CHAPTER 36

モジュラ ポリシー フレームワークを使用したサービス ポリシーの設定

モジュラ ポリシー フレームワークを使用したサービス ポリシーにより、一貫性のある柔軟な方法で ASA の機能を設定できます。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 プロファイルは、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で構成されます。

この章は、次の項で構成されています。

サービス ポリシーに関する情報

この項では、サービス ポリシーの機能について説明します。説明する項目は次のとおりです。

通過トラフィックでサポートされる機能

表 36-1 に、モジュラ ポリシー フレームワークでサポートされる機能を示します。

表 36-1　モジュラ ポリシー フレームワーク

機能	次の各項を参照してください。
アプリケーション インスペクション（複数タイプ）	• 第 46 章「アプリケーション レイヤ プロトコル インスペクションの準備」 • 第 47 章「基本インターネット プロトコルのインスペクションの設定」 • 第 48 章「音声とビデオのプロトコルのインスペクションの設定」 • 第 49 章「データベースとディレクトリのプロトコル インスペクションの設定」 • 第 50 章「管理アプリケーション プロトコルのインスペクションの設定」 • 第 60 章「Cisco クラウド Web セキュリティ用の ASA の設定」
ASA CSC	第 67 章「ASA CSC モジュールの設定」
ASA IPS	第 65 章「ASA IPS モジュールの設定」
ASA CX	第 66 章「ASA CX モジュールの設定」
NetFlow セキュア イベント ロギングのフィルタリング	第 82 章「NetFlow セキュア イベント ロギング（NSEL）の設定」
QoS 入出力ポリシング	第 58 章「QoS の設定」
QoS 標準プライオリティ キュー	第 58 章「QoS の設定」
QoS トラフィック シェーピング、階層型プライオリティ キュー	第 58 章「QoS の設定」
TCP と UDP の接続制限値とタイムアウト、および TCP シーケンス番号のランダム化	第 57 章「接続の設定」
TCP の正規化	第 57 章「接続の設定」
TCP ステート バイパス	第 57 章「接続の設定」

管理トラフィックでサポートされる機能

モジュラ ポリシー フレームワークでは、管理トラフィック用に次の機能をサポートします。

- RADIUS アカウンティング トラフィックのアプリケーション インスペクション（第 50 章「管理アプリケーション プロトコルのインスペクションの設定」を参照）
- 接続の制限値（第 57 章「接続の設定」を参照）

機能の方向

アクションは、機能に応じて双方向または単方向にトラフィックに適用されます。双方向に適用される機能の場合、トラフィックが両方向のクラス マップと一致した場合に、ポリシー マップを適用するインターフェイスを出入りするすべてのトラフィックが影響を受けます。

（注）　グローバル ポリシーを使用する場合は、すべての機能が単方向です。単一インターフェイスに適用する場合に通常双方向の機能は、グローバルに適用される場合、各インターフェイスの入力にのみ適用されます。ポリシーはすべてのインターフェイスに適用されるため、ポリシーは両方向に適用され、この場合の双方向は冗長になります。

QoS プライオリティ キューなど単方向に適用される機能の場合は、ポリシー マップを適用するインターフェイスに出入りする（機能によって異なります）トラフィックだけが影響を受けます。各機能の方向については、表 36-2 を参照してください。

表 36-2　機能の方向

機能	単一インターフェイスでの方向	グローバルでの方向
アプリケーション インスペクション（複数タイプ）	双方向	入力
ASA CSC	双方向	入力
ASA CX	双方向	入力
ASA CX 認証プロキシ	入力	入力
ASA IPS	双方向	入力
NetFlow セキュア イベント ロギングのフィルタリング	該当なし	入力
QoS 入力ポリシング	入力	入力
QoS 出力ポリシング	出力	出力
QoS 標準プライオリティ キュー	出力	出力
QoS トラフィック シェーピング、階層型プライオリティ キュー	出力	出力
TCP と UDP の接続制限値とタイムアウト、および TCP シーケンス番号のランダム化	双方向	入力
TCP の正規化	双方向	入力
TCP ステート バイパス	双方向	入力

サービス ポリシー内の機能照合

パケットが所定のインターフェイスのポリシーマップ内のクラス マップと照合される方法については、次の情報を参照してください。

1. パケットは、各機能タイプのポリシー マップで、1 つのクラス マップにだけ一致します。
2. パケットが機能タイプのクラス マップに一致した場合、ASA は、その機能タイプの後続のクラス マップとは照合しません。

3. ただし、パケットが別の機能タイプの後続のクラス マップと一致した場合、ASA は、後続のクラス マップのアクションも適用します（サポートされている場合）。サポートされていない組み合わせの詳細については、「特定の機能アクションの非互換性」（P.36-5）を参照してください。

（注）　アプリケーション インスペクションには、複数のインスペクション タイプが含まれ、ほとんどのタイプは相互に排他的です。組み合わせ可能なインスペクションの場合、各インスペクションは個々の機能と見なされます。

たとえば、パケットが接続制限値のクラス マップと一致し、アプリケーション インスペクションのクラス マップとも一致した場合、両方のクラス マップ アクションが適用されます。

パケットが HTTP インスペクションで 1 つのクラス マップと一致し、HTTP インスペクションを含む別のクラス マップとも一致した場合、2 番目のクラス マップのアクションは適用されません。

パケットが FTP インスペクションで 1 つのクラス マップと一致し、HTTP インスペクションを含む別のクラス マップとも一致した場合、HTTP および FTP インスペクションは組み合わせることができないため、2 番目のクラス マップのアクションは適用されません。

パケットが HTTP インスペクションで 1 つのクラス マップと一致し、さらに IPv6 インスペクションを含む別のクラス マップとも一致した場合、IPv6 インスペクションは他のとタイプのインスペクションと組み合わせることができるため、両方のアクションが適用されます。

複数の機能アクションが適用される順序

ポリシー マップの各種のアクションが実行される順序は、ポリシー マップ中に出現する順序とは無関係です。

（注）　NetFlow セキュア イベント ロギングのフィルタリングは、順序に関係なく実行されます。

アクションは次の順序で実行されます。

1. QoS 入力ポリシング
2. TCP の正規化、TCP と UDP の接続制限値とタイムアウト、TCP シーケンス番号のランダム化、および TCP ステート バイパス

（注）　ASA がプロキシ サービス（AAA や CSC など）を実行したり、TCP ペイロード（FTP インスペクション）を変更したりするときは、TCP ノーマライザはデュアル モードで動作します。その場合、サービスを変更するプロキシやペイロードの前後で適用されます。

