ゾーンのフィルタの設定

この章では、Cisco Traffic Anomaly Detector Module（Detector モジュール）のフィルタの設定方法について説明します。

この章には、次の項があります。

概要

ゾーン フィルタは、Detector が特定のトラフィック フローを処理する方法を定義します。ユーザは、フィルタを設定して、カスタマイズされたトラフィック誘導メカニズムや DDoS 攻撃検出メカニズムをさまざまに設計することができます。

Detector モジュールには、次のタイプのフィルタがあります。

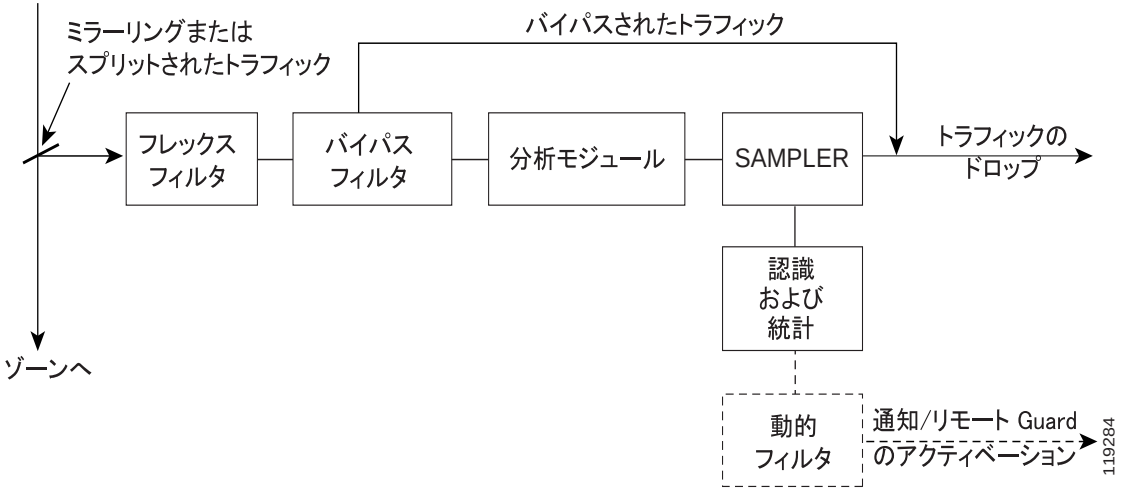
- **バイパス フィルタ**：バイパス フィルタは、特定のトラフィック フローが Detector の保護メカニズムによって処理されないようにします。

  信頼されたトラフィックが Detector モジュールの検出メカニズムを通らないように誘導し、Detector モジュールによって分析されないようにすることができます。

  詳細については、P.6-19 の「バイパス フィルタの設定」を参照してください。
- **フレックスコンテンツ フィルタ**：フレックスコンテンツ フィルタは、特定のトラフィック フローをカウントします。フレックスコンテンツ フィルタは、バークリー パケット フィルタとパターン フィルタを組み合せたもので、IP ヘッダーと TCP ヘッダーのフィールドに基づいたフィルタリングやペイロード コンテンツに基づいたフィルタリングなどの非常に柔軟なフィルタリング機能、および複雑なブール式をユーザに提供します。詳細については、P.6-6 の「フレックスコンテンツ フィルタの設定」を参照してください。
- **動的フィルタ**：動的フィルタは、特定のトラフィック フローを関連する Detector の検出モジュールに誘導します。Detector は、トラフィック フローの分析結果として動的フィルタを作成します。この一連のフィルタは、ゾーンのトラフィックおよび特定の DDoS 攻撃に合せて継続的に調整されます。動的フィルタは有効期間が限定されており、攻撃が終了すると消去されます。詳細については、P.6-21 の「動的フィルタの設定」を参照してください。

図 6-1 に、Detector モジュールのフィルタ システムを示します。

図 6-1　Detector のフィルタ システム



Detector は、トラフィック フローを、ゾーンのトラフィックを分析する分析検出モジュールに誘導します。バイパス フィルタを使用すると、特定のフローが Detector の検出メカニズムをバイパスするように設定できます。

トラフィック フローの統計分析を実行するため、Detector には特定のタイプのトラフィックを処理する方法についての定義が含まれています。これらの定義をポリシーといいます。ポリシーは、トラフィック フローを持続的に測定し、特定のトラフィック フローが悪意のあるものまたは異常であると判断すると、そのフローに対してアクションを実行します。このアクションは、フローがポリシーのしきい値を超過すると発生します。Detector が実行するアクションでは、イベントが syslog に記録されるか、新しいフィルタ（動的フィルタ）が作成されます。動的フィルタは、Detector のリストにあるリモート Guard をリモートでアクティブにできます。

