Flexible Packet Matching

Flexible Packet Matching（FPM）は、アクセス コントロール リスト（ACL）パターン マッチング ツールです。より詳細でカスタマイズされたパケット フィルタが用意されています。FPM を使用すると、パケットの任意のビット、パケット ヘッダーおよびペイロードの任意の深さでマッチングできます。FPM は、パケット インスペクションが制限された特定のフィールドに対する制約を取り除きます。

FPM が便利なのは、独自のステートレス パケットの分類基準を作成し、複数のアクション（インターネット制御メッセージプロトコル（ICMP）unreachable のドロップ、ログ、送信など[1]）についてのポリシーを定義して、新しいウイルス、ワーム、および攻撃をただちにブロックできるためです。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、Bug Search Tool およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

[1] ICMP unreachable の送信は現在、Supervisor Engine 32 PISA ではサポートされていません。

プラットフォームのサポートおよびシスコ ソフトウェアイメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

Flexible Packet Matching の前提条件

XML エディタへのアクセスは必要ありませんが、XML によって、プロトコル ヘッダー記述ファイル（PHDF）の作成は容易になります。

Flexible Packet Matching の制約事項

- FPM では、パケットの先頭 256 バイト以内で、最大 32 バイトの長さまでのパターンを検索できます。
- policy-map では最大 32 クラスがサポートされます。
- IP オプション パケットの場合、FPM はレイヤ 2 ヘッダーのフィールドと IP ヘッダーの先頭 20 バイトのみを検査します。
- 非初期 IP フラグメントの場合、FPM はレイヤ 2 ヘッダーのフィールドと IP ヘッダーの先頭 20 バイトのみを検査します。
- FPM は、ステートフル分類が必要な攻撃の軽減には使用できません。
- FPM はステートレスなので、ダイナミックにポートをネゴシエートするプロトコルで使用されているポート番号は追跡できません。そのため、FPM を使用する場合は、ポート番号を明示的に指定する必要があります。
- FPM は、IP フラグメンテーションまたは TCP フローの再アセンブリを実行できません。
- FPM は、IPv4 ユニキャスト パケットのみを検査します。
- FPM は、IP オプションを使用してパケットを分類できません。
- FPM はマルチキャスト パケット インスペクションをサポートしません。
- FPM はトンネルおよび MPLS インターフェイスでサポートされません。
- FPM エンジンでは非初期フラグメントがマッチングされません。
- マッチング開始構造体では、オフセットに定数のみを使用できます。
- FPM は複数のパケットにわたるマッチングを実行できません。
- コントロール プレーンに対する FPM ポリシーのマッピングはサポートされません。

Flexible Packet Matching の概要

Flexible Packet Matching 機能の概要

FPM を使用すると、新しいウイルスや攻撃をただちに検出してブロックできる独自のフィルタリング ポリシーを作成できます。

フィルタリング ポリシーは次の作業で定義されます。

- PHDF をロードします（プロトコル ヘッダー フィールド マッチングの場合）
- クラス マップを定義し、プロトコル スタック チェーン（トラフィック クラス）を定義します
- サービス ポリシー（トラフィック ポリシー）を定義します
- サービス ポリシーをインターフェイスに適用します

プロトコル ヘッダー記述ファイル

プロトコル ヘッダーは、PHDF という個別のファイルで定義されます。PHDF 内に定義されているフィールド名は、パケット フィルタの定義に使用されます。PHDF は、XML の柔軟性を利用して、ほぼすべてのプロトコル ヘッダーを記述できるファイルです。PHDF の重要なコンポーネントは、バージョン、XML ファイル スキーマの場所、およびプロトコル フィールド定義です。プロトコル フィールド定義では、プロトコル ヘッダーの適切なフィールドに名前を付け、フィールドを説明するコメントを考慮し、ヘッダーのプロトコル ヘッダー フィールドの場所を提供し（オフセットはプロトコル ヘッダーの開始に相対的です）、フィールドの長さを提供します。バイト単位またはビット単位を指定できます。