3. ASA CSC
4. 他のインスペクションと組み合わせることができるアプリケーション インスペクション：
   a. IPv6
   b. IP オプション
   c. WAAS
5. 他のインスペクションと組み合わせることができないアプリケーション インスペクション：詳細については、「特定の機能アクションの非互換性」（P.36-5）を参照してください。
6. ASA IPS
7. ASA CX

8. QoS 出力ポリシング
9. QoS 標準プライオリティ キュー
10. QoS トラフィック シェーピング、階層型プライオリティ キュー

特定の機能アクションの非互換性

一部の機能は同じトラフィックに対して相互に互換性がありません。次のリストには、すべての非互換性が含まれていない場合があります。各機能の互換性については、機能に関する章または項を参照してください。

- QoS プライオリティ キューイングと QoS ポリシングは同じトラフィックの集合に対して設定できません。
- ほとんどのインスペクションは別のインスペクションと組み合わせられないため、同じトラフィックに複数のインスペクションを設定しても、ASA は 1 つのインスペクションだけを適用します。HTTP インスペクションはクラウド Web セキュリティ インスペクションと組み合わせることができます。他の例外は、「複数の機能アクションが適用される順序」（P.36-4）に記載されています。
- トラフィックを ASA CX および ASA IPS などの複数のモジュールに送信されるように設定できません。
- HTTP インスペクションは、ASA CX と互換性がありません。
- ASA CX はクラウド Web セキュリティと互換性がありません。

（注）　デフォルト グローバル ポリシーで使用される **match default-inspection-traffic** コマンドは、デフォルト ポートをすべてのインスペクションと照合する特別な CLI ショートカットです。ポリシー マップで使用すると、このクラス マップでは、トラフィックの宛先ポートに基づいて、各パケットに正しいインスペクションが適用されます。たとえば、宛先がポート 69 の UDP トラフィックが ASA に到達すると、ASA は TFTP インスペクションを適用し、宛先がポート 21 の TCP トラフィックが到着すると、ASA は FTP インスペクションを適用します。そのため、この場合に限って同じクラス マップに複数のインスペクションを設定できます。通常、ASA は、ポート番号を使用して適用するインスペクションを決定しないため、標準以外のポートなどにも柔軟にインスペクションを適用できます。

このトラフィック クラスには、クラウド Web セキュリティ インスペクション用のデフォルト ポートは含まれません（80 および 443）。

誤った設定例は、同じポリシー マップに複数のインスペクションを設定しても、default-inspection-traffic ショートカットを使用しないことです。例 36-1 では、ポート 21 宛てのトラフィックが、FTP インスペクションと HTTP インスペクションの両方に誤って設定されています。例 36-2 では、ポート 80 宛てのトラフィックが、FTP インスペクションと HTTP インスペクションの両方に誤って設定されています。どちらの誤った設定例の場合も、FTP インスペクションだけが適用されています。これは、適用されたインスペクションの順序では、FTP が HTTP よりも先になるためです。

例 36-1　　FTP パケットの誤設定（HTTP インスペクションも設定されている）

```
class-map ftp
   match port tcp eq 21
class-map http
   match port tcp eq 21 [80 の誤り ]
policy-map test
   class ftp
     inspect ftp
```

```
   class http
     inspect http
```

例 36-2　　HTTP パケットの誤設定（FTP インスペクションも設定されている）

```
class-map ftp
   match port tcp eq 80 [21 の誤り ]
class-map http
   match port tcp eq 80
policy-map test
   class http
     inspect http
   class ftp
     inspect ftp
```

複数のサービス ポリシーの場合の機能照合

TCP および UDP トラフィック（およびステートフル ICMP インスペクションがイネーブルの場合は ICMP）の場合、サービス ポリシーはトラフィック フローに対して作用し、個々のパケットに限定されません。トラフィックが、1 つのインターフェイスのポリシーで定義されている機能に一致する既存の接続の一部である場合、そのトラフィック フローを別のインターフェイスのポリシーにある同じ機能と照合することはできません。最初のポリシーのみが使用されます。

たとえば、HTTP トラフィックが、HTTP トラフィックを検査する内部インターフェイスのポリシーと一致するときに、HTTP インスペクション用の外部インターフェイスに別のポリシーがある場合、そのトラフィックが外部インターフェイスの出力側でも検査されることはありません。同様に、その接続のリターン トラフィックが外部インターフェイスの入力ポリシーによって検査されたり、内部インターフェイスの出力ポリシーによって検査されたりすることもありません。

ステートフル ICMP インスペクションをイネーブルにしない場合の ICMP のように、フローとして扱われないトラフィックの場合は、リターン トラフィックを戻り側のインターフェイスの別のポリシーマップと照合できます。たとえば、内部および外部のインターフェイスで IPS を設定するとき、内部ポリシーでは仮想センサー 1 を使用するのに対して、外部ポリシーでは仮想センサー 2 を使用する場合、非ステートフル ping は仮想センサー 1 の発信側を照合するだけでなく、仮想センサー 2 の着信側も照合します。

サービス ポリシーのライセンス要件

モデル	ライセンス要件
すべてのモデル	基本ライセンス

ガイドラインと制限事項

この項では、この機能のガイドラインと制限事項について説明します。

コンテキスト モードのガイドライン

シングル コンテキスト モードとマルチ コンテキスト モードでサポートされています。

ファイアウォール モードのガイドライン

ルーテッド ファイアウォール モードとトランスペアレント ファイアウォール モードでサポートされています。

IPv6 のガイドライン

IPv6 は次の機能でサポートされています。

- DNS、FTP、HTTP、ICMP、ScanSafe、SIP、SMTP、IPsec-pass-thru、および IPv6 のアプリケーション インスペクション。
- ASA IPS
- ASA CX
- NetFlow セキュア イベント ロギングのフィルタリング
- TCP と UDP の接続制限値とタイムアウト、および TCP シーケンス番号のランダム化
- TCP の正規化
- TCP ステート バイパス

クラス マップのガイドライン

すべてのタイプのクラス マップの最大数は、シングル モードで 255、またはマルチ モードのコンテキストごとに 255 です。クラス マップには、次のタイプがあります。

- レイヤ 3/4 クラスマップ（通過トラフィックと管理トラフィック向け）。
- インスペクション クラス マップ
- 正規表現クラス マップ
- インスペクション ポリシー マップ下で直接使用される **match** コマンド

この制限には、すべてのタイプのデフォルト クラス マップも含まれ、ユーザ設定のクラス マップを約 235 に制限します。「デフォルトのクラス マップ」（P.36-9）を参照してください。