フィルタのトラフィック フロー

フィルタが処理するトラフィック フローを設定する必要があります。表 6-1 に、フィルタのフローの引数を示します。

詳細については、「バイパス フィルタの設定」と「動的フィルタの設定」の各項を参照してください。

表 6-1　フィルタのフローの引数

パラメータ	説明
src-ip	特定の IP アドレスからのフローを処理します。すべての IP アドレスを示すには、アスタリスク（*）を使用します。
ip-mask	（オプション）特定のサブネットからのフローを処理します。サブネット マスクには、クラス C の値のみを指定できます。デフォルトのサブネットは、255.255.255.255 です。
protocol	特定のプロトコルのフローを処理します。すべてのプロトコルを示すには、アスタリスク（*）を使用します。 指定可能なプロトコル番号については、次に示す Internet Assigned Numbers Authority（IANA; インターネット割り当て番号局）の Web サイトを参照してください。 http://www.iana.org/assignments/protocol-numbers
dest-port	特定の宛先ポートに向かうトラフィックを処理します。すべての宛先ポートを示すには、アスタリスク（*）を使用します。 指定可能なポート番号については、次に示す IANA の Web サイトを参照してください。 http://www.iana.org/assignments/port-numbers
fragments-type	（オプション）断片化されたトラフィックをフィルタが処理するかどうかを指定します。断片化のタイプは、次のとおりです。 • **no-fragments**：断片化されていないトラフィック • **fragments**：断片化されたトラフィック • **any-fragments**：断片化されたトラフィックと断片化されていないトラフィック デフォルトは、**no-fragments** です。

表 6-2 に、フィルタの **show** コマンドのフィールドを示します。

詳細については、「バイパス フィルタの表示」と「動的フィルタの表示」の各項を参照してください。

表 6-2　フィルタの show コマンドのフィールドの説明

フィールド	説明
Source IP	フィルタが処理するトラフィックの送信元 IP アドレスを指定します。
Source Mask	フィルタが処理するトラフィックの送信元アドレスのマスクを指定します。
Proto	フィルタが処理するトラフィックのプロトコル番号を指定します。
DPort	フィルタが処理するトラフィックの宛先ポートを指定します。
Frg	フィルタが断片化されたトラフィックを処理するかどうかを指定します。 • **yes**：フィルタは断片化されたトラフィックを処理します。 • **no**：フィルタは断片化されていないトラフィックを処理します。 • **any**：フィルタは、断片化されたトラフィックと断片化されていないトラフィックの両方を処理します。

送信元 IP アドレス、送信元アドレスのマスク、プロトコル番号、および宛先ポートは、特定のものでなくてもかまいません。アスタリスク（*）は、フィルタがすべてのフィールド値に対して動作するか、フィルタに複数の値が一致したことを示します。

フレックスコンテンツ フィルタの設定

フレックスコンテンツ フィルタを使用すると、パケット ペイロードにあるパケット ヘッダーまたはパターンのフィールドに基づいて、ゾーンのトラフィックをフィルタリングできます。着信トラフィックに現れているパターンに基づいて攻撃を識別できます。そのようなパターンは、一定のパターンを持つ既知のワームまたはフラッド攻撃を識別できます。ただし、フレックスコンテンツ フィルタはリソースを大量に消費します。フレックスコンテンツ フィルタはパフォーマンスに影響を及ぼす可能性があるため、十分に注意して使用することをお勧めします。フレックスコンテンツ フィルタを使用して特定の攻撃を検出しようとする場合、その攻撃が動的フィルタで識別可能なときは（特定のポートへの TCP トラフィックなど）、動的フィルタを使用してトラフィックをフィルタリングすることをお勧めします。

フレックスコンテンツ フィルタは、目的のパケット フローをカウントし、トラフィックの特定の悪意ある送信元を明らかにするために使用します。

フレックスコンテンツ フィルタは、強力な選別フィルタリング機能を持つ、バークリー パケット フィルタとパターン フィルタを組み合せたものです。

この項では、次のトピックについて取り上げます。

フレックスコンテンツ フィルタの追加

フレックスコンテンツ フィルタは、行番号の昇順でアクティブになります。したがって、新しいフレックスコンテンツ フィルタを追加する場合には、リストの適切な位置に配置することが重要です。

