サービスポリシー

Firepower Threat Defense サービス ポリシーを使用して、特定のトラフィック クラスにサービスを適用することができます。たとえば、サービス ポリシーを使用すると、すべての TCP アプリケーションに適用されるタイムアウト コンフィギュレーションではなく、特定の TCP アプリケーションに固有のタイムアウト コンフィギュレーションを作成できます。サービス ポリシーは、1 つのインターフェイスに適用されるか、またはグローバルに適用される複数のアクションまたはルールで構成されます。

Firepower Threat Defense のサービス ポリシーについて

Firepower Threat Defense サービス ポリシーを使用して、特定のトラフィック クラスにサービスを適用することができます。サービスポリシーを使用すると、デバイスまたは特定のインターフェイスに着信するすべての接続に同じサービス以外を適用することができます。

トラフィック クラスはインターフェイスと拡張アクセス コントロール リスト（ACL）の組み合わせです。ACL の「許可」ルールによってクラスに含まれる接続が決定されます。ACL の「拒否」トラフィックには、そのトラフィックに適用されているサービスがないというだけで、これらの接続は実際にはドロップされません。IP アドレスと TCP/UCP ポートを使用し、必要な精度で対応する接続を特定できます。

トラフィック クラスには 2 つのタイプがあります。

- インターフェイスベースのルール：サービス ポリシー ルールでセキュリティ ゾーンまたはインターフェイス グループを指定すると、インターフェイス オブジェクトに含まれているすべてのインターフェイスを通過する ACL の「許可」トラフィックにルールが適用されます。

特定の機能では、入力インターフェイスに適用されたインターフェイスベースのルールがグローバル ルールよりも常に優先されます。入力インターフェイスベースのルールを接続に適用すると、対応するグローバル ルールは無視されます。入力インターフェイスまたはグローバル ルールが適用されていない場合は、出力インターフェイスのインターフェイス サービス ルールが適用されます。

- グローバル ルール：すべてのインターフェイスにこれらのルールが適用されます。インターフェイスベースのルールを接続に適用しない場合は、グローバル ルールが確認され、ACL で「許可」されているすべての接続に適用されます。何も適用しない場合は、どのサービスも適用されずに接続が続行されます。

特定の接続が一致するのは、特定の機能のインターフェイス ベースまたはグローバルのいずれか 1 つのトラフィック クラスのみです。特定のインターフェイス オブジェクト/トラフィック フローの組み合わせには設定できるルールは 1 つのみです。

サービス ポリシーのルールは、アクセス制御ルールの後に適用されます。これらのサービスは、許可している接続にのみ設定されます。

FlexConfig とその他の機能にサービス ポリシーを関連付ける方法

バージョン 6.3(0) よりも前では、接続関連のサービス ルールは TCP_Embryonic_Conn_Limit と TCP_Embryonic_Conn_Timeout の事前定義の FlexConfig オブジェクトを使用して設定できました。これらのオブジェクトを削除し、Firepower Threat Defense Service サービス ポリシーを使用してルールを作り直す必要があります。これらの接続関連コマンドの実装にカスタム FlexConfig オブジェクトを作成した場合（**set connection** コマンド）は、それらのオブジェクトも削除し、サービス ポリシー経由で機能を実装する必要があります。

接続関連のサービス ポリシーの機能は、その他のサービスルールで実装された機能とは異なる機能グループとして処理されます。そのため、トラフィック クラスが重複する問題に直面することはありません。ただし、次を設定する際には十分注意してください。

- QoS ポリシー ルールはサービス ポリシー CLI を使用して実装されます。これらのルールは接続ベースのサービス ポリシー ルールよりも前に適用されます。ただし、QoS と接続の両方の設定を同じトラフィック クラスか、または重複するトラフィック クラスに適用できます。

- FlexConfig ポリシーを使用してカスタマイズされたアプリケーションのインスペクションと NetFlow を実装できます。**show running-config** コマンドを使用して、サービス ルールをすでに設定している **policy-map** コマンド、**class-map** コマンド、**service-policy** コマンドなど、CLI を調査できます。NetFlow とアプリケーション インスペクションは QoS および接続の設定との互換性はありますが、FlexConfig を実装する前に既存の設定を把握しておく必要があります。接続の設定は、アプリケーション インスペクションと NetFlow よりも前に適用されます。

（注） Firepower Threat Defense サービス ポリシーから作成されたトラフィック クラスは **class_map_***ACLname* という名前になります。*ACLname* はサービス ポリシー ルールで使用された拡張 ACL オブジェクトの名前。

What are connection settings?

Connection settings comprise a variety of features related to managing traffic connections, such as a TCP flow through the Firewall Threat Defense. Some features are named components that you would configure to supply specific services.

Connection settings include the following:

- **Global timeouts for various protocols**—All global timeouts have default values, so you need to change them only if you are experiencing premature connection loss. You configure global timeouts in the Threat Defense platform policy. Select **Devices** > **Platform Settings**.
- **Connection timeouts per traffic class**—You can override the global timeouts for specific types of traffic using service policies. All traffic class timeouts have default values, so you do not have to set them.
- **Connection limits and TCP Intercept**—By default, there are no limits on how many connections can go through (or to) the Firewall Threat Defense. You can set limits on particular traffic classes using service policy rules to protect servers from denial of service (DoS) attacks. Particularly, you can set limits on embryonic connections (those that have not finished the TCP handshake), which protects against SYN flooding attacks. When embryonic limits are exceeded, the TCP Intercept component gets involved to proxy connections and ensure that attacks are throttled.
- **Dead connection detection (DCD)**—If you have persistent connections that are valid but often idle, so that they get closed because they exceed idle timeout settings, you can enable dead connection detection to identify idle but valid connections and keep them alive (by resetting their idle timers). Whenever idle times are exceeded, DCD probes both sides of the connection to see if both sides agree the connection is valid. The **show service-policy** command output includes counters to show the amount of activity from DCD. You can use the **show conn detail** command to get information about the initiator and responder and how often each has sent probes.
- **TCP sequence randomization**—Each TCP connection has two initial sequence numbers (ISN): one generated by the client and one generated by the server. By default, the Firewall Threat Defense randomizes the ISN of the TCP SYN passing in both the inbound and outbound directions. Randomization prevents an attacker from predicting the next ISN for a new connection and potentially hijacking the new session. However, TCP sequence randomization effectively breaks TCP SACK (Selective Acknowledgement), as the sequence numbers the client sees are different from what the server sees. You can disable randomization per traffic class if desired.
- **TCP normalization**—The TCP Normalizer protects against abnormal packets. You can configure how some types of packet abnormalities are handled by traffic class. You can configure TCP Normalization using the FlexConfig policy.
- **TCP state bypass**—You can bypass TCP state checking if you use asymmetrical routing in your network.

