1/0/1

   [RouterB-GigabitEthernet1/0/1] ip address 1.1.1.2 255.255.255.0

   #GigabitEthernet 1/0/1で厳密なuRPFチェックを設定します。

   [RouterB-GigabitEthernet1/0/1] ip urpf strict acl 2010

2. ルーターAを設定します。

   #GigabitEthernet 1/0/1のIPアドレスを指定します。

<RouterA> system-view

[RouterA] interface gigabitethernet 1/0/1

[RouterA-GigabitEthernet1/0/1] ip address 1.1.1.1 255.255.255.0

#GigabitEthernet 1/0/1に厳密なuRPFチェックを設定し、uRPFチェックにデフォルトルートを使用できるようにします。

[RouterA-GigabitEthernet1/0/1] ip urpf strict allow-default-route

IPv6 uRPFの設定

IPv6 uRPFについて

IPv6 Unicast Reverse Path Forwarding(uRPF)は、DoS攻撃やDDoS攻撃などの送信元アドレススプーフィング攻撃からネットワークを保護します。

IPv6 uRPFアプリケーションのシナリオ

攻撃者は、許可されたユーザーまたは管理者の名前で、IPv 6ベースの認証を使用するシステムにアクセスするために、偽造された送信元アドレスを持つパケットを送信します。攻撃者または他のホストが応答パケットを受信できない場合でも、攻撃は攻撃されたターゲットを混乱させます。

図36 送信元アドレススプーフィング攻撃

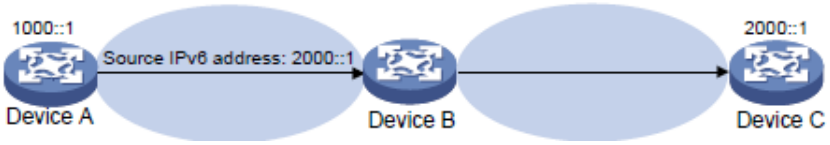


図36に示すように、デバイスAの攻撃者は、偽造された送信元IPv6アドレス2000::1を含むサーバー(デバイスB)要求を高いレートで送信します。デバイスBは、IPv6アドレス2000::1(デバイスC)に応答パケットを送信します。その結果、デバイスBとデバイスCの両方が攻撃されます。管理者が誤ってデバイスCを切断すると、ネットワークサービスが中断されます。

攻撃者は、異なる偽造された送信元アドレスを持つパケットを送信したり、複数のサーバーを同時に攻撃して接続をブロックしたり、ネットワークを破壊したりすることもできます。

IPv6 uRPFは、これらの送信元アドレススプーフィング攻撃を防止できます。uRPFは、パケットを受信したインターフェイスが、パケットの送信元アドレスと一致するFIBエントリの出力インターフェイスであるかどうかを確認します。一致しない場合、IPv6 uRPFはスプーフィング攻撃と見なし、パケットを廃棄します。

IPv6 uRPFチェックモード

IPv6 uRPFは、厳密なチェックモードと緩いチェックモードをサポートします。

厳密なIPv6 uRPFチェック

厳密なIPv6 uRPFチェックに合格するには、パケットの送信元アドレスと受信インターフェイスが、IPv6 FIBエントリの宛先アドレスと出力インターフェイスに一致する必要があります。一部のシナリオ(非対称ルーティングなど)では、厳密なIPv6 uRPFによって有効なパケットが廃棄される場合があります。

厳密なIPv6 uRPFは、多くの場合、PEとCEの間に展開されます。

緩いIPv6 uRPFチェック

Loose IPv6 uRPFチェックを通過させるには、パケットの送信元アドレスがIPv6 FIBエントリの宛先アドレスと一致する必要があります。Loose IPv6 uRPFは、有効なパケットの廃棄を回避できますが、攻撃パケットを放棄する可能性があります。

Loose IPv6 uRPFは、特に非対称ルーティングにおいて、ISP間で展開されることがよくあります。

IPv6 uRPF拡張関数

IPv6 uRPFチェックでのデフォルトルートの使用

デフォルトルートが存在する場合、特定のIPv6 FIBエントリに一致しなかったすべてのパケットは、IPv6 uRPFチェック中にデフォルトルートに一致するため、通過が許可されます。(allow-default-routeを使用して)デフォルトルートの使用を許可すると、IPv6 uRPFはデフォルトルートに一致するパケットだけを許可します。