ポリシー マップのガイドライン

ポリシー マップを使用する場合は、次のガイドラインを参考にしてください。

- 各インターフェイスには、ポリシー マップを 1 つだけ割り当てることができます （ただし、設定では最大 64 のポリシー マップを作成できます）。
- 同一のポリシー マップを複数のインターフェイスに適用できます。
- 1 つのレイヤ 3/4 ポリシー マップで最大 63 のレイヤ 3/4 クラス マップを識別できます。
- クラス マップごとに、1 つ以上の機能タイプから複数のアクションを割り当てることができます（サポートされている場合）。「特定の機能アクションの非互換性」（P.36-5）を参照してください。

サービス ポリシーのガイドライン

- インターフェイス サービス ポリシーは、特定の機能に対するグローバル サービス ポリシーより優先されます。たとえば、FTP インスペクションのグローバル ポリシーと、TCP 正規化のインターフェイス ポリシーがある場合、FTP インスペクションと TCP 正規化の両方がインターフェイスに適用されます。これに対し、FTP インスペクションのグローバル ポリシーと、FTP インスペクションのインターフェイス ポリシーがある場合は、インターフェイス ポリシーの FTP インスペクションだけがインターフェイスに適用されます。
- 適用できるグローバル ポリシーは 1 つだけです。たとえば、機能セット 1 が含まれたグローバル ポリシーと、機能セット 2 が含まれた別のグローバル ポリシーを作成できません。すべての機能は 1 つのポリシーに含める必要があります。

デフォルト設定

モジュラ ポリシー フレームワークのデフォルト設定については、次の項目で説明します。

- 「デフォルト コンフィギュレーション」（P.36-8）
- 「デフォルトのクラス マップ」（P.36-9）

デフォルト コンフィギュレーション

デフォルトでは、すべてのデフォルト アプリケーション インスペクション トラフィックに一致するポリシーがコンフィギュレーションに含まれ、特定のインスペクションがすべてのインターフェイスのトラフィックに適用されます（グローバル ポリシー）。すべてのインスペクションがデフォルトでイネーブルになっているわけではありません。適用できるグローバル ポリシーは 1 つだけなので、グローバル ポリシーを変更する場合は、デフォルトのポリシーを編集するか、デフォルトのポリシーをディセーブルにして新しいポリシーを適用します。（特定の機能では、グローバル ポリシーはインターフェイス ポリシーより優先されます）。

デフォルト ポリシーには、次のアプリケーション インスペクションが含まれます。

- メッセージの最大長 512 バイトに対する DNS インスペクション
- FTP
- H323（H225）
- H323（RAS）
- RSH
- RTSP
- ESMTP
- SQLnet
- Skinny（SCCP）
- SunRPC
- XDMCP
- SIP
- NetBios
- TFTP
- IP オプション

デフォルト ポリシー コンフィギュレーションには、次のコマンドが含まれます。

```
class-map inspection_default
 match default-inspection-traffic
policy-map type inspect dns preset_dns_map
 parameters
    message-length maximum 512
policy-map global_policy
 class inspection_default
  inspect dns preset_dns_map
  inspect ftp
  inspect h323 h225
  inspect h323 ras
  inspect rsh
  inspect rtsp
```

```
  inspect esmtp
  inspect sqlnet
  inspect skinny
  inspect sunrpc
  inspect xdmcp
  inspect sip
  inspect netbios
  inspect tftp
  inspect ip-options
service-policy global_policy global
```

（注）　デフォルトのクラス マップで使用される特別な **match default-inspection-traffic** コマンドの詳細については、「特定の機能アクションの非互換性」（P.36-5）を参照してください。

デフォルトのクラス マップ

コンフィギュレーションには、デフォルト グローバル ポリシーで ASA が使用するデフォルトのレイヤ 3/4 クラス マップが含まれます。これは default-inspection-traffic と呼ばれ、デフォルト インスペクション トラフィックと一致します。デフォルト グローバル ポリシーで使用されるこのクラスは、デフォルト ポートをすべてのインスペクションと照合する特別なショートカットです。ポリシーで使用すると、このクラスでは、トラフィックの宛先ポートに基づいて、各パケットに正しいインスペクションが適用されます。たとえば、宛先がポート 69 の UDP トラフィックが ASA に到達すると、ASA は TFTP インスペクションを適用し、宛先がポート 21 の TCP トラフィックが到着すると、ASA は FTP インスペクションを適用します。そのため、この場合に限って同じクラス マップに複数のインスペクションを設定できます。通常、ASA は、ポート番号を使用して適用するインスペクションを決定しないため、標準以外のポートなどにも柔軟にインスペクションを適用できます。

```
class-map inspection_default
 match default-inspection-traffic
```

デフォルト コンフィギュレーションにある別のクラス マップは、class-default と呼ばれ、すべてのトラフィックと一致します。このクラス マップは、すべてのレイヤ 3/4 ポリシー マップの最後に表示され、原則的に、他のすべてのトラフィックでどんなアクションも実行しないように ASA に通知します。必要であれば、独自の **match any** クラス マップを作成せずに、class-default クラスを使用できます。実際のところ、class-default で使用可能な機能は、QoS トラフィック シェーピングなどの一部の機能だけです。

```
class-map class-default
 match any
```

サービス ポリシーを設定するためのタスク フロー

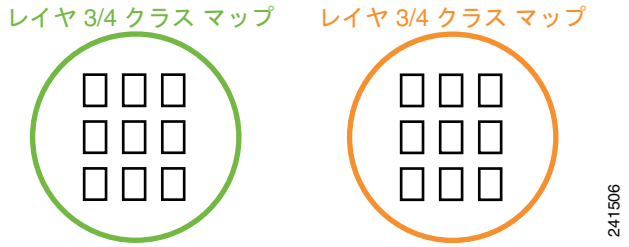
この項は、次の内容で構成されています。

モジュラ ポリシー フレームワークを使用するためのタスク フロー

モジュラ ポリシー フレームワークを設定するには、次の手順を実行します。

ステップ 1　トラフィックの特定：レイヤ 3/4 クラス マップを作成して、モジュラ ポリシー フレームワーク アクションを実行するトラフィックを特定します。

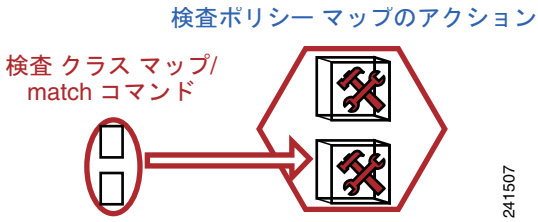
たとえば、ASA を通過するすべてのトラフィックでアクションを実行したり、10.1.1.0/24 から任意の宛先アドレスまでのトラフィックで特定のアクションだけを実行したりできます。