サービスポリシーの要件と前提条件

Model support

Firewall Threat Defense

Supported domains

Any

User roles

Admin

Access Admin

Network Admin

サービス ポリシーのガイドラインと制限事項

- サービス ポリシーは、ルーテッド モードまたはトランスペアレント モードのいずれかのルーテッド インターフェイスまたはスイッチ インターフェイスのみに適用されます。インライン セットまたはパッシブ インターフェイスには適用されません。
- 特定のインターフェイスまたはグローバル ポリシーに最大 25 のトラフィック クラスを設定できます。つまり、25 を超えるサービス ポリシー ルールを特定のセキュリティ ゾーンまたはインターフェイス グループのグローバル ポリシーに設定することはできません。ただし、インターフェイスの場合、同じインターフェイスをセキュリティ ゾーンとインターフェイス グループに表示できるため、実際の制限はゾーンやグループではなく、インターフェイスに基づきます。したがって、ゾーン/グループのメンバーシップに基づき、ゾーン/グループごとに 25 のルールを設定できない場合があります。
- 特定のインターフェイス オブジェクト/トラフィック フローの組み合わせに設定できるルールは 1 つのみです。
- コンフィギュレーションに対してサービス ポリシーの変更を加えた場合は、すべての新しい接続で新しいサービス ポリシーが使用されます。既存の接続では、その接続が確立された時点で設定されていたポリシーの使用が続行されます。すべての接続に新しいポリシーをすぐに使用するには、現在の接続を切断し、新しいポリシーを使用して再度接続できるようにする必要があります。SSH またはコンソール CLI セッションから **clear conn** コマンドまたは **clear local-host** コマンドを入力します。

Threat Defense サービスポリシーの設定

Threat Defense サービスポリシーを使用して、特定のトラフィッククラスにサービスを適用できます。たとえば、サービス ポリシーを使用すると、すべての TCP アプリケーションに適用されるタイムアウト コンフィギュレーションではなく、特定の TCP アプリケーションに固有のタイムアウト コンフィギュレーションを作成できます。サービス ポリシーは、1 つのインターフェイスに適用されるか、またはグローバルに適用される複数のアクションまたはルールで構成されます。

手順

ステップ 1 **[ポリシー（Policies）] > [アクセス制御（Access Control）]**を選択し、編集する Threat Defense サービスポリシーのアクセス コントロール ポリシーで **Edit** (✎) をクリックします。

ステップ 2 パケットフロー行の最後にある [詳細（More）] ドロップダウン矢印から [詳細設定（Advanced Settings）] をクリックします。

ステップ 3 [Threat Defenseサービスポリシー（Threat Defense Service Policy）] グループで **Edit** (✎) をクリックします。

既存のポリシーが表示されたダイアログボックスが開きます。ポリシーは番号付きのルールのリストから構成されており、グローバル ルール（すべてのインターフェイスに適用）とインターフェイスベースのルールに分かれています。テーブルには、インターフェイス オブジェクトおよび拡張アクセス コントロール リスト名（これらの組み合わせでルールのトラフィッククラスを定義）と適用されたサービスが表示されます。

ステップ 4 次のいずれかを実行します。

- [ルールの追加（Add Rule）] をクリックして、新しいルールを作成します。サービス ポリシー ルールの設定 （6 ページ）を参照してください。
- **Edit** (✎) をクリックして、既存のルールを編集します。サービス ポリシー ルールの設定（6 ページ）を参照してください。
- **Delete** (🗑) をクリックしてルールを削除します。
- ルールをクリックし、移動先の新しい場所までドラッグします。インターフェイスとグローバル リスト間ではルールはドラッグできません。その代りに、ルールを編集してインターフェイス/グローバル設定を変更する必要があります。接続と一致するリスト内の最初のルールが接続に適用されます。

ステップ 5 ポリシーの編集が終了したら、[OK] をクリックします。

ステップ 6 [詳細（Advanced）] ウィンドウで [保存（Save）] をクリックします。[保存（Save）] をクリックするまで、変更は保存されません。

サービス ポリシー ルールの設定

特定のトラフィック クラスにサービスを適用するサービス ポリシー ルールを設定します。

始める前に

[オブジェクト（Objects）] > [オブジェクト管理（Object Management）] > [アクセス リスト（Access List）] > [拡張（Extended）] に移動し、ルールが適用されるトラフィックを定義する拡張アクセス リストを作成します。ルールは、拡張アクセス リスト内の許可ルールに一致するすべての接続に適用されます。ACL ルールは正確に定義し、サービスが必要なトラフィックにのみサービス ポリシー ルールが適用されるようにします。

インターフェイスベースのルールを作成している場合は、割り当てられたデバイスにインターフェイスを設定し、それらのインターフェイスをセキュリティ ゾーンまたはインターフェイス グループに追加する必要もあります。

手順

ステップ 1 [Threat Defenseサービスポリシー（Threat Defense Service Policy）] ダイアログボックスがまだ表示されていない場合は、**[ポリシー（Policies）] > [アクセス制御（Access Control）]** を選択してアクセス コントロール ポリシーを編集し、パケットフロー行の最後にある [詳細（More）] ドロップダウン矢印から [詳細設定（Advanced Settings）] を選択して、[Threat Defenseサービスポリシー（Threat Defense Service Policy）] を編集します。

ステップ 2 次のいずれかを実行します。

- [ルールの追加（Add Rule）] をクリックして、新しいルールを作成します。
- **Edit** (✎) をクリックして、既存のルールを編集します。

サービス ポリシー ルール ウィザードが開き、ルールの設定プロセスの手順が表示されます。

ステップ 3 [インターフェイス オブジェクト（Interface Object）] 手順で、ポリシーを使用するインターフェイスを定義するオプションを選択します。

- [グローバルに適用（Apply Globally）]：すべてのインターフェイスに適用されるグローバル ルールを作成するには、このオプションを選択します。
- [インターフェイス オブジェクトを選択（Select Interface Objects）]：インターフェイスベースのルールを作成するには、このオプションを選択します。次に、目的のインターフェイスを含むセキュリティ ゾーンまたはインターフェイス オブジェクトを選択し、[>] をクリックして、それらを [次（Next）] 選択済みリストに移動します。サービス ポリシー ルールは、選択したオブジェクトに含まれる各インターフェイスで設定されますが、ゾーンやグループ自体には設定されません。

インターフェイスの基準が完成したらクリックします。

ステップ 4 [トラフィック フロー（Traffic Flow）] 手順で、ルールが適用される接続を定義する拡張 ACL オブジェクトを選択して [次へ（Next）] をクリックします。

ステップ 5 [接続の設定（Connection Setting）] 手順で、このトラフィック クラスに適用するサービスを設定します。

- [TCP 状態バイパスの有効化（Enable TCP State Bypass）]（TCP 接続のみ）：TCP 状態バイパスを実装します。TCP 状態バイパスの対象である接続は、インスペクション エンジンによる検査はされず、すべての TCP 状態のチェックと TCP 正規化をバイパスします。詳細については、Bypass TCP state checks for asymetrical routing (TCP state bypass) （9 ページ）を参照してください。

  （注）
  TCP状態バイパスは、トラブルシューティングのために、または非対称ルーティングを解決できない場合に使用します。この機能は複数のセキュリティ機能を無効化するため、定義が狭いトラフィック クラスを指定して適切に実装しないと、多数の接続が発生することがあります。

- [TCP シーケンス番号のランダム化（Randomize TCP Sequence Number）]（TCP 接続のみ）：TCP シーケンス番号のランダム化を有効にするか、無効にするかを示します。デフォルトでは、ランダム化が有効になっています。詳細については、TCP シーケンスのランダム化のディセーブル （14 ページ）を参照してください。