デフォルトでは、IPv6 uRPFはデフォルトルートにしか一致しないパケットを廃棄します。

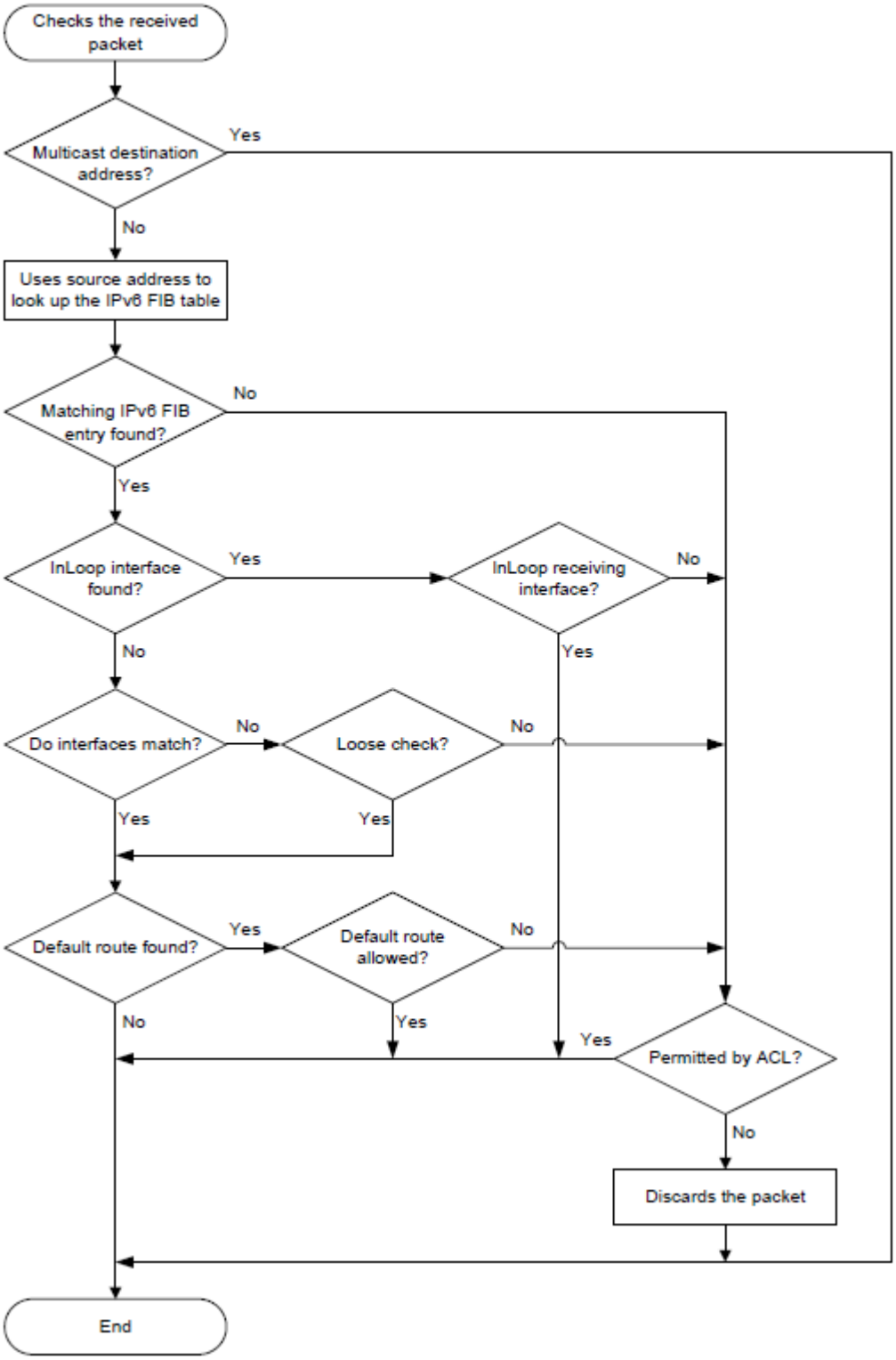
通常、PEデバイスにはCEデバイスを指すデフォルトルートがないため、PEデバイスでallow-default-routeキーワードを設定する必要はありません。CEインターフェイスでuRPFをイネーブルにし、CEインターフェイスにPEを指すデフォルトルートがある場合は、allow-default-routeキーワードを指定します。