（注） ヘッダーの合計の長さは、各 PHDF の末尾で指定する必要があります。

（注） 冗長 sup PHDF ファイルが FPM ポリシーで使用される場合、ファイルはスタンバイ sup の対応するディスク上にも存在する必要があります。ファイルが使用できない場合、FPM ポリシーはスイッチオーバー後に動作しません。

既存または専用のプロトコルの場合、XML を介して独自のカスタム PHDF を作成できます。ただし、**load protocol** コマンドを介して、ip.phdf、ether.phdf、tcp.phdf、および udp.phdf の標準 PHDF もルータに読み込むことができます。

（注） PHDF は XML を介して定義するため、実行コンフィギュレーションでは表示されません。ただし、**show protocol phdf** コマンドを使用して、読み込まれた PHDF を確認できます。

標準 PHDF は、Cisco.com（URL：http://www.cisco.com/cgi-bin/tablebuild.pl/fpm）で入手できます。

フィルタ記述

フィルタ記述は、（**matchfield** コマンドを使用して）PHDF に定義するヘッダー フィールドを含むことができるトラフィック クラスの定義です。PHDF が読み込まれていない場合、トラフィック クラスは、（**matchstart** コマンドを使用して）データグラム ヘッダーの開始（レイヤ 2）またはネットワーク ヘッダーの開始（レイヤ 3）を介して定義します。PHDF がルータに読み込まれた場合、クラスの指定は、パケット内のプロトコル ヘッダーのリストから始まります。

また、フィルタ定義にはポリシーマップも含まれます。つまり、クラスマップを定義した後は、一致をアクションにバインドするためにポリシーマップが必要です。ポリシーマップは、順序が指定されたクラスと関連するアクション（ICMP unreachable のドロップ、ログ、送信など）のセットです。

FPM のクラス マップとポリシー マップの設定方法については、セクション「How to Configure a Flexible Packet Matching Traffic Class and Traffic Policy」を参照してください。

Flexible Packet Matching の設定方法

Flexible Packet Matching のトラフィック クラスの作成

（注） PHDF プロトコル フィールドがアクセス コントロール クラスマップで参照されない場合、FPM が適切に動作するには、スタック クラスマップが必要です。

手順の概要

1. **enable**
2. **configureterminal**
3. **load protocol***location:filename*
4. **class-map** [**type** {**stack** | **access-control**}] *class-map-name* [**match-all** | **match-any**]
5. **description***character-string*
6. **match field***protocol protocol-field* {**eq** [*mask*] | **neq** | [*mask*] | **gt** | **lt** | **range***range* | **regex***string*} *value* [**next***next-protocol*]
7. **match start** {**l2-start** | **l3-start**} **offset***number***size***number* {**eq** | **neq** | **gt** | **lt** | **range***range* | **regex***string*} {*value* [*value2*] | [*string*]}
8. **match class***class-name* [**packet-range***low high* | **byte-range***low high*] **session**
9. **exit**
10. **exit**
11. **show class-map** [**type** {**stack** | **access-control**} | *class-map-name*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configureterminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**load protocol***location:filename* 例： `Router(config)# load protocol disk2:udp.phdf`	（任意）PHDF をルータにロードします。 • 指定する場所は、ルータのローカルにする必要があります。 （注） PHDF が読み込まれていない場合、**matchstart** コマンドのみを使用できます。つまり、**matchfield** コマンドは発行できません。 （注） ASR プラットフォームの場合、PHDF ファイルは、アクティブおよびスタンバイ ルート プロセッサ（RP）ファイル システムに（**loadprotocol** コマンドを介して）手動でコピーする必要があります。