- [デクリメント TTL の有効化（Enable Decrement TTL）]（TCP 接続のみ）：クラスに一致するパケットの存続可能時間（TTL）をデクリメントします。存続可能時間を減らすと、TTL が 1 のパケットはドロップされますが、接続に TTL がもっと長いパケットが含まれている可能性があるという仮定の下に、セッションに対して接続が開かれます。OSPF hello パケットなどの一部のパケットは TTL が 1 で送信されるため、存続可能時間を減らすと予期しない結果が生じることがある点に注意してください。

  （注）
  Firewall Threat Defense デバイスをトレースルートに表示する場合は、デクリメント TTL オプションを設定し、プラットフォームの設定のポリシーに ICMP 到達不能レート制限も設定する必要があります。Firewall Threat Defense デバイスをトレースルートに表示する（19 ページ）を参照してください。

- [接続（Connections）]：クラス全体で許可される接続の数を制限します。次のオプションを設定可能です。

  - [TCP と UDP の最大数（Maximum TCP and UDP）]（TCP または UDP 接続のみ）：クラス全体で許可される同時接続の最大数（0 ～ 2000000）。TCP の場合、この数は確立された接続にのみ適用されます。デフォルトは 0 で、この場合は接続数が制限されません。制限がクラスに適用されるため、1 つの攻撃ホストがすべての接続を使い果たし、クラスに一致する他のホストが使用できる接続がなくなる可能性があります。この問題を改善するには、クライアントごとの制限を設定します。

  - [最大初期接続数（Maximum Embryonic）]（TCP 接続のみ）：許可される TCP の同時初期接続（TCP ハンドシェイクで完了しない接続）の最大数（0~2000000）。デフォルトは 0 で、この場合は接続数が制限されません。0 以外の制限を設定することで、

TCP 代行受信をイネーブルにします。代行受信によって、TCP SYN パケットを使用してインターフェイスをフラッディングする DoS 攻撃から内部システムを保護します。また、クライアントごとのオプションを設定して、SYN フラッディングから保護します。詳細については、SYN フラッド DoS 攻撃からのサーバーの保護（TCP 代行受信）（15 ページ）を参照してください。

- [クライアントあたりの接続数（Connections Per Client）]：特定のクライアント（送信元 IP アドレス）で許可される接続数の制限。次のオプションを設定可能です。
  - [TCP と UDP の最大数（Maximum TCP and UDP）]（TCP または UDP 接続のみ）：クライアントごとに許可される同時接続の最大数（0 ～ 2000000）。TCP の場合は、確立済み接続、ハーフオープン（初期）接続、ハーフクローズ接続が含まれます。デフォルトは 0 で、この場合は接続数が制限されません。このオプションでは、クラスに一致する各ホストに許可される同時接続の最大数が制限されます。
  - [最大初期接続数（Maximum Embryonic）]（TCP 接続のみ）：クライアントごとに許可される TCP の同時初期接続の最大数（0 ～ 2000000）。デフォルトは 0 で、この場合は接続数が制限されません。詳細については、SYN フラッド DoS 攻撃からのサーバーの保護（TCP 代行受信）（15 ページ）を参照してください。
- [接続の SYN Cookie MSS（Connections Syn Cookie MSS）]：初期接続数制限に達したときに初期接続の SYN cookie を生成するためのサーバーの最大セグメントサイズ（MSS）（48 ～ 65,535）。デフォルトは 1380 です。この設定は、接続またはクライアントごと、あるいは両方に対して [最大初期接続数（Maximum Embryonic）] を設定する場合にのみ意味があります。
- [接続タイムアウト（Connections Timeout）]：トラフィック クラスに適用されるタイムアウトの設定。これらのタイムアウトで、プラットフォーム設定ポリシーに定義されているグローバル タイムアウトがオーバーライドされます。次の設定を行えます。
  - [初期接続（Embryonic）]（TCP 接続のみ）：TCP 初期（ハーフオープン）接続が閉じられるまでのタイムアウト期間（0:0:5 ～ 1193:00:00）。デフォルト値は 0:0:30 です。
  - [ハーフクローズ（Half Closed）]（TCP 接続のみ）：ハーフクローズ接続が閉じられるまでのアイドル タイムアウト期間（0:0:30 ～ 1193:0:0）。デフォルト値は 0:10:0 です。ハーフクローズ接続は、Dead Connection Detection（DCD; デッド接続検出）の影響を受けません。また、システムは、ハーフクローズ接続の切断時にリセットを送信しません。
  - [アイドル（Idle）]（TCP、UDP、ICMP、IP 接続）：プロトコルの確立された接続が閉じた後のアイドル タイムアウト期間（0:0:1 ～ 1193:0:0）。デフォルトは 1:0:0 です。ただし、デフォルトが 0:2:0 である [TCP 状態バイパス（TCP State Bypass）] オプションを選択している場合を除く。
  - [タイムアウト時に接続をリセット（Reset Connection Upon Timeout）]（TCP 接続のみ）：アイドル接続が削除された後に、両方のエンド システムに TCP RST パケットを送信するかどうかを示します。

- [デッド接続の検出（Detect Dead Connections）]（TCP 接続のみ）：Dead Connection Detection（DCD; デッド接続検出）を有効にするかどうかを示します。アイドル接続の期限が切れる前に、システムはエンドホストにプローブを送信して接続が有効であるかどうかを判断します。両方のホストが応答した場合は、接続が維持されます。それ以外の場合は、接続が解放されます。トランスペアレント ファイアウォール モードで動作している場合、エンドポイントにスタティック ルートを設定する必要があります。オフロードもされている接続では DCD を構成できないため、プレフィルタポリシーで高速パス処理している接続ではDCDを構成しないでください。発信側と受信側で送信されたDCDプローブの数を追跡するには、Firewall Threat Defense CLI で **show conn detail** コマンドを使用します。

  次のオプションを設定します。

  - [検出のタイムアウト（Detection Timeout）]：DCD プローブに応答がない場合に別のプローブを送信するまで待機する時間（hh:mm:ss 形式で、0:0:1 ～ 24:0:0 の範囲で指定）。デフォルト値は 0:0:15 です。

    クラスタまたは高可用性構成で動作しているシステムでは、間隔を1分（0:1:0）未満に設定しないことを推奨します。接続をシステム間で移動する必要がある場合、必要な変更には30秒以上かかり、変更が行われる前に接続が削除される場合があります。

  - [検出の再試行（Detection Retries）]：接続がデッドであると宣言する前に行われる DCD の再試行の連続失敗回数（1 ~ 255）。デフォルトは 5 です。

ステップ 6 [終了（Finish）] をクリックして変更を保存します。

ルールは適切なリスト（インターフェイスまたはグローバル）の下部に追加されます。グローバル ルールは上から下の順に照合されます。インターフェイス リスト内のルールは、各インターフェイス オブジェクトで上から下の順に照合されます。定義が狭いトラフィック クラスのルールは、定義が広いルールの上に配置し、適切なサービスが適用されるようにします。各リスト内のルールはドラッグ アンド ドロップで移動できます。リスト間でルールを移動することはできません。

Bypass TCP state checks for asymetrical routing (TCP state bypass)

If you have an asymetrical routing environment in your network, where the outbound and inbound flow for a given connection can go through two different Firewall Threat Defense devices, you need to implement TCP state bypass on the affected traffic.