「トラフィックの特定（レイヤ 3/4 クラス マップ）」（P.36-12）を参照してください。

ステップ 2　インスペクション トラフィックでの追加のアクションの実行：実行するアクションの 1 つがアプリケーション インスペクションで、インスペクション トラフィックで追加アクションを実行する場合は、インスペクション ポリシー マップを作成します。インスペクション ポリシー マップはトラフィックを特定し、そのトラフィックで何をするかを指定します。

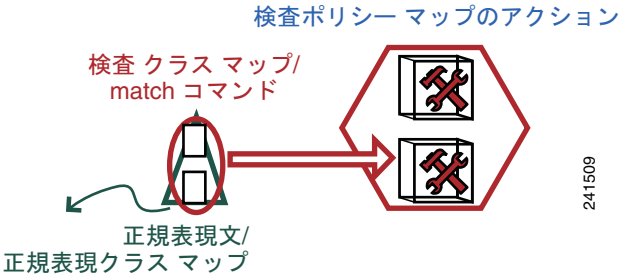
たとえば、本文の長さが 1000 バイトを上回るすべての HTTP 要求をドロップできます。



match コマンドでトラフィックを直接特定する独立したインスペクション ポリシー マップを作成したり、再利用のために、またはより複雑な照合のためにインスペクション クラス マップを作成したりできます。「インスペクション ポリシー マップのアクションの定義」（P.37-3）および「インスペクション クラス マップ内のトラフィックの特定」（P.37-6）を参照してください。

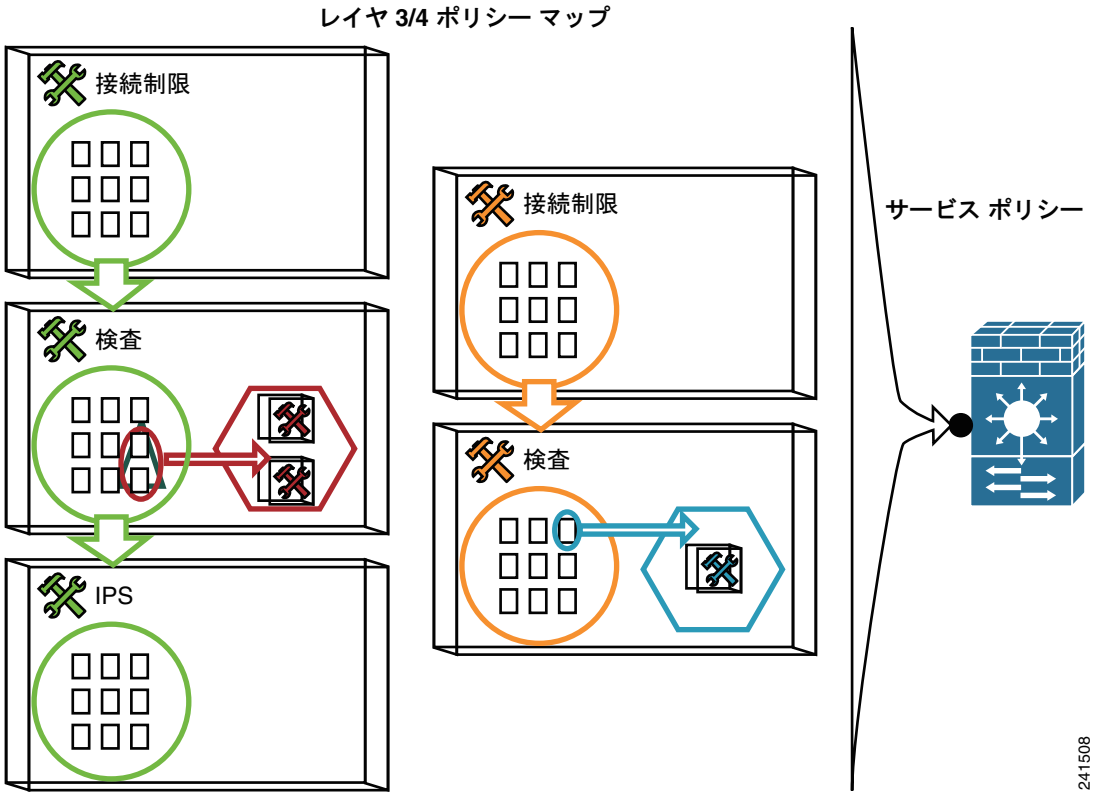
ステップ 3　正規表現の作成：検査されたパケット内の正規表現にテキストを照合する場合、正規表現または正規表現のグループ（正規表現クラス マップ）を作成できます。そして、トラフィックがインスペクション ポリシー マップと一致するように定義するときに、既存の正規表現を呼び出すことができます。

たとえば、「example.com」というテキストが含まれた URL を持つすべての HTTP 要求をドロップできます。



「正規表現の作成」（P.18-15）および「正規表現クラス マップの作成」（P.18-17）を参照してください。

ステップ 4　レイヤ 3/4 ポリシー マップを作成して、各レイヤ 3/4 クラス マップに実行するアクションを定義します。次に、サービス ポリシーを使用して、ポリシー マップを適用するインターフェイスを決定します。



「アクションの定義（レイヤ 3/4 ポリシー マップ）」（P.36-15）および「インターフェイスへのアクションの適用（サービス ポリシー）」（P.36-17）を参照してください。

QoS トラフィック シェーピング用の階層型ポリシー マップを設定するためのタスク フロー

クラス マップで QoS トラフィック シェーピングをイネーブルにすると、オプションでシェーピングされるトラフィックのサブセットに対するプライオリティ キューイングをイネーブルにできます。これを行うには、プライオリティ キューイングのポリシー マップを作成する必要があります。そうすると、トラフィック シェーピングのポリシー マップ内で、プライオリティ クラス マップを呼び出すことができます。インターフェイスに適用できるのは、トラフィック シェーピングのクラス マップだけです。

この機能の詳細については、第 58 章「QoS に関する情報」を参照してください。

階層型ポリシー マップは、トラフィック シェーピングとプライオリティ キューイングだけでサポートされます。

階層型ポリシー マップを実装するには、次の手順を実行します。

ステップ 1　「トラフィックの特定（レイヤ 3/4 クラス マップ）」（P.36-12）の説明に従って、優先トラフィックを特定します。

階層型ポリシー マップで使用される複数のクラス マップを作成できます。

ステップ 2　「アクションの定義（レイヤ 3/4 ポリシー マップ）」（P.36-15）の説明に従って、ポリシー マップを作成し、各クラス マップに対する単独のアクションを **priority** として指定します。

ステップ 3　「アクションの定義（レイヤ 3/4 ポリシー マップ）」（P.36-15）の説明に従って、別のポリシー マップを作成し、**class-default** クラス マップに **shape** アクションを指定します。