IPv6 uRPFチェック免除のためのACLの使用

特定のパケットを有効なパケットとして識別するには、IPv6 ACLを使用してこれらのパケットを照合します。パケットがIPv6 uRPFチェックに合格しない場合でも、転送されます。

IPv6 uRPF動作

図37 IPv6 uRPFのワークフロー



1. IPv6 uRPFは、受信したパケットにマルチキャスト宛先アドレスが含まれているかどうかを確認します。

- yesの場合、IPv6 uRPFはパケットを許可します。
- 「いいえ」の場合、IPv6 uRPFは手順2に進みます。

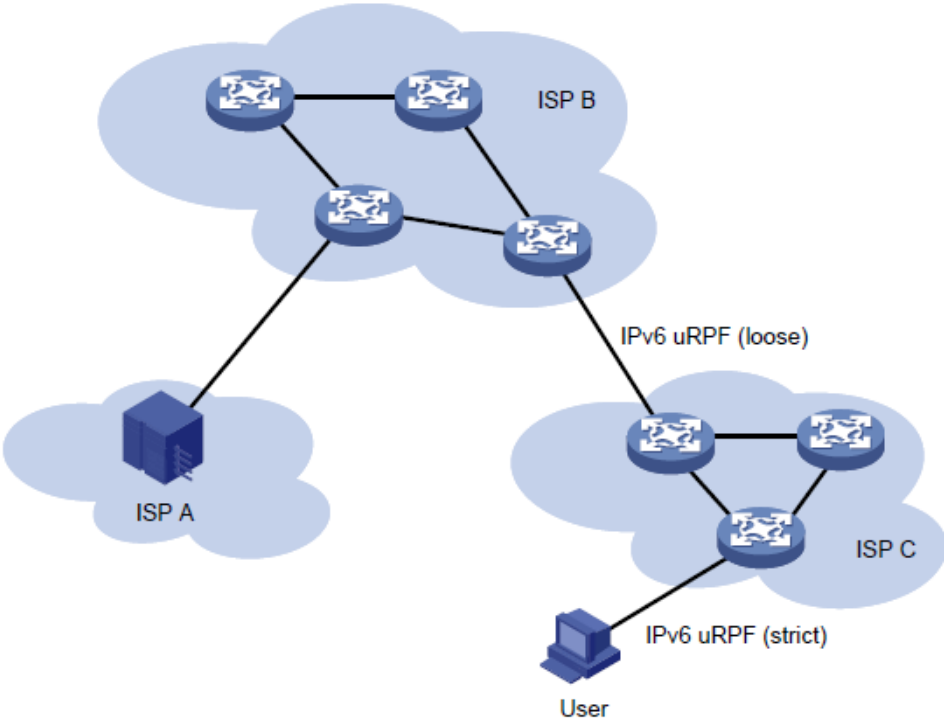
2. IPv6 uRPFは、送信元アドレスがユニキャストルートと一致するかどうかを確認します。
   - 「はい」の場合、IPv6 uRPFは手順3に進みます。
   - 一致しない場合、IPv6 uRPFはステップ6に進みます。非ユニキャスト送信元アドレスは、非ユニキャストルートと一致します。
3. IPv6 uRPFは、一致するルートがホスト自体に対するものであるかどうかを確認します。
   - 「はい」の場合、一致するルートの出力インターフェイスはInLoopインターフェイスです。IPv6 uRPFは、パケットの受信インターフェイスがInLoopインターフェイスであるかどうかを確認します。「はい」の場合、IPv6 uRPFはパケットを許可します。「いいえ」の場合、IPv6 uRPFは手順6に進みます。送信元アドレスがリンクローカルアドレスで、受信インターフェイスアドレスである場合も、手順6に進みます。
   - 「いいえ」の場合、IPv6 uRPFは手順4に進みます。
4. IPv6 uRPFは、受信インターフェイスが、一致するFIBエントリの出力インターフェイスと一致するかどうかを確認します。
   - 「はい」の場合、IPv6 uRPFは手順5に進みます。
   - noの場合、IPv6 uRPFはチェックモードが緩いかどうかをチェックします。yesの場合は、ステップ5に進みます。noの場合は、ステップ6に進みます。
5. IPv6 uRPFは、一致するルートがデフォルトルートであるかどうかを確認します。
   - yesの場合、IPv6 uRPFは、allow-default-routeキーワードがデフォルトルートの使用を許可するように設定されているかどうかを確認します。yesの場合、パケットは転送されます。noの場合、IPv6 uRPFはステップ6に進みます。
   - noの場合、パケットは転送されます。
6. IPv6 uRPFは、パケットがIPv6 ACLで許可されているかどうかを確認します。
   - yesの場合、パケットは転送されます(このようなパケットは、uRPF情報に「suppressed drop」として表示されます)。
   - noの場合、パケットは破棄されます。