However, TCP state bypass weakens the security of your network, so you should apply bypass on very specific, limited traffic classes.

The following topics explain the problem and solution in more detail.

The asymetrical routing problem

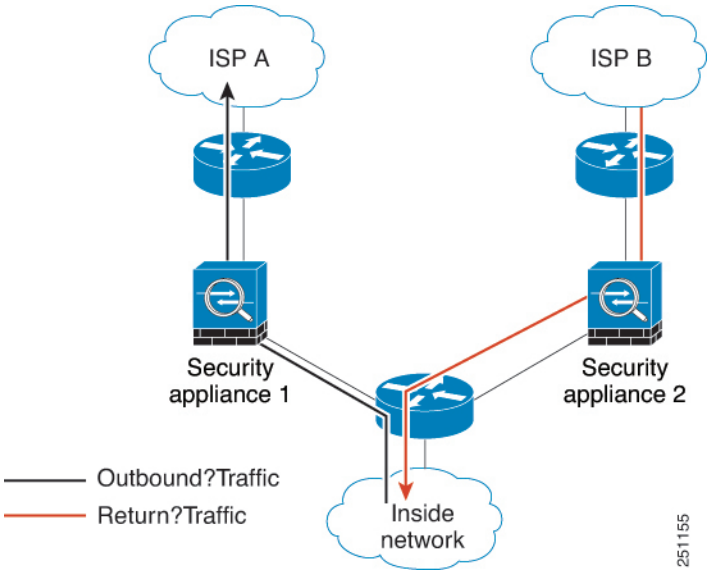
By default, all traffic that goes through the Firewall Threat Defense is inspected using the Adaptive Security Algorithm and is either allowed through or dropped based on the security policy. The Firewall Threat Defense maximizes the firewall performance by checking the state of each packet (new connection or

established connection) and assigning it to either the session management path (a new connection SYN packet), the fast path (an established connection), or the control plane path (advanced inspection).

TCP packets that match existing connections in the fast path can pass through the Firewall Threat Defense without rechecking every aspect of the security policy. This feature maximizes performance. However, the method of establishing the session in the fast path using the SYN packet, and the checks that occur in the fast path (such as TCP sequence number), can stand in the way of asymmetrical routing solutions: both the outbound and inbound flow of a connection must pass through the same Firewall Threat Defense device.

For example, a new connection goes to Security Appliance 1. The SYN packet goes through the session management path, and an entry for the connection is added to the fast path table. If subsequent packets of this connection go through Security Appliance 1, then the packets match the entry in the fast path, and are passed through. But if subsequent packets go to Security Appliance 2, where there was not a SYN packet that went through the session management path, then there is no entry in the fast path for the connection, and the packets are dropped. The following figure shows an asymmetric routing example where the outbound traffic goes through a different Firewall Threat Defense than the inbound traffic:

図 *1 : Asymmetric Routing*



If you have asymmetric routing configured on upstream routers, and traffic alternates between two Firewall Threat Defense devices, then you can configure TCP state bypass for specific traffic. TCP state bypass alters the way sessions are established in the fast path and disables the fast path checks. This feature treats TCP traffic much as it treats a UDP connection: when a non-SYN packet matching the specified networks enters the Firewall Threat Defense device, and there is not a fast path entry, then the packet goes through the session management path to establish the connection in the fast path. Once in the fast path, the traffic bypasses the fast path checks.

Guidelines and limitations for TCP state bypass

TCP state bypass unsupported features

The following features are not supported when you use TCP state bypass:

- Application inspection—Inspection requires both inbound and outbound traffic to go through the same Firewall Threat Defense, so inspection is not applied to TCP state bypass traffic.
- Snort inspection—Inspection requires both inbound and outbound traffic to go through the same device. However, Snort inspection is not automatically bypassed for TCP state bypass traffic. You must also configure a prefilter fastpath rule for the same traffic class for which you configure TCP state bypass. Otherwise, packets might be dropped unexpectedly because the TCP Normalizer is also not engaged.
- TCP Intercept, maximum embryonic connection limit, TCP sequence number randomization—The Firewall Threat Defense does not keep track of the state of the connection, so these features are not applied.
- TCP normalization—The TCP normalizer is disabled.
- TLS server identity discovery cannot be used with TCP state bypass on an inline or inline tap interface.

TCP state bypass NAT guidelines

Because the translation session is established separately for each device, be sure to configure static NAT on both devices for TCP state bypass traffic. If you use dynamic NAT, the address chosen for the session on Device 1 will differ from the address chosen for the session on Device 2.

TCP ステート バイパスの設定

非対称ルーティング環境で TCP ステートチェックをバイパスするには、影響を受けるホストまたはネットワークのみに適用するトラフィッククラスを注意深く定義してから、サービスポリシーを使用してトラフィッククラスで TCP ステートバイパスを有効にします。また、同じトラフィックに対応するプレフィルタ fastpath ポリシーを設定してトラフィックもインスペクションをバイパスさせる必要もあります。

バイパスによってネットワークのセキュリティが低下するため、そのアプリケーションをできるだけ制限します。

手順

ステップ 1 トラフィック クラスを定義する拡張 ACL を作成します。

たとえば、10.1.1.1 to 10.2.2.2 からの TCP トラフィックのトラフィック クラスを定義するには、次の手順を実行します。

a) **Objects** > **Object Management** > **Access List** > **Extended**を選択します。
b) [拡張アクセスリストを追加（Add Extended Access List）] をクリックします。
c) オブジェクトの [名前（Name）]（bypass など）を入力します。
d) [追加（Add）] をクリックしてルールを追加します。
e) アクションは [許可（Allow）] のままにします。
f) [送信元（Source）] リストの下に 10.1.1.1 と入力して [追加（Add）] をクリックし、[宛先（Destination）] リストの下に 10.2.2.2 と入力して [追加（Add）] をクリックします。

g) [ポート（Port）] をクリックし、[選択済みの送信元ポート（Selected Source Ports）] リストの下で [TCP (6)] を選択して [追加（Add）] をクリックします。ポート番号は入力せず、すべてのポートをカバーするプロトコルとして TCP を単純に追加します。

h) [拡張アクセス リスト エントリ（Extended Access List Entry）] ダイアログボックスで [追加（Add）] をクリックして、ルールを ACL に追加します。

i) [拡張アクセス リスト オブジェクト（Extended Access List Object）] ダイアログボックスで [保存（Save）] をクリックして、ACL オブジェクトを保存します。