トラフィック シェーピングは **class-default** クラス マップだけに適用できます。

ステップ 4　同じクラス マップでは、ステップ 2 で作成したプライオリティ ポリシー マップを **service-policy** *priority_policy_map* コマンドを使用して特定します。

ステップ 5　「インターフェイスへのアクションの適用（サービス ポリシー）」（P.36-17）の説明に従って、シェーピングのポリシー マップをインターフェイスに適用します。

トラフィックの特定（レイヤ 3/4 クラス マップ）

レイヤ 3/4 クラス マップにより、アクションを適用するレイヤ 3 および 4 のトラフィックを特定します。1 つのレイヤ 3/4 ポリシー マップに複数のレイヤ 3/4 クラス マップを作成できます。

この項は、次の内容で構成されています。

- 「通過トラフィック用のレイヤ 3/4 クラス マップの作成」（P.36-12）
- 「管理トラフィック用のレイヤ 3/4 クラス マップの作成」（P.36-15）

通過トラフィック用のレイヤ 3/4 クラス マップの作成

レイヤ 3/4 クラス マップでは、プロトコル、ポート、IP アドレス、およびレイヤ 3 またはレイヤ 4 の他の属性に基づいてトラフィックを照合します。

手順の詳細

	コマンド	目的
ステップ 1	**class-map** *class_map_name* **例：** `hostname(config)# class-map all_udp`	*class_map_name* が最大 40 文字の文字列であるレイヤ 3/4 クラス マップを作成します。「class-default」という名前は予約されています。すべてのタイプのクラス マップで同じ名前スペースが使用されるため、別のタイプのクラス マップですでに使用されている名前は再度使用できません。CLI はクラスマップ コンフィギュレーション モードに移行します。
ステップ 2	（任意） **description** *string* **例：** `hostname(config-cmap)# description All UDP traffic`	クラス マップに説明を追加します。
ステップ 3	次のいずれかを使用するトラフィックの照合	特に指定がない場合、クラス マップに含めることができる **match** コマンドは 1 つだけです。

コマンド	目的
match any **例：** hostname(config-cmap)# match any	すべてのトラフィックと照合します。
match access-list *access_list_name* **例：** hostname(config-cmap)# match access-list udp	拡張アクセス リストで指定されたトラフィックと照合します。ASA が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は、EtherType アクセス リストを使用できます。
match port {**tcp** | **udp**} {**eq** *port_num* | **range** *port_num port_num*} **例：** hostname(config-cmap)# match tcp eq 80	TCP または UDP の宛先ポート（1 つのポートまたは連続する一定範囲のポート）と照合します。 **ヒント** 複数の非連続ポートを使用するアプリケーションに対しては、**match access-list** コマンドを使用して、各ポートと一致する ACE を定義します。
match default-inspection-traffic **例：** hostname(config-cmap)# match default-inspection-traffic	インスペクション用のデフォルト トラフィックと照合します（ASA が検査できるすべてのアプリケーションで使用されるデフォルトの TCP および UDP ポート）。 デフォルト グローバル ポリシーで使用されるこのコマンドは、ポリシー マップで使用されると、トラフィックの宛先ポートに基づいて各パケットに正しいインスペクションを適用する特別な CLI ショートカットです。たとえば、宛先がポート 69 の UDP トラフィックが ASA に到達すると、ASA は TFTP インスペクションを適用し、宛先がポート 21 の TCP トラフィックが到着すると、ASA は FTP インスペクションを適用します。そのため、この場合に限って同じクラス マップに複数のインスペクションを設定できます（他のインスペクションとともに設定可能な WAAS インスペクションを除きます。アクションの組み合わせの詳細については、「特定の機能アクションの非互換性」（P.36-5）を参照してください）。通常、ASA は、ポート番号を使用して適用するインスペクションを決定しないため、標準以外のポートなどにも柔軟にインスペクションを適用できます。 デフォルト ポートのリストについては、「デフォルト設定」（P.46-4）を参照してください。**match default-inspection-traffic** コマンドにポートが含まれているすべてのアプリケーションが、ポリシー マップでデフォルトでイネーブルになっているわけではありません。 **match access-list** コマンドを **match default-inspection-traffic** コマンドとともに指定すると、一致するトラフィックを絞り込むことができます。**match default-inspection-traffic** コマンドによって照合するポートとプロトコルが指定されるため、アクセス リストのポートとプロトコルはすべて無視されます。 **ヒント** トラフィック インスペクションは、アプリケーション トラフィックが発生するポートだけで行うことをお勧めします。**match any** などを使用してすべてのトラフィックを検査すると、ASA のパフォーマンスに影響が出る場合があります。

コマンド	目的
match dscp *value1* [*value2*] [...] [*value8*] **例：** hostname(config-cmap)# match dscp af43 cs1 ef	IP ヘッダーの DSCP 値（最大 8 個の DSCP 値）と照合します。
match precedence *value1* [*value2*] [*value3*] [*value4*] **例：** hostname(config-cmap)# match precedence 1 4	IP ヘッダーの TOS 値で表される最大 4 個の優先値と照合します。*value1* ～ *value4* は 0 ～ 7 になります。この値は該当の優先順位に対応します。
match rtp *starting_port range* **例：** hostname(config-cmap)# match rtp 4004 100	RTP トラフィックと照合します。*starting_port* は 2000 ～ 65534 の偶数番号の UDP 宛先ポートを指定します。*range* には、*starting_port* よりも上の追加 UDP ポートの数を 0 ～ 16383 で指定します。
match tunnel-group *name* （任意） **match flow ip destination-address** **例：** hostname(config-cmap)# match tunnel-group group1 hostname(config-cmap)# match flow ip destination-address	QoS を適用する VPN トンネル グループのトラフィックを照合します。 トラフィック照合を調整するために、**match** コマンドをもう 1 つ指定できます。上記のコマンドのいずれかを指定できますが、**match any**、**match access-list**、および **match default-inspection-traffic** コマンドは指定できません。または、**match flow ip destination-address** コマンドを入力して、各 IP アドレス宛てのトンネル グループのフローを照合することもできます。

例

次に、**class-map** コマンドの例を示します。

```
hostname(config)# access-list udp permit udp any any
hostname(config)# access-list tcp permit tcp any any
hostname(config)# access-list host_foo permit ip any 10.1.1.1 255.255.255.255

hostname(config)# class-map all_udp
hostname(config-cmap)# description "This class-map matches all UDP traffic"
hostname(config-cmap)# match access-list udp

hostname(config-cmap)# class-map all_tcp
hostname(config-cmap)# description "This class-map matches all TCP traffic"
hostname(config-cmap)# match access-list tcp

hostname(config-cmap)# class-map all_http
hostname(config-cmap)# description "This class-map matches all HTTP traffic"
hostname(config-cmap)# match port tcp eq http

hostname(config-cmap)# class-map to_server
hostname(config-cmap)# description "This class-map matches all traffic to server 10.1.1.1"
hostname(config-cmap)# match access-list host_foo
```