	コマンドまたはアクション	目的
ステップ 4	**class-map** [**type** {**stack** | **access-control**}] *class-map-name* [**match-all** | **match-any**] 例： `Router(config)# class-map type access-control c1`	指定したクラスとパケットのマッチングに使用されるクラスマップを作成し、クラスマップ コンフィギュレーション モードを開始します。 • **type stack**：FPM で、確認するための正しいプロトコル スタックを決定できます。 • **type access-control**：該当するプロトコル スタック内を検索するための正確なパターンを決定します。 • *class-map-name*：最大 40 個の英数字を指定できます。 • **match-all** または **match-any** が指定されない場合、トラフィックがトラフィック クラスの一部として分類されるには、すべての一致基準に適合する必要があります。
ステップ 5	**description***character-string* 例： `Router(config-cmap)# description "match on slammer packets"`	（任意）説明をクラス マップに追加します。
ステップ 6	**match field***protocol protocol-field* {**eq** [*mask*] | **neq** | [*mask*] | **gt** | **lt** | **range***range* | **regex***string*} *value* [**next***next-protocol*] 例： `Router(config-cmap)# match field udp dest-port eq 0x59A`	（任意）PHDFに定義されているフィールドに基づいて、クラスマップの一致基準を設定します。 • **next***next-protocol* キーワード引数ペアは、**class-map type stack** コマンドを設定した後にのみ使用できます。
ステップ 7	**match start** {**l2-start** | **l3-start**} **offset***number***size***number* {**eq** | **neq** | **gt** | **lt** | **range***range* | **regex***string*} {*value* [*value2*] | [*string*]} 例： `Router(config-cmap)# match start l3-start offset 224 size 4 eq 0x4011010`	（任意）データグラム ヘッダー（レイヤ 2）またはネットワーク ヘッダー（レイヤ 3）に基づいて、クラス マップの一致基準を設定します。
ステップ 8	**match class***class-name* [**packet-range***low high* | **byte-range***low high*] **session** 例： `Router(config-cmap)# match class c2 packet-range 1 5 session`	（任意）対象のパケットを含むセッション（フロー）を特定するクラスマップの一致基準を設定します。この基準は、セッション中に送信されるすべてのパケットに適用されます。 **packet-range** キーワードおよび **byte-range** キーワードは、各パケット フローのパケット番号/パケット バイトの限られた範囲内にあるトラフィックを分類することによって、regex ベースの

	コマンドまたはアクション	目的
		FPM クラス マップのパフォーマンスと一致精度を向上させるフィルタ メカニズムを作成します。 **session** キーワードが引数 *class-name* とともに使用されると、分類結果は同じパケット セッションの後続のパケット用に保存されます。 **session** キーワードが **packet-range** または **byte-range** キーワードとともに使用されると、分類結果は同じパケット セッションの指定されたパケットまたはバイト用に保存されます。
ステップ 9	**exit** 例： `Router(config-cmap)# exit`	クラスマップ コンフィギュレーション モードを終了します。
ステップ 10	**exit** 例： `Router(config)# exit`	グローバル コンフィギュレーション モードを終了します。
ステップ 11	**show class-map** [**type** {**stack** | **access-control**} | *class-map-name*] 例： `Router# show class-map type access-control slammer`	（任意）設定された FPM クラス マップを表示します。

トラブルシューティングのヒント

すべての FPM イベントを追跡するには、**debug fpm event** コマンドを発行します。

次に、**debug fpm event** コマンドの出力例を示します。

```
*Jun 21 09:22:21.607: policy-classification-inline(): matches class: class-default *Jun 21
 09:22:21.607: packet-access-control(): policy-map: fpm-policy, dir: input, match. retval:
 0x0, ip-flags: 0x80000000
```

次の作業

ネットワークに 1 つ以上のクラス マップを定義したら、次のタスク「Flexible Packet Matching のトラフィック ポリシーの作成」に示すように、トラフィック ポリシーを作成し、そのポリシーをインターフェイスに適用する必要があります。