ステップ 2 TCP ステート バイパスのサービス ポリシー ルールを設定します。

たとえば、このトラフィック クラスの TCP ステート バイパスをグローバルに設定するには、次の手順を実行します。

a) **Policies** > **Access Control heading** > **Access Control** を選択して、このサービスを必要とするデバイスに割り当てられているポリシーを編集します。

b) パケットフロー行の最後にある [詳細（More）] ドロップダウン矢印から [詳細設定（Advanced Settings）] をクリックし、[Threat Defenseサービスポリシー（Threat Defense Service Policy）] の **Edit** (✎) をクリックします。[

c) [ルールの追加（Add Rule）] をクリックします。

d) **[グローバルに適用（Apply Globally）]** > **[次へ（Next）]** を選択します。

e) このルールに対して作成した拡張 ACL オブジェクトを選択して、[次へ（Next）] をクリックします。

f) [TCP ステート バイパスの有効化（Enable TCP State Bypass）] を選択します。

g) （オプション）バイパスされる接続の [アイドル（Idle）] タイムアウトを調整します。デフォルトは 2 分です。

h) [終了（Finish）]をクリックしてルールを追加します。必要に応じて、ルールをサービスポリシー内の必要な位置にドラッグアンドドロップします。

i) [OK] をクリックして、サービス ポリシーに加えた変更を保存します。

j) [詳細（Advanced）] で [保存（Save）] をクリックして、アクセス コントロール ポリシーに加えた変更を保存します。

ステップ 3 トラフィック クラスのプレフィルタ fastpath のルールを設定します。

プレフィルタ ルール内に ACL オブジェクトを使用できません。そのため、プレフィルタ ルールに直接か、またはクラスを定義するネットワーク オブジェクトを最初に作成するかのいずれかでトラフィック クラスを再度作成する必要があります。

次の手順では、アクセス コントロール ポリシーに接続されているプレフィルタ ポリシーがすでにあることを前提としています。プレフィルタポリシーをまだ作成していない場合は、**[ポリシー（Policies）]** > **[プレフィルタ（Prefilter）]**に移動して、まずポリシーを作成します。アクセス コントロール ポリシーに接続し、ルールを作成するには、この手順を使用できます。

この手順は 10.1.1.1 から 10.2.2.2 への TCP トラフィックの fastpath ルールを作成する例に沿っています。

a) **Policies** > **Access Control heading** > **Access Control** を選択して、TCP バイパス サービス ポリシー ルールを含むポリシーを編集します。

b) ポリシーの説明のすぐ下の左側にある [プレフィルタ ポリシー（Prefilter Policy）] のリンクをクリックします。

c) [プレフィルタ ポリシー（Prefilter Policy）] ダイアログボックスで、適切なポリシーがまだ選択されていないデバイスに割り当てるポリシーを選択します。この時点ではまだ [OK] をクリックしないでください。

デフォルトのプレフィルタ ポリシーにはルールを追加できないため、カスタム ポリシーを選択する必要があります。

d) [プレフィルタ ポリシー（Prefilter Policy）] ダイアログボックスで、**Edit** (✎) をクリックします。このアクションによって、ポリシーの編集が可能な新しいブラウザウィンドウが開きます。

e) [プレフィルタ ルールの追加（Add Prefilter Rule）] をクリックし、次のプロパティを使用してルールを設定します。

- [名前（Name）]：自分にとってわかりやすい名前（TCPBypass など）。
- [アクション（Action）]：[Fastpath] を選択します。
- [インターフェイスオブジェクト（Interface Objects）]：TCP ステートバイパスをグローバルルールとして設定した場合は送信元も宛先もデフォルトの [任意（any）] のままにします。インターフェイスベースのルールを作成した場合は、[送信元インターフェイス オブジェクト（Source Interface Objects）] リストのルールに使用したものと同じインターフェイス オブジェクトを選択し、宛先は [任意（any）] のままにします。
- [ネットワーク（Networks）]：[送信元ネットワーク（Source Networks）] リストに 10.1.1.1 を、[宛先ネットワーク（Destination Networks）] リストに 10.2.2.2 を追加します。ネットワーク オブジェクトを使用するか、またはアドレスを手動で追加することができます。
- [ポート（Ports）]：[選択済み送信元ポート（Selected Source Ports）] で、TCP(6) を選択し、**ポートを入力せずに** [追加（Add）] をクリックします。こうすることで、TCP ポート番号に関係なく、すべての（および唯一の）TCP トラフィックにルールが適用されます。

f) [追加（Add）] をクリックしてプレフィルタ ポリシーにルールを追加します。

g) [保存（Save）] をクリックしてプレフィルタ ポリシーに変更を保存します。

これで、プレフィルタ編集ウィンドウを閉じてアクセス コントロール ポリシーの編集ウィンドウに戻ることができます。

h) アクセス コントロール ポリシーの編集ウィンドウには [プレフィルタ ポリシー（Prefilter Policy）] ダイアログボックスが開かれたままになっています。[OK] をクリックしてプレフィルタ ポリシーの割り当てに変更を保存します。

i) プレフィルタ ポリシーの割り当てを変更した場合は、アクセス コントロール ポリシーで [保存（Save）] をクリックしてその変更を保存します。

これで、影響を受けるデバイスに変更を展開できます。

TCP シーケンスのランダム化のディセーブル

各 TCP 接続には 2 つの初期シーケンス番号（ISN）が割り当てられており、1 つはクライアントで生成され、もう 1 つはサーバで生成されます。Firewall Threat Defense デバイスは、着信と発信の両方向で通過する TCP SYN の ISN をランダム化します。

保護対象のホストの ISN をランダム化することにより、攻撃者が新しい接続に使用される次の ISN を予測して新しいセッションをハイジャックするのを阻止します。ただし、TCP シーケンスのランダム化は、TCP SACK（選択的確認応答）を実質的に破棄します。クライアントが認識するシーケンス番号がサーバーが認識するものと異なるためです。

たとえば、データがスクランブルされるため、必要に応じて TCP 初期シーケンス番号ランダム化を無効化することができます。次に、ランダム化を無効にする状況をいくつか示します。

- 別の直列接続されたファイアウォールでも初期シーケンス番号がランダム化され、トラフィックに影響することはないものの、両方のファイアウォールでこの動作を実行する必要がない場合。
- デバイスで eBGP マルチホップを使用していて、eBGP ピアで MD5 を使用している場合。ランダム化により、MD5 チェックサムは分解されます。
- Firewall Threat Defense デバイスによる接続のシーケンス番号のランダム化が不要な WAAS デバイスを使用する場合。
- ISA 3000 のハードウェア バイパスを有効にしている場合、ISA 3000 がデータ パスの一部でなくなると、TCP 接続はドロップされます。

手順

ステップ 1 トラフィック クラスを定義する拡張 ACL を作成します。

たとえば、任意のホストから 10.2.2.2 に送信される TCP トラフィックのトラフィック クラスを定義するには、次の手順を実行します。

a) **[オブジェクト（Objects）]** > **[オブジェクト管理（Object Management）]** を選択します。
b) 目次から **[アクセス リスト（Access List）]** > **[拡張（Extended）]** を選択します。
c) [拡張アクセスリストを追加（Add Extended Access List）] をクリックします。
d) オブジェクトの [名前（Name）]（preserve-sq-no など）を入力します。
e) [追加（Add）] をクリックしてルールを追加します。
f) アクションは [許可（Allow）] のままにします。
g) [送信元（Source）] リストを空白のままにして、[宛先（Destination）] リストの下に 10.2.2.2 と入力して、[追加（Add）] をクリックします。
h) [ポート（Port）] をクリックし、[選択済みの送信元ポート（Selected Source Ports）] リストの下で [TCP (6)] を選択して [追加（Add）] をクリックします。ポート番号は入力せず、すべてのポートをカバーするプロトコルとして TCP を単純に追加します。

i) [拡張アクセス リスト エントリ（Extended Access List Entry）] ダイアログボックスで [追加（Add）] をクリックして、ルールを ACL に追加します。
j) [拡張アクセス リスト オブジェクト（Extended Access List Object）] ダイアログボックスで [保存（Save）] をクリックして、ACL オブジェクトを保存します。