フレックスコンテンツ フィルタを設定するには、次の手順を実行します。

ステップ 1　フレックスコンテンツ　フィルタのリストを表示して、リスト内で新しいフィルタを追加する位置を確認します。詳細については、P.6-15 の「フレックスコンテンツ フィルタの表示」を参照してください。

ステップ 2　現在の行番号が連続したものである場合は、新しいフレックスコンテンツ　フィルタを挿入できるようにフレックスコンテンツ フィルタの番号を順に増加させます。次のコマンドを入力します。

```
flex-content-filter renumber [start [step]]
```

表 6-3 に、**flex-filter renumber** コマンドの引数を示します。

表 6-3　flex-filter renumber コマンドの引数

パラメータ	説明
start	(オプション)フレックスコンテンツ フィルタ リストの新しい開始番号を示す 1 ～ 9999 の整数。デフォルトは 10 です。
step	(オプション) フレックスコンテンツ フィルタの各行番号の増分を指定する 1 ～ 999 の整数。デフォルトは 10 です。

ステップ 3　(オプション) 進行中の攻撃または以前に記録した攻撃のパターン式をフィルタリングする場合、Detector モジュールをアクティブにし、**show packet-dump signatures** コマンドを使用して攻撃のシグニチャを生成できます。詳細については、P.10-26 の「パケットダンプ キャプチャ ファイルからの攻撃シグニチャの生成」を参照してください。

ステップ 4　新しいフレックスコンテンツ　フィルタを追加します。次のコマンドを入力します。

```
flex-content-filter row-num {disabled | enabled} {drop | count}
protocol port [start start-offset [end end-offset]] [ignore-case]
expression tcpdump-expression pattern pattern-expression
```

表 6-4 に、**flex-filter** コマンドの引数とキーワードを示します。

表 6-4　flex-filter コマンドの引数とキーワード

パラメータ	説明
row-num	1 ～ 9999 の固有な番号。行番号はフィルタの ID で、これによって複数のフレックスコンテンツ フィルタの優先順位が定義されます。Detector モジュールは、行番号の昇順でフィルタを操作します。
disabled	フィルタの状態をディセーブルに設定します。フィルタはトラフィックに関連付けられません。
enabled	フィルタの状態をイネーブルに設定します。フィルタはトラフィックに関連付けられ、一致が検出されるとアクションを実行します。 これがデフォルトの状態です。
count	フィルタに一致するフローをカウントします。
protocol	特定のプロトコルのフローを処理します。すべてのプロトコルを示すには、アスタリスク（*）を使用します。0 ～ 255 の整数を入力します。 指定可能なプロトコル番号については、次に示す IANA の Web サイトを参照してください。 http://www.iana.org/assignments/protocol-numbers
port	特定の宛先ポートに向かうトラフィックを処理します。0 ～ 65535 の整数を入力します。特定のポート番号を定義するには、特定のプロトコル番号を定義する必要があります。 すべての宛先ポートを示すには、アスタリスク（*）を使用します。プロトコル番号を 6（TCP）または 17（UDP）に設定する場合のみ、アスタリスクを使用できます。 指定可能なポート番号については、次に示す IANA の Web サイトを参照してください。 http://www.iana.org/assignments/port-numbers

表 6-4　flex-filter コマンドの引数とキーワード（続き）

パラメータ	説明
start-offset	パケット ペイロードの先頭から、パターン マッチングを開始する位置までのオフセット（バイト単位）。デフォルトは 0（ペイロードの先頭）です。このオフセットは、*pattern* フィールドに適用されます。0 ～ 1800 の整数を入力します。 **show packet-dump signatures** コマンド出力からパターンをコピーする場合は、コマンド出力の Start Offset フィールドからこの引数をコピーします。
end-offset	パケット ペイロードの先頭から、パターン マッチングを終了する位置までのオフセットを指定します（バイト単位）。デフォルトはパケット長（ペイロードの末尾）です。このオフセットは、*pattern* フィールドに適用されます。0 ～ 1800 の整数を入力します。 **show packet-dump signatures** コマンド出力からパターンをコピーする場合は、コマンド出力の End Offset フィールドからこの引数をコピーします。
ignore-case	*pattern-expression* 引数で大文字と小文字が区別されないようにします。 デフォルトでは、*pattern-expression* 引数では大文字と小文字が区別されます。
tcpdump-expression	パケットと照合する式を指定します。式はバークリー パケット フィルタの形式です。詳細と設定例については、P.6-11 の「TCPDump 式の構文について」を参照してください。 式にスペースが含まれる場合は、式を引用符で囲みます。 空の式を入力するには、二重引用符（“ “）を使用します。 式に引用符を含めるには、バックスラッシュ（\）をエスケープ文字として使用します。 TCPDump 式の構文については、ヘルプを使用できません。