ネットワークアプリケーション

図38に示すように、ISPネットワークとカスタマーネットワークの間には、厳密なIPv6 uRPFチェックが設定されています。ISP間には、緩やかなIPv6 uRPFチェックが設定されています。

特殊なパケットまたはユーザーに対しては、IPv6 ACLを設定できます。

図38 ネットワークダイアグラム



制約事項およびガイドライン:IPv6 uRPF設定

インターフェイスでipv6 uRPFをイネーブルにする場合、display ipv6 interfaceコマンドを使用して、ipv6 uRPFによって廃棄されたパケットに関する統計情報を表示できます(「Drops」および「Suppressed drops」として表示されます)。

Loose IPv6 uRPFチェックには、allow-default-routeキーワードを設定しないでください。設定すると、IPv6 uRPFが動作しなくなる可能性があります。

ネットワークでECMPルーティングが使用可能な場合は、厳密なIPv6 uRPFを使用しないでください。ECMPルートに沿って移動するサービスパケットは、厳密なuRPFチェックを通過できず、ドロップされます。

インターフェイスでのIPv6 uRPFのイネーブル化

1. システムビューに入ります。
   system-view
2. インターフェイスビューを入力します。
   interface *interface-type interface-number*
3. IPv6 uRPFをイネーブルにします。
   ipv6 urpf { **loose** | **strict** } [**allow-default-route**] [**acl** *acl-number*]
   デフォルトでは、IPv6 uRPFはディセーブルです。

IPv6 uRPFの表示およびメンテナンスコマンド

任意のビューで表示コマンドを実行します。

タスク	コマンド
IPv6 uRPF設定を表示します。	スタンドアロンモードの場合: **display ipv6 urpf** [**interface** ***interface-type interface-number***] IRFモードの場合: **display ipv6 urpf** [**interface** *interface-type interface-number*] [**slot** *slot-number*]

IPv6 uRPFの設定例

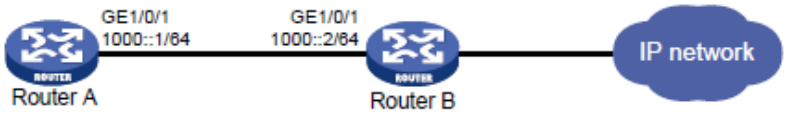
例:インターフェイスのIPv6 uRPFの設定

ネットワーク構成

図39に示すように、次のタスクを実行します。

- ルーターBのGigabitEthernet 1/0/1で厳密なIPv6 uRPFチェックを設定し、ネットワーク1010::/64からのパケットを許可します。
- ルーターAのGigabitEthernet 1/0/1で厳密なIPv6 uRPFチェックを設定し、IPv6 uRPFチェックにデフォルトルートを使用できるようにします。

図39ネットワークダイアグラム



手順

1. ルーターBを設定します。

   #IPv6 ACL 2010を設定して、ネットワーク1010::/64からのトラフィックを許可します。

   <RouterB> system-view

   [RouterB] acl ipv6 basic 2010

   [RouterB-acl-ipv6-basic-2010] rule permit source 1010:: 64
   [RouterB-acl-ipv6-basic-2010] quit

   #GigabitEthernet 1/0/1のIPv6アドレスを指定します。
   [RouterB] interface gigabitethernet 1/0/1

   [RouterB-GigabitEthernet1/0/1] ipv6 address 1000::2/64

   #GigabitEthernet 1/0/1で厳密なuRPFチェックを設定します。

   [RouterB-GigabitEthernet1/0/1] ipv6 urpf strict acl 2010

2. ルーターAを設定します。

   #GigabitEthernet 1/0/1のIPv6アドレスを指定します。

   <RouterA> system-view

   [RouterA] interface gigabitethernet 1/0/1

   [RouterA-GigabitEthernet1/0/1] ipv6 address 1000::1/64

#GigabitEthernet 1/0/1に厳密なuRPFチェックを設定し、IPv6 uRPFチェックのデフォルトルートの使用を許可します。

[RouterA-GigabitEthernet1/0/1] ipv6 urpf strict allow-default-route