ステップ 2 TCP シーケンス番号のランダム化を無効にするサービス ポリシー ルールを設定します。

たとえば、このトラフィック クラスのランダム化をグローバルに無効にするには、次の手順を実行します。

a) **[ポリシー（Policies）]** > **[アクセス制御（Access Control）]** を選択して、このサービスを必要とするデバイスに割り当てられているポリシーを編集します。
b) パケットフロー行の最後にある [詳細（More）] ドロップダウン矢印から [詳細設定（Advanced Settings）] をクリックし、[Threat Defenseサービスポリシー（Threat Defense Service Policy）] の **Edit** (✎) をクリックします。[
c) [ルールの追加（Add Rule）] をクリックします。
d) **[グローバルに適用（Apply Globally）]** > **[次へ（Next）]** を選択します。
e) このルールに対して作成した拡張 ACL オブジェクトを選択して、[次へ（Next）] をクリックします。
f) [TCP シーケンス番号のランダム化（Randomize TCP Sequence Number）] の選択を解除します。
g) （オプション）その他の接続オプションを必要に応じて調節します。
h) [終了（Finish）]をクリックしてルールを追加します。必要に応じて、ルールをサービスポリシー内の必要な位置にドラッグアンドドロップします。
i) [OK] をクリックして、サービス ポリシーに加えた変更を保存します。
j) [詳細（Advanced）] で [保存（Save）]をクリックして、アクセス コントロール ポリシーに加えた変更を保存します。

これで、影響を受けるデバイスに変更を展開できます。

サービス ポリシーのルールの例

次のトピックにサービス ポリシー ルールの例を示します。

SYN フラッド DoS 攻撃からのサーバーの保護（TCP 代行受信）

攻撃者が一連の SYN パケットをホストに送信すると、SYN フラッディング サービス妨害（DoS）攻撃が発生します。これらのパケットは通常、スプーフィングされた IP アドレスから発信されます。SYN パケットのフラッディングが定常的に生じると、SYN キューが一杯になる状況が続き、正規ユーザーからの接続要求に対してサービスを提供できなくなります。

SYN フラッディング攻撃を防ぐために初期接続数を制限できます。初期接続とは、送信元と宛先の間で必要になるハンドシェイクを完了していない接続要求のことです。

接続の初期接続しきい値を超えると、Firewall Threat Defense はサーバーのプロキシとして動作し、その接続がターゲットホストの SYN キューに追加されないように、SYN Cookie 方式を使用してクライアント SYN 要求に対する SYN-ACK 応答を生成します。SYN クッキーは、基本的に秘密を作成するために、MSS、タイムスタンプ、およびその他の項目の数学的ハッシュから構築される SYN-ACK で返される最初のシーケンス番号です。Firewall Threat Defense は、正しいシーケンス番号で有効な時間ウィンドウ内にクライアントから返された ACK を受信すると、クライアントが本物であることを認証し、サーバーへの接続を許可できます。プロキシを実行するコンポーネントは、TCP 代行受信と呼ばれます。

接続制限を設定すると、サーバを SYN フラッド攻撃から保護できます。必要に応じて、TCP 代行受信の統計情報を有効にして、ポリシーの結果をモニタできます。次の手順では、エンドツーエンドのプロセスについて説明します。

始める前に

- 保護するサーバーの TCP SYN バックログ キューより低い初期接続制限を設定していることを確認します。これより高い初期接続制限を設定すると、有効なクライアントが、SYN 攻撃中にサーバーにアクセスできなくなります。初期接続制限に適切な値を決定するには、サーバーの容量、ネットワーク、サーバーの使用状況を入念に分析してください。
- Secure Firewall Threat Defense デバイス上の CPU コア数によっては、各コアによる接続の管理方法が原因で、同時接続および初期接続の最大数が設定されている数を超える場合があります。最悪の場合、デバイスは最大 n-1 の追加接続および初期接続を許可します。ここで、n はコアの数です。たとえば、モデルに 4 つのコアがあり、6 つの同時接続および 4 つの初期接続を設定した場合は、各タイプで 3 つの追加接続を使用できます。モデルのコア数を確認するには、デバイスの CLI で **show cpu core** コマンドを入力します。

手順

ステップ 1 保護するサーバのリストであるトラフィック クラスを定義する拡張 ACL を作成します。

たとえば、IP アドレスが 10.1.1.5 と 10.1.1.6 の Web サーバーを保護するためのトラフィック クラスを定義します。

a) **[オブジェクト（Objects）]** > **[オブジェクト管理（Object Management）]** を選択します。
b) 目次から **[アクセス リスト（Access List）]** > **[拡張（Extended）]** を選択します。
c) [拡張アクセスリストを追加（Add Extended Access List）] をクリックします。
d) オブジェクトの [名前（Name）]（protected-servers など）を入力します。
e) [追加（Add）] をクリックしてルールを追加します。
f) アクションは [許可（Allow）] のままにします。
g) [送信元（Source）] リストを空白のままにして、[宛先（Destination）] リストの下に 10.1.1.5 と入力して、[追加（Add）] をクリックします。
h) また、[宛先（Destination）] リストの下に 10.1.1.6 と入力して、[追加（Add）] をクリックします。

i) [ポート（Port）] をクリックし、利用可能なポートのリストで [HTTP] を選択して、[宛先に追加（Add to Destination）] をクリックします。サーバーで HTTPS 接続もサポートされている場合は、HTTPS ポートも追加します。

j) [拡張アクセス リスト エントリ（Extended Access List Entry）] ダイアログボックスで [追加（Add）] をクリックして、ルールを ACL に追加します。

k) [拡張アクセス リスト オブジェクト（Extended Access List Object）] ダイアログボックスで [保存（Save）] をクリックして、ACL オブジェクトを保存します。

ステップ 2 初期接続制限を設定するサービス ポリシールールを設定します。

たとえば、同時初期接続の合計を 1000 接続に設定し、クライアントごとの制限を 50 接続に設定する場合は、次の手順を実行します。

a) **[ポリシー（Policies）] > [アクセス制御（Access Control）]** を選択して、このサービスを必要とするデバイスに割り当てられているポリシーを編集します。

b) パケットフロー行の最後にある [詳細（More）] ドロップダウン矢印から [詳細設定（Advanced Settings）] をクリックし、[Threat Defenseサービスポリシー（Threat Defense Service Policy）] の **Edit** (✎) をクリックします。[

c) [ルールの追加（Add Rule）] をクリックします。

d) **[グローバルに適用（Apply Globally）] > [次へ（Next）]** を選択します。

e) このルールに対して作成した拡張 ACL オブジェクトを選択して、[次へ（Next）] をクリックします。

f) **[接続（Connections）] > [最大初期接続数（Maximum Embryonic）]** に 1000 を入力します。

g) **[クライアントあたりの接続数（Connections Per Client）] > [最大初期接続数（Maximum Embryonic）]** に 50 を入力します。

h) （オプション）その他の接続オプションを必要に応じて調節します。

i) [終了（Finish）]をクリックしてルールを追加します。必要に応じて、ルールをサービスポリシー内の必要な位置にドラッグアンドドロップします。

j) [OK] をクリックして、サービス ポリシーに加えた変更を保存します。

k) [詳細（Advanced）] で [保存（Save）]をクリックして、アクセス コントロール ポリシーに加えた変更を保存します。

ステップ 3 （オプション）TCP 代行受信の統計情報のレートを設定します。

TCP代行受信では次のオプションを使用して、統計情報の収集レートが決定されます。すべてのオプションにはデフォルト値があります。それらのレートがニーズに合っている場合は、この手順を省略できます。