Flexible Packet Matching のトラフィック ポリシーの作成

手順の概要

1. **enable**
2. **configure terminal**
3. **policy-map type access-control***policy-map-name*
4. **description***character-string*
5. **class***class-name***insert-before***class-name*
6. **drop** [**all**]
7. **log** [**all**]
8. **service-policy***policy-map-name*
9. **exit**
10. **interface***type number*
11. **service-policy type access-control** {**input** | **output**} *policy-map-name*
12. **exit**
13. **exit**
14. **show policy-map** [**type access-control** | **interface***typenumber* | **input** | **output**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**policy-map type access-control***policy-map-name* 例： `Router(config)# policy-map type access-control fpm-udp-policy`	サービス ポリシーを指定するために 1 つ以上のインターフェイスに対応付けることができるポリシー マップを作成または修正し、policy-map コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	**description***character-string* 例： `Router(config-pmap)# description "policy for UDP based attacks"`	（任意）説明をポリシー マップに追加します。
ステップ 5	**class***class-name***insert-before***class-name* 例： `Router(config-pmap)# class slammer`	**class-map** コマンドを使用して設定された、事前定義済みトラフィック クラスの名前を指定します。また、**class** コマンドは、トラフィックをトラフィック ポリシーに合わせて分類し、ポリシー マップ クラス コンフィギュレーション モードを開始します。 • **insert-before***class-name* キーワードと引数は、ポリシー マップ内の任意の場所にクラスマップを追加します。このオプションを指定しない場合、クラスマップはポリシーマップの末尾に追加されます。
ステップ 6	**drop** [**all**] 例： `Router(config-pmap-c)# drop all`	（任意）トラフィック クラスを、特定のクラスに属するパケットを廃棄するよう設定します。 **all** キーワードは、そのトラフィック クラスに属するパケットのストリーム全体を廃棄するのに使用します。 このコマンドを発行する場合、次の制約事項に注意してください。 • パケットの廃棄は、トラフィック クラスで設定できる唯一のアクションです。 • トラフィック クラスを **drop** コマンドで設定する場合、この特定のトラフィック クラスの「子」（入れ子になっている）ポリシーは、**servicepolicy** コマンドを使用して設定することはできません。 • パケットの廃棄は、**classclass-default** コマンドを介して指定されたデフォルト クラスには設定できません。 • **dropall** コマンドが指定されている場合、このコマンドを関連付けることができるのは **classmaptypeaccess-control** コマンドのみです。
ステップ 7	**log** [**all**] 例： `Router(config-pmap-c)# log all`	（任意）トラフィック クラスのログ メッセージを生成します。 **all** キーワードは、そのトラフィック クラスに属する廃棄されたパケットのストリーム全体を記録するのに使用します。このキーワードは、**class-map type access-control** コマンドで作成したクラス マップでのみ使用可能です。

	コマンドまたはアクション	目的
ステップ 8	**service-policy***policy-map-name* 例： `Router(config-pmap-c)# service policy fpm-udp-policy`	階層的なサービス ポリシーを作成します。

ステップ 9	**exit**
	例：
	`Router(config-pmap-c)# exit`
	例：
	`Router(config-pmap)# exit`
		ポリシー マップ クラス コンフィギュレーション モードおよびポリシーマップ コンフィギュレーション モードを終了します。

ステップ 10	**interface***type number* 例： `Router(config)# interface gigabitEthernet 0/1`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 11	**service-policy type access-control** {**input** | **output**} *policy-map-name* 例： `Router(config-if)# service-policy type access-control input fpm-policy`	インターフェイスの入力方向または出力方向に対応付けるトラフィック ポリシーの種類と名前を指定します。
ステップ 12	**exit** 例： `Router(config-if)# exit`	インターフェイス コンフィギュレーション モードを終了します。
ステップ 13	**exit** 例： `Router(config)# exit`	グローバル コンフィギュレーション モードを終了します。
ステップ 14	**show policy-map** [**type access-control** | **interface***typenumber* | **input** | **output**] 例： `Router# show policy-map type`	（任意）FPM コンフィギュレーションを確認します。 （注） トラフィック ポリシーを FPM 向けに作成した後は、一致したパケットを別の宛先インターフェイスにコピーまたはリダイレクトできます。