表 6-4　flex-filter コマンドの引数とキーワード（続き）

パラメータ	説明
pattern-expression	パケット ペイロードと照合する正規表現のデータ パターンを指定します。詳細については、P.6-14 の「パターン式の構文について」を参照してください。 Detector モジュールをアクティブにし、**show packet-dump signatures** コマンドを使用してシグニチャを生成できます。P.10-26 の「パケットダンプ キャプチャ ファイルからの攻撃シグニチャの生成」を参照してください。 式にスペースが含まれる場合は、式を引用符で囲みます。 空の式を入力するには、二重引用符（“ “）を使用します。 式に引用符を含めるには、バックスラッシュ（\）をエスケープ文字として使用します。 パターン式の構文については、ヘルプを使用できません。

フレックスコンテンツ フィルタは、次の順序でフィルタリング基準を適用します。

- フレックスコンテンツ フィルタは最初に、*protocol* および *port* に基づいてパケットをフィルタリングする。
- 次に、*tcpdump-expression* を適用する。
- フレックスコンテンツ フィルタは、残りのパケットに対して *pattern-expression* を使用してパターン マッチングを実行する。

フィルタの状態はいつでも変更できます。詳細については、P.6-18 の「フレックスコンテンツ フィルタの状態の変更」を参照してください。

次の例を参考にしてください。

```
user@DETECTOR-conf-zone-scannet# flex-content-filter enabled count * *
expression “ip[6:2] & 0x1fff=0” pattern
“/ HTTP/1\.1\ x0D\0AAccept: .*/.*\x0D\x0AAccept-Language:
en*\x0D\x0AAccept-Encoding: gzip, deflate\x0D\x0AUser-Agent:
Mozilla/4\.0”
```

TCPDump 式の構文について

TCPDump 式は、パケットを照合する式を指定します。式はバークリー パケット フィルタの形式です。

（注）　TCPDump 式を使用すると、宛先ポートおよびプロトコルに基づいてトラフィックをフィルタリングできます。ただし、パフォーマンスへの影響を考慮し、これらの基準でトラフィックをフィルタリングする場合は、フレックスコンテンツ フィルタで *protocol* 引数と *port* 引数を使用することをお勧めします。

表 6-5 に、フレックスコンテンツ フィルタの式のパラメータの説明を示します。

表 6-5　フレックスコンテンツ フィルタのパラメータ

パラメータ	説明
dst host *host_ip_address*	宛先ホスト IP アドレスへのトラフィック。
src host *host_ip_address*	送信元ホスト IP アドレスからのトラフィック。
host *host_ip_ address*	送信元および宛先の両方のホスト IP アドレスの間のトラフィック。
net *net* **mask** *mask*	特定のネットワークへのトラフィック。
net *net/len*	特定のサブネットへのトラフィック。
dst port *destination_port_number*	宛先ポート番号への TCP または UDP トラフィック。
src port *source_port_number*	送信元ポート番号からの TCP または UDP トラフィック。
port *port_number*	送信元および宛先の両方のポート番号間の TCP または UDP トラフィック。
less *packet_length*	特定のバイト長以下の長さを持つパケット。
greater *packet_length*	特定のバイト長以上の長さを持つパケット。
ip proto *protocol*	ICMP、UDP、または TCP のプロトコル番号を持つパケット。
ip broadcast	ブロードキャスト IP パケット。

表 6-5　フレックスコンテンツ フィルタのパラメータ（続き）

パラメータ	説明
ip multicast	マルチキャスト パケット。
ether proto *protocol*	IP、ARP、または RARP などの特定のプロトコル番号またはプロトコル名を持つイーサネット プロトコル パケット。
expr relop expr	特定の式に適合するトラフィック。詳細については、表 6-6 を参照してください。