管理トラフィック用のレイヤ 3/4 クラス マップの作成

ASA への管理トラフィックに対して、この種類のトラフィックに特有のアクションの実行が必要になる場合があります。管理クラス マップを指定して、アクセス リストまたは TCP や UDP のポートと照合できます。ポリシー マップの管理クラス マップで設定可能なアクションのタイプは、管理トラフィック専用です。「管理トラフィックでサポートされる機能」（P.36-2）を参照してください。

手順の詳細

	コマンド	目的
ステップ 1	**class-map type management** *class_map_name* **例：** `hostname(config)# class-map type management all_mgmt`	*class_map_name* が最大 40 文字の文字列である管理クラス マップを作成します。「class-default」という名前は予約されています。すべてのタイプのクラス マップで同じ名前スペースが使用されるため、別のタイプのクラス マップですでに使用されている名前は再度使用できません。CLI はクラスマップ コンフィギュレーション モードに移行します。
ステップ 2	（任意） **description** *string* **例：** `hostname(config-cmap)# description All management traffic`	クラス マップに説明を追加します。
ステップ 3	次のいずれかを使用するトラフィックの照合	特に指定がない場合、クラス マップに含めることができる **match** コマンドは 1 つだけです。
	match access-list *access_list_name* **例：** `hostname(config-cmap)# match access-list udp`	拡張アクセス リストで指定されたトラフィックと照合します。ASA が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は、EtherType アクセス リストを使用できます。
	match port {**tcp** | **udp**} {**eq** *port_num* | **range** *port_num port_num*} **例：** `hostname(config-cmap)# match tcp eq 80`	TCP または UDP の宛先ポート（1 つのポートまたは連続する一定範囲のポート）と照合します。 **ヒント** 複数の非連続ポートを使用するアプリケーションに対しては、**match access-list** コマンドを使用して、各ポートと一致する ACE を定義します。

アクションの定義（レイヤ 3/4 ポリシー マップ）

この項では、レイヤ 3/4 ポリシー マップを作成して、アクションをレイヤ 3/4 クラス マップに関連付ける方法について説明します。

制限事項

ポリシー マップの最大数は 64 ですが、各インターフェイスには、ポリシー マップを 1 つだけ適用できます。

手順の詳細

	コマンド	目的
ステップ 1	**policy-map** *policy_map_name* **例：** `hostname(config)# policy-map global_policy`	ポリシー マップを追加します。*policy_map_name* 引数は、最大 40 文字のポリシー マップ名です。すべてのタイプのポリシー マップで同じ名前スペースが使用されるため、別のタイプのポリシー マップですでに使用されている名前は再度使用できません。CLI はポリシー マップ コンフィギュレーション モードに入ります。
ステップ 2	（任意） **class** *class_map_name* **例：** `hostname(config-pmap)# description global policy map`	設定済みのレイヤ 3/4 クラス マップを指定します。*class_map_name* は、クラス マップの名前です。クラス マップを追加するには、「トラフィックの特定（レイヤ 3/4 クラス マップ）」（P.36-12）を参照してください。 **（注）** クラス マップに **match default-inspection-traffic** コマンドがない場合、そのクラスに最大 1 つの **inspect** コマンドを設定できます。 QoS では、トラフィック シェーピング機能とプライオリティ キュー機能に階層型ポリシー マップを設定できます。詳細については、「QoS トラフィック シェーピング用の階層型ポリシー マップを設定するためのタスク フロー」（P.36-11）を参照してください。
ステップ 3	このクラス マップに、1 つ以上のアクションを指定します。	「通過トラフィックでサポートされる機能」（P.36-2）を参照してください。
ステップ 4	このポリシー マップに含めるクラス マップごとに、ステップ 2 とステップ 3 を繰り返します。	

例

接続ポリシーの **policy-map** コマンドの例を次に示します。このコマンドは、Web サーバ 10.1.1.1 への接続許可数を制限します。

```
hostname(config)# access-list http-server permit tcp any host 10.1.1.1
hostname(config)# class-map http-server
hostname(config-cmap)# match access-list http-server

hostname(config)# policy-map global-policy
hostname(config-pmap)# description This policy map defines a policy concerning connection
to http server.
hostname(config-pmap)# class http-server
hostname(config-pmap-c)# set connection conn-max 256
```

次の例は、ポリシー マップでの複数の照合の動作を示しています。

```
hostname(config)# class-map inspection_default
hostname(config-cmap)# match default-inspection-traffic
hostname(config)# class-map http_traffic
hostname(config-cmap)# match port tcp eq 80

hostname(config)# policy-map outside_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect http http_map
hostname(config-pmap-c)# inspect sip
hostname(config-pmap)# class http_traffic
hostname(config-pmap-c)# set connection timeout idle 0:10:0
```

次の例は、トラフィックが最初の利用可能なクラス マップと一致した場合に、同じ機能ドメインのアクションが指定されている後続のクラス マップと照合されないことを示しています。

```
hostname(config)# class-map telnet_traffic
hostname(config-cmap)# match port tcp eq 23
hostname(config)# class-map ftp_traffic
hostname(config-cmap)# match port tcp eq 21
hostname(config)# class-map tcp_traffic
hostname(config-cmap)# match port tcp range 1 65535
hostname(config)# class-map udp_traffic
hostname(config-cmap)# match port udp range 0 65535
hostname(config)# policy-map global_policy
hostname(config-pmap)# class telnet_traffic
hostname(config-pmap-c)# set connection timeout idle 0:0:0
hostname(config-pmap-c)# set connection conn-max 100
hostname(config-pmap)# class ftp_traffic
hostname(config-pmap-c)# set connection timeout idle 0:5:0
hostname(config-pmap-c)# set connection conn-max 50
hostname(config-pmap)# class tcp_traffic
hostname(config-pmap-c)# set connection timeout idle 2:0:0
hostname(config-pmap-c)# set connection conn-max 2000
```

Telnet 接続は、開始時に **class telnet_traffic** と一致します。同様に FTP 接続は、開始時に **class ftp_traffic** と一致します。Telnet および FTP 以外の TCP 接続の場合は、**class tcp_traffic** と一致します。Telnet 接続または FTP 接続は **class tcp_traffic** と一致しますが、すでに他のクラスと一致しているため、ASA はこの照合を行いません。