	コマンドまたはアクション	目的
	`access-control interface gigabitethernet 0/1`	

FPM コンフィギュレーションの設定例

ASR プラットフォームでの FPM の設定と確認：例

次に、ASR プラットフォームで FPM を設定する例を示します。

```
load protocol bootflash:ip.phdf
load protocol bootflash:tcp.phdf
class-map type stack match-all ip_tcp
 match field IP protocol eq 6 next TCP
class-map type access-control match-all test_class
 match field TCP dest-port gt 10
 match start l3-start offset 40 size 32 regex "ABCD"
policy-map type access-control child
 class test_class
  drop
policy-map type access-control parent
 class ip_tcp
  service-policy child
interface GigabitEthernet0/3/0
 ip address 10.1.1.1 255.0.0.0
 service-policy type access-control input parent
```

次の出力例では、すべての TCP パケットは class-map「ip_tcp」の下に表示され、特定のパターンに一致するすべてのパケットは class-map「test_class」の下に表示されます。特定のパターンのない TCP パケットは子ポリシーの「class-default」の下に表示され、すべての非 TCP パケットは親ポリシーの「class-default」の下に表示されます。（この例では、カウンタは 0 です）。

```
Router# show policy-map type access-control interface GigabitEthernet0/3/0
GigabitEthernet0/3/0
 Service-policy access-control input: parent
  Class-map: ip_tcp (match-all)
  2024995578 packets, 170099628552 bytes
  5 minute offered rate 775915000 bps
  Match: field IP version eq 4
  Match: field IP ihl eq 5
  Match: field IP protocol eq 6 next TCP
 Service-policy access-control : child
 Class-map: test_class (match-all)
  1598134279 packets, 134243279436 bytes
  5 minute offered rate 771012000 bps, drop rate 771012000 bps
  Match: field TCP dest-port gt 10
  Match: start l3-start offset 40 size 32 regex "ABCD"
 drop
 Class-map: class-default (match-any)
  426861294 packets, 35856348696 bytes
  5 minute offered rate 4846000 bps, drop rate 0 bps
  Match: any
 Class-map: class-default (match-any)
  0 packets, 0 bytes
  5 minute offered rate 0 bps, drop rate 0 bps
```

```
  Match: any
Router#
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』
セキュリティ コマンド	『*Cisco IOS Security Command Reference*』
トラフィック分類定義ファイルを使用したFPMの設定	『*Cisco IOS Security Configuration Guide: Securing the Data Plane*』の「Flexible Packet Matching XML Configuration」モジュール
Quality of Service（QoS）コマンド群	『*Cisco IOS Quality of Service Solutions Command Reference*』

MIB

MIB	MIB のリンク
なし	選択したプラットフォーム、Cisco IOS XE ソフトウェア リリース、およびフィーチャ セットのMIBの場所を検索しダウンロードするには、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

シスコのテクニカル サポート

説明	リンク
右の URL にアクセスして、シスコのテクニカルサポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

Flexible Packet Matching に関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *1*：*Flexible Packet Matching* に関する機能情報

機能名	リリース	機能情報
Flexible Packet Matching	Cisco IOS XE Release 2.2	FPMは、標準のマッチング演算子とユーザ定義のプロトコル ヘッダー フィールドを組み合わせることで、1つまたは複数のネットワーク トラフィック クラスを定義できるパケット分類機能です。 導入された、または変更されたコマンドは、**class (policy-map) class-map debug fpm event**、**description(class-map) load protocol matchfield matchstart**、**policy-map**、**service-policy**、**show class-map**、**show policy-map interface**、**showprotocolphdf** です。