- [レート間隔（Rate Interval）]：履歴監視ウィンドウのサイズ（1 ～ 1440 分）。デフォルトは 30 分です。この間隔の間に、システムは攻撃の数を 30 回サンプリングします。
- [バースト レート（Burst Rate）]：Syslog メッセージ生成のしきい値（25 ～ 2147483647）。デフォルトは 1 秒間に 400 です。バースト レートを超えると、デバイスは Syslog メッセージ 733104 を生成します。

- [平均レート（Average Rate）]：Syslog メッセージ生成の平均レートのしきい値（25 ～ 2147483647）。デフォルトは 1 秒間に 200 回です。平均レートを超えると、デバイスは Syslog メッセージ 733105 を生成します。

これらのオプションを調整する場合は、次の手順を実行します。

a) **[オブジェクト（Objects）]** > **[オブジェクト管理（Object Management）]** を選択します。
b) **[FlexConfig]** > **[テキスト オブジェクト（Text Object）]** を選択します。
c) システム定義オブジェクト threat_defense_statistics の **Edit** (🔗) をクリックします。
d) 値は直接変更できますが、[オーバーライド（Override）]セクションを開き、[追加（Add）] をクリックして、デバイス オーバーライドを作成することを推奨します。
e) （アクセス コントロール ポリシーの割り当てを介して）サービス ポリシーを割り当てるデバイスを選択し、[追加（Add）] をクリックして、選択済みリストにデバイスを移動します。
f) [オーバーライド（Override）] をクリックします。
g) オブジェクトには 3 つのエントリが必要なため、3 になるまで必要に応じて [カウント（Count）] をクリックします。
h) レート間隔、バースト レート、および平均レートとして 1 ～ 3 の順序で必要な値を入力します。オブジェクトの説明を参照し、正しい順序で値を入力していることを確認してください。
i) [オブジェクトのオーバーライド（Object Override）] ダイアログボックスで [追加（Add）] をクリックします。
j) [テキスト オブジェクトの編集（Edit Text Object）] ダイアログボックスで [保存（Save）] をクリックします。

ステップ 4 TCP 代行受信の統計情報を有効にします。

TCP 代行受信の統計情報を有効にするには FlexConfig ポリシーを設定する必要があります。

a) **[デバイス（Devices）]** > **[FlexConfig]** を選択します。
b) ポリシーをすでにデバイスに割り当てている場合は、そのポリシーを編集します。割り当てていない場合は、新しいポリシーを作成して、影響を受けるデバイスに割り当てます。
c) [利用可能な FlexConfig（Available FlexConfig）] リストで [Threat_Detection_Configure] を選択して [>>] をクリックします。オブジェクトが [選択済み追加 FlexConfig（Selected Append FlexConfigs）] リストに追加されます。
d) [保存（Save）] をクリックします。
e) （オプション）[プレビュー設定（Preview Config）] をクリックし、いずれかのデバイスを選択することで、設定が正しいことを確認できます。

次の展開時にデバイスに書き込まれる CLI コマンドが生成されます。それらのコマンドには、サービス ポリシーおよび脅威検出の統計情報に必要なコマンドが含まれます。プレビューの下にスクロールして、追加された CLI を確認します。デフォルト値を使用している場合、TCP 代行受信の統計情報のコマンドは、次のようになります（わかりやすくするために改行されています）。

```
###Flex-config Appended CLI ###
```

```
threat-detection statistics tcp-intercept rate-interval 30
       burst-rate 400 average-rate 200
```

ステップ 5 これで、影響を受けるデバイスに変更を展開できます。

ステップ 6 次のコマンドを使用して、デバイスの CLI から TCP 代行受信の統計情報をモニターします。

- **show threat-detection statistics top tcp-intercept** [**all** | **detail**]：攻撃を受けて保護された上位 10 のサーバーを表示します。**all** キーワードは、トレースされているすべてのサーバーの履歴データを表示します。**detail** キーワードは、履歴サンプリング データを表示します。システムはレート間隔の間に攻撃の数を 30 回サンプリングするため、デフォルトの 30 分間隔の場合、60 秒ごとに統計情報が収集されます。

  （注）
  shun コマンドを使用して、ホスト IP アドレスへの攻撃をブロックできます。ブロックを削除するには、**no shun** コマンドを使用します。

- **clear threat-detection statistics tcp-intercept**TCP 代行受信の統計情報を削除します。

例：

```
hostname(config)# show threat-detection statistics top tcp-intercept
Top 10 protected servers under attack (sorted by average rate)
Monitoring window size: 30 mins    Sampling interval: 30 secs
<Rank> <Server IP:Port> <Interface> <Ave Rate> <Cur Rate> <Total> <Source IP (Last Attack
 Time)>
--------------------------------------------------------------------------------
1    10.1.1.5:80 inside 1249 9503 2249245 <various> Last: 10.0.0.3 (0 secs ago)
2    10.1.1.6:80 inside 10 10 6080 10.0.0.200 (0 secs ago)
```

Firewall Threat Defense デバイスをトレースルートに表示する

デフォルトでは、Firewall Threat Defense デバイスは、トレースルートにホップとして表示されません。表示されるようにするには、デバイスを通過するパケットの存続可能時間を減らし、ICMP 到達不能メッセージのレート制限を増やす必要があります。これを行うには、サービスポリシー ルールを設定し、ICMP プラットフォーム設定ポリシーを調整する必要があります。

（注） 存続可能時間を減らすと、TTL が 1 のパケットはドロップされますが、接続に TTL がもっと長いパケットが含まれている可能性があるという仮定の下に、セッションに対して接続が開かれます。OSPF hello パケットなどの一部のパケットは TTL が 1 で送信されるため、存続可能時間を減らすと予期しない結果が生じることがある点に注意してください。トラフィック クラスを定義する際には、これらの考慮事項に注意してください。

手順

ステップ 1 Traceroute レポートを有効にするトラフィック クラスを定義する拡張 ACL を作成します。

たとえば、OSPF トラフィックを除く、すべてのアドレスのトラフィック クラスを定義するには、次の手順を実行します。

a) **[オブジェクト（Objects）]** > **[オブジェクト管理（Object Management）]** を選択します。
b) 目次から **[アクセス リスト（Access List）]** > **[拡張（Extended）]** を選択します。
c) [拡張アクセスリストを追加（Add Extended Access List）] をクリックします。
d) オブジェクトの [名前（Name）]（traceroute-enabled など）を入力します。
e) [追加（Add）] をクリックして、OSPF を除外するルールを追加します。
f) アクションを [ブロック（Block）] に変更し、[ポート（Port）] をクリックします。[宛先ポート（Destination Ports）] リストの下でプロトコルとして [OSPFIGP (89)] を選択し、[追加（Add）] をクリックして、プロトコルを選択済みリストに追加します。
g) [拡張アクセス リスト エントリ（Extended Access List Entry）] ダイアログボックスで [追加（Add）] をクリックして、OSPF ルールを ACL に追加します。
h) [追加（Add）] をクリックして、その他すべての接続を含めるルールを追加します。
i) アクションは [許可（Allow）] のままにして、[送信元（Source）] と [宛先（Destination）] リストの両方を空にします。
j) [拡張アクセス リスト エントリ（Extended Access List Entry）] ダイアログボックスで [追加（Add）] をクリックして、ルールを ACL に追加します。