インターフェイスへのアクションの適用（サービス ポリシー）

レイヤ 3/4 ポリシー マップをアクティブにするには、1 つ以上のインターフェイスに適用するサービス ポリシー、またはすべてのインターフェイスにグローバルに適用するサービス ポリシーを作成します。

制限事項

適用できるグローバル ポリシーは 1 つだけなので、グローバル ポリシーを変更する場合は、デフォルトのポリシーを編集するか、デフォルトのポリシーをディセーブルにして新しいポリシーを適用します。デフォルトでは、すべてのデフォルト アプリケーション インスペクション トラフィックに一致するグローバル ポリシーがコンフィギュレーションに含まれ、すべてのインスペクションがトラフィックにグローバルに適用されます。デフォルト サービス ポリシーには、次のコマンドが含まれます。

```
service-policy global_policy global
```

手順の詳細

コマンド	目的
service-policy *policy_map_name* **interface** *interface_name* [**fail-close**] **例：** hostname(config)# service-policy inbound_policy interface outside	インターフェイスにポリシー マップを関連付けてサービス ポリシーを作成します。**fail-close** オプションを指定して、IPv6 トラフィックをサポートしないアプリケーション インスペクションによってドロップされた IPv6 トラフィックの syslog（767001）を生成します。デフォルトでは、syslog が生成されません。IPv6 をサポートするインスペクションのリストについては、「IPv6 のガイドライン」（P.36-7）を参照してください。
service-policy *policy_map_name* **global** [**fail-close**] **例：** hostname(config)# service-policy inbound_policy global	特定のポリシーを持たないすべてのインターフェイスに適用するサービス ポリシーを作成します。**fail-close** オプションを指定して、IPv6 トラフィックをサポートしないアプリケーション インスペクションによってドロップされた IPv6 トラフィックの syslog（767001）を生成します。デフォルトでは、syslog が生成されません。IPv6 をサポートするインスペクションのリストについては、「IPv6 のガイドライン」（P.36-7）を参照してください。

例

たとえば、次のコマンドは、外部インターフェイスで inbound_policy ポリシー マップをイネーブルにします。

```
hostname(config)# service-policy inbound_policy interface outside
```

次のコマンドは、デフォルト グローバル ポリシーをディセーブルにし、他のすべての ASA インターフェイスで新しいポリシー new_global_policy をイネーブルにします。

```
hostname(config)# no service-policy global_policy global
hostname(config)# service-policy new_global_policy global
```

モジュラ ポリシー フレームワークのモニタリング

モジュラ ポリシー フレームワークをモニタするには、次のコマンドを入力します。

コマンド	目的
show service-policy	サービス ポリシーの統計情報を表示します。

モジュラ ポリシー フレームワークの設定例

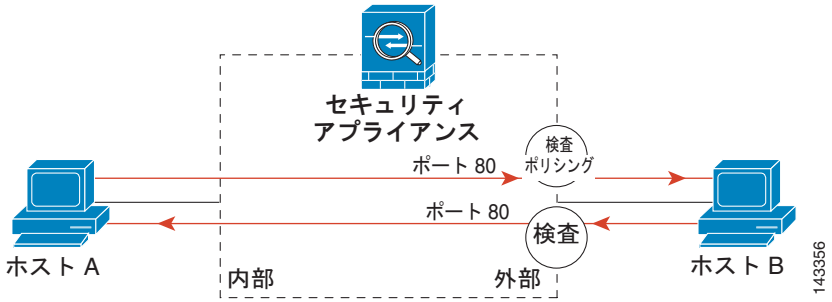
この項では、モジュラ ポリシー フレームワークの例をいくつか示します。次の項目を取り上げます。

- 「HTTP トラフィックへのインスペクションと QoS ポリシングの適用」（P.36-19）
- 「HTTP トラフィックへのインスペクションのグローバルな適用」（P.36-19）
- 「特定のサーバへの HTTP トラフィックに対するインスペクションと接続制限値の適用」（P.36-20）
- 「NAT による HTTP トラフィックへのインスペクションの適用」（P.36-21）

HTTP トラフィックへのインスペクションと QoS ポリシングの適用

この例（図 36-1）では、外部インターフェイスを通過して ASA を出入りするすべての HTTP 接続（ポート 80 の TCP トラフィック）が HTTP インスペクション対象として分類されます。外部インターフェイスを出るすべての HTTP トラフィックがポリシング対象として分類されます。

図 36-1　　HTTP インスペクションと QoS ポリシング



この例について、次のコマンドを参照してください。

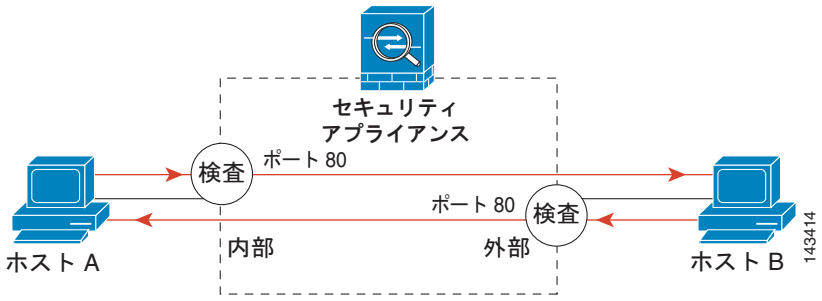
```
hostname(config)# class-map http_traffic
hostname(config-cmap)# match port tcp eq 80

hostname(config)# policy-map http_traffic_policy
hostname(config-pmap)# class http_traffic
hostname(config-pmap-c)# inspect http
hostname(config-pmap-c)# police output 250000
hostname(config)# service-policy http_traffic_policy interface outside
```

HTTP トラフィックへのインスペクションのグローバルな適用

この例（図 36-2）では、任意のインターフェイスを通過して ASA に入るすべての HTTP 接続（ポート 80 の TCP トラフィック）が HTTP インスペクション対象として分類されます。このポリシーはグローバル ポリシーなので、インスペクションが発生するのは各インターフェイスにトラフィックが入ったときだけです。

図 36-2　　グローバル HTTP インスペクション



この例について、次のコマンドを参照してください。

```
hostname(config)# class-map http_traffic
hostname(config-cmap)# match port tcp eq 80
```

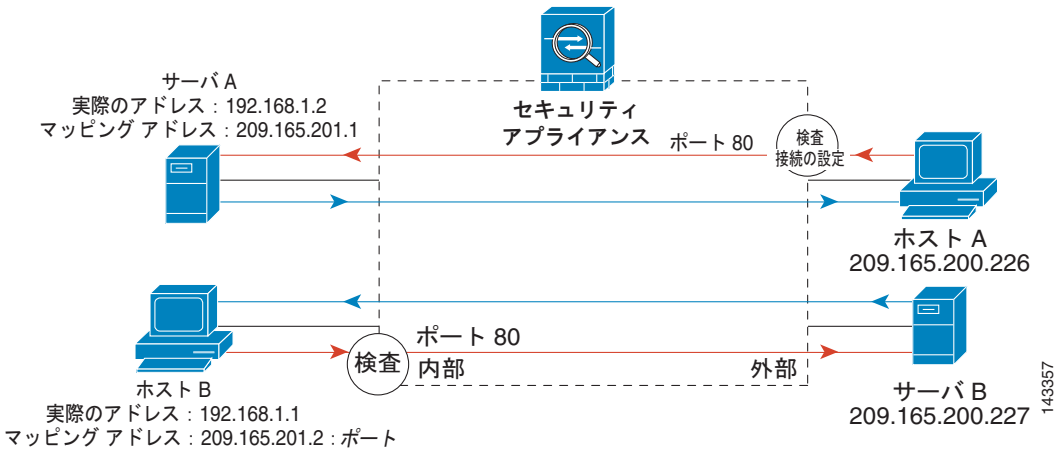
```
hostname(config)# policy-map http_traffic_policy
hostname(config-pmap)# class http_traffic
hostname(config-pmap-c)# inspect http
hostname(config)# service-policy http_traffic_policy global
```

特定のサーバへの HTTP トラフィックに対するインスペクションと接続制限値の適用

この例（図 36-3）では、外部インターフェイスを通過して ASA に入るサーバ A 宛ての HTTP 接続（ポート 80 の TCP トラフィック）が HTTP インスペクションおよび最大接続数制限値の対象として分類されます。サーバ A から発信されたホスト A への接続は、クラスマップのアクセス リストと一致しないので、影響を受けません。

内部インターフェイスを通じて ASA に入るサーバ B 宛てのすべての HTTP 接続は、HTTP インスペクション対象として分類されます。サーバ B から発信されたホスト B への接続は、クラスマップのアクセス リストと一致しないので、影響を受けません。

図 36-3　特定のサーバに対する HTTP インスペクションと接続制限値



この例について、次のコマンドを参照してください。

```
hostname(config)# object network obj-192.168.1.2
hostname(config-network-object)# host 192.168.1.2
hostname(config-network-object)# nat (inside,outside) static 209.165.201.1
hostname(config)# object network obj-192.168.1.0
hostname(config-network-object)# subnet 192.168.1.0 255.255.255.0
hostname(config-network-object)# nat (inside,outside) dynamic 209.165.201.2
hostname(config)# access-list serverA extended permit tcp any host 209.165.201.1 eq 80
hostname(config)# access-list ServerB extended permit tcp any host 209.165.200.227 eq 80

hostname(config)# class-map http_serverA
hostname(config-cmap)# match access-list serverA
hostname(config)# class-map http_serverB
hostname(config-cmap)# match access-list serverB

hostname(config)# policy-map policy_serverA
hostname(config-pmap)# class http_serverA
hostname(config-pmap-c)# inspect http
hostname(config-pmap-c)# set connection conn-max 100
hostname(config)# policy-map policy_serverB
```

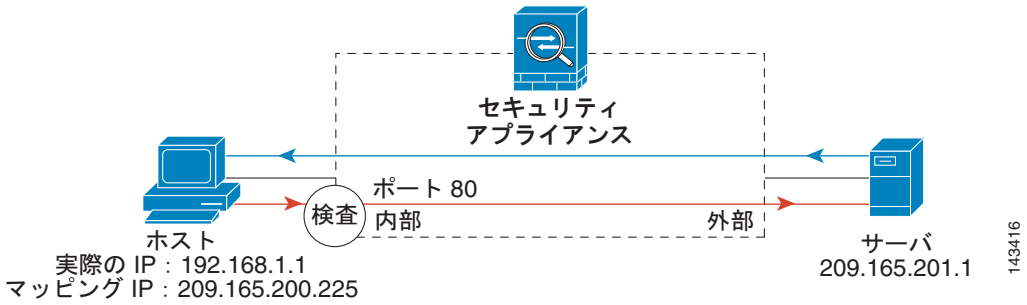
```
hostname(config-pmap)# class http_serverB
hostname(config-pmap-c)# inspect http

hostname(config)# service-policy policy_serverB interface inside
hostname(config)# service-policy policy_serverA interface outside
```

NAT による HTTP トラフィックへのインスペクションの適用

この例では、ネットワーク内のホストに 2 つのアドレスがあります。1 つは、実際の IP アドレスの 192.168.1.1 です。もう 1 つは、外部ネットワークで使用するマッピング IP アドレスの 209.165.200.225 です。クラス マップのアクセス リストに実際の IP アドレスを使用する必要があります。outside インターフェイスに適用する場合にも、実際のアドレスを使用します。

図 36-4　　NAT による HTTP インスペクション



この例について、次のコマンドを参照してください。

```
hostname(config)# object network obj-192.168.1.1
hostname(config-network-object)# host 192.168.1.1
hostname(config-network-object)# nat (VM1,outside) static 209.165.200.225

hostname(config)# access-list http_client extended permit tcp host 192.168.1.1 any eq 80

hostname(config)# class-map http_client
hostname(config-cmap)# match access-list http_client

hostname(config)# policy-map http_client
hostname(config-pmap)# class http_client
hostname(config-pmap-c)# inspect http

hostname(config)# service-policy http_client interface inside
```

サービス ポリシーの機能履歴

表 36-3 に、この機能のリリース履歴の一覧を示します。

表 36-3　サービス ポリシーの機能履歴

機能名	リリース	機能情報
モジュラ ポリシー フレームワーク	7.0(1)	モジュラ ポリシー フレームワークが導入されました。
RADIUS アカウンティング トラフィックで使用する管理クラス マップ	7.2(1)	RADIUS アカウンティング トラフィックで使用する管理クラス マップが導入されました。**class-map type management** コマンドおよび **inspect radius-accounting** コマンドが導入されました。
インスペクション ポリシー マップ	7.2(1)	インスペクション ポリシー マップが導入されました。**class-map type inspect** コマンドが導入されました。
正規表現およびポリシー マップ	7.2(1)	インスペクション ポリシー マップで使用される正規表現およびポリシー マップが導入されました。**class-map type regex** コマンド、**regex** コマンド、および **match regex** コマンドが導入されました。
インスペクション ポリシー マップの match any	8.0(2)	インスペクション ポリシー マップで使用される **match any** キーワードが導入されました。トラフィックを 1 つ以上の基準に照合してクラス マップに一致させることができます。以前は、**match all** だけが使用可能でした。