   OSPF 拒否ルールが [すべて許可（Allow Any）] ルールの上にあることを確認します。必要に応じて、ルールをドラッグアンドドロップして移動します。

k) [拡張アクセス リスト オブジェクト（Extended Access List Object）] ダイアログボックスで [保存（Save）] をクリックして、ACL オブジェクトを保存します。

ステップ 2 存続可能時間の値をデクリメントするサービス ポリシールールを設定します。

たとえば、存続可能時間をグローバルにデクリメントするには、次の手順を実行します。

a) **[ポリシー（Policies）]** > **[アクセス制御（Access Control）]** を選択して、このサービスを必要とするデバイスに割り当てられているポリシーを編集します。
b) パケットフロー行の最後にある [詳細（More）] ドロップダウン矢印から [詳細設定（Advanced Settings）] をクリックし、[Threat Defenseサービスポリシー（Threat Defense Service Policy）] の **Edit** (✎) をクリックします。[
c) [ルールの追加（Add Rule）] をクリックします。
d) [グローバルに適用（Apply Globally）] を選択して、[次へ（Next）] をクリックします。
e) このルールに対して作成した拡張 ACL オブジェクトを選択して、[次へ（Next）] をクリックします。
f) [デクリメント TTL の有効化（Enable Decrement TTL）] を選択します。
g) （オプション）その他の接続オプションを必要に応じて調節します。

h) [終了（Finish）]をクリックしてルールを追加します。必要に応じて、ルールをサービスポリシー内の必要な位置にドラッグアンドドロップします。

i) [OK] をクリックして、サービス ポリシーに加えた変更を保存します。

j) [詳細（Advanced）]で[保存（Save）]をクリックして、アクセス コントロール ポリシーに加えた変更を保存します。

これで、影響を受けるデバイスに変更を展開できます。

ステップ 3 ICMP 到達不能メッセージのレート制限を増やします。

a) **[デバイス（Devices）] > [プラットフォーム設定（Platform Settings）]** を選択します。

b) ポリシーをすでにデバイスに割り当てている場合は、そのポリシーを編集します。割り当てていない場合は、新しい Threat Defense プラットフォーム設定ポリシーを作成して、影響を受けるデバイスに割り当てます。

c) 目次から [ICMP] を選択します。

d) [レート制限（Rate Limit）] を（50 などに）増やします。レート制限内で十分な数の応答が生成されるように、[バーストサイズ（Burst Size）] を 10 などに増やすこともできます。

ICMPルールテーブルは、このタスクには無関係なので、空のままにすることができます。

e) [保存（Save）] をクリックします。

ステップ 4 これで、影響を受けるデバイスに変更を展開できます。

サービス ポリシーのモニタリング

デバイスのCLIを使用してサービスポリシー関連の情報をモニターできます。次に、便利なコマンドをいくつか示します。

- **show conn** [**detail**]

  接続情報を表示します。詳細情報は、フラグを使用して特別な接続の特性を示します。たとえば、「b」フラグは、TCPステート バイパスの対象であるトラフィックを示します。

  detail キーワードを使用すると、デッド接続検出（DCD）プローブの情報が表示されます。この情報は、発信側と応答側で接続がプローブされた頻度を示します。たとえば、DCD対応接続の接続詳細は次のようになります。

```
TCP dmz: 10.5.4.11/5555 inside: 10.5.4.10/40299,
    flags UO , idle 1s, uptime 32m10s, timeout 1m0s, bytes 11828,
cluster sent/rcvd bytes 0/0, owners (0,255)
  Traffic received at interface dmz
        Locally received: 0 (0 byte/s)
  Traffic received at interface inside
        Locally received: 11828 (6 byte/s)
  Initiator: 10.5.4.10, Responder: 10.5.4.11
  DCD probes sent: Initiator 5, Responder 5
```

- **show service-policy**

Dead Connection Detection（DCD; デッド接続検出）の統計情報を含むサービス ポリシーの統計情報を表示します。

- **show threat-detection statistics top tcp-intercept** [**all** | **detail**]

  攻撃を受けて保護された上位 10 サーバーを表示します。**all** キーワードは、トレースされているすべてのサーバーの履歴データを表示します。**detail** キーワードは、履歴サンプリング データを表示します。システムはレート間隔の間に攻撃の数を 30 回サンプリングするため、デフォルトの 30 分間隔の場合、60 秒ごとに統計情報が収集されます。

Threat Defense サービスポリシーの履歴

特長	Minimum Firewall Management Center	Minimum Firewall Threat Defense	説明
Threat Defense サービスポリシー	6.3	任意（Any）	Threat Defense サービスポリシーをアクセス コントロール ポリシーの高度なオプションの一部として設定できるようになりました。Threat Defense サービスポリシーを使用して、特定のトラフィッククラスにサービスを適用できます。サポートされている機能には、TCP ステートバイパス、TCP シーケンス番号のランダム化、パケットでの存続可能時間（TTL）の値の減分、デッド接続検出、トラフィック クラスごとおよびクライアントごとの接続および初期接続の最大数の制限の設定、初期接続、ハーフクローズ接続、およびアイドル接続のタイムアウトなどがあります。 新規画面：**[ポリシー（Policies）]**>**[アクセス制御（Access Control）]**>**[アクセス制御（Access Control）]**、[詳細（Advanced）] タブ、[Threat Defenseサービスポリシー（Threat Defense Service Policy）]。 サポートされているプラットフォーム： Secure Firewall Threat Defense
デッド接続検出（DCD）の発信側および応答側の情報、およびクラスタ内の DCD のサポート。	6.5	任意（Any）	デッド接続検出（DCD）を有効にした場合は、**show conn detail** コマンドを使用して発信側と応答側に関する情報を取得できます。デッド接続検出を使用すると、非アクティブな接続を維持できます。**show conn** の出力は、エンドポイントがプローブされた頻度が示されます。さらに、DCD がクラスタでサポートされるようになりました。 新しい/変更されたコマンド：**show conn**（出力のみ） サポートされているプラットフォーム： Secure Firewall Threat Defense

特長	Minimum Firewall Management Center	Minimum Firewall Threat Defense	説明
初期接続の最大セグメントサイズ（MSS）を設定します。	7.1	任意（Any）	サービスポリシーを設定して、初期接続制限に達したときに初期接続の SYN cookie を生成するためのサーバーの最大セグメントサイズ（MSS）を設定できます。これは、最大初期接続数も設定するサービスポリシーの場合に意味があります。 追加または変更された画面：[Add/Edit Service Policy] ウィザードの [Connection Settings]

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。