表 6-6 に、フレックスコンテンツ フィルタの式の規則の説明を示します。

表 6-6　フレックスコンテンツ フィルタの式の規則

式の規則	
relop	>、<、>=、<=、=、!=
expr	整数の定数（標準の C 構文で表現されたもの）、通常のバイナリ演算子（+、-、*、/、&、|）、長さ演算子、および特殊なパケット データ アクセスで構成される算術式。パケット内のデータにアクセスするには、次の構文を使用します。 *proto* [*expr*: *size*]
proto	インデックス操作用のプロトコル層を指定します。指定可能な値は、**ether**、**ip**、**tcp**、**udp**、または **icmp** です。指定されたプロトコル層までの相対的なバイト オフセットは、*expr* で指定されます。*size* 引数はオプションです。目的のフィールドのバイト数を示し、1、2、または 4 になります。デフォルトは 1 です。*len* 引数には、パケットの長さを指定します。

次の方法により、プリミティブを組み合せることができます。

- プリミティブとオペレータを小カッコで囲んだグループ（小カッコはシェルの特殊文字であるため、エスケープする必要があります）。
- 否定：**!** または **not** を使用します。
- 連結：**&&** または **and** を使用します。
- 代替：|| または **or** を使用します。

否定は、最も高い優先度を持ちます。代替と連結の優先順位は同じで、左から右に関連付けられます。連結には、並置ではなく、明示的な **and** トークンが必要です。キーワードなしで識別子を指定した場合は、最後に指定されたキーワードが使用されます。

バークリー パケット フィルタの設定オプションの詳細については、http://www.freesoft.org/CIE/Topics/56.htm を参照してください。

次の例は、断片化されていないデータグラムと断片化されたデータグラムのフラグメント 0 のみをカウントする方法を示しています。このフィルタは、TCP と UDP のインデックス操作に暗黙的に適用されます。たとえば、tcp[0] は常に TCP ヘッダーの最初のバイトを意味し、中間のフラグメントの最初のバイトを意味することはありません。

```
user@DETECTOR-conf-zone-scannet# flex-content-filter enabled count * *
expression ip[6:2]&0x1fff=0 pattern ""
```

次の例は、すべての TCP RST パケットをカウントする方法を示しています。

```
user@DETECTOR-conf-zone-scannet# user@DETECTOR-conf-zone-scannet#
flex-content-filter enabled count * * expression tcp[13]&4!=0 pattern
""
```

次の例は、エコー要求およびエコー応答（ping）ではないすべての ICMP パケットをカウントする方法を示しています。

```
user@DETECTOR-conf-zone-scannet# flex-content-filter enabled count * *
expression "icmp [0]!=8 and icmp[0] != 0" pattern ""
```

次の例は、ポート 80 を宛先とし、ポート 1000 を送信元としないすべての TCP パケットをカウントする方法を示しています。

```
user@DETECTOR-conf-zone-scannet# flex-content-filter enabled count * *
expression "tcp and dst port 80 and not src port 1000" pattern ""
```

パターン式の構文について

パターン（正規表現）は、一連の文字を含んだ文字列を記述したものです。パターンは、一連の文字列をその要素を実際にリストせずに表現します。パターンは、一般文字と特殊文字で構成されます。一般文字には、特殊文字とは見なされない印刷可能な ASCII 文字が含まれます。特殊文字は、どのようなマッチングを実行するのかを示します。フレックスコンテンツ フィルタは、このパターンをパケットの内容（パケットのペイロード）と照合します。たとえば、*version 3.1*、*version 4.0*、および *version 5.2* という 3 つの文字列は、*version .*\..** というパターンで表現できます。

特殊文字は、特殊な意味を持つ文字であり、Detector モジュールが式に対してどのようなマッチングを実行するかを指定します。表 6-7 に、使用できる特殊文字を示します。

表 6-7　フレックスコンテンツ パターン フィールドの説明

特殊文字	説明
.*	0 個またはそれ以上の文字を含んでいる文字列と一致します。たとえば、パターン *goo.*s* は *goos*、*goods*、*good for ddos* などと一致します。
\	特殊文字が持つ特殊な意味を取り除きます。特殊文字を文字列の中で 1 つの文字パターンとして使用するには、各文字の先頭にバックスラッシュ（\）を入力して特殊な意味を取り除きます。たとえば、シーケンス \\ は \ と一致し、シーケンス \. は . と一致します。 文字として使用するアスタリスク（*）の前にもバックスラッシュを配置する必要があります。
\xHH	16 進値と一致します。H は 16 進数の数字で、大文字と小文字は区別されません。16 進値は 2 桁で入力する必要があります。たとえば、\x41 は A と一致します。

デフォルトでは、パターンでは大文字と小文字が区別されます。パターン式で大文字と小文字が区別されないようにするには、**ignore-case** キーワードを使用します。P.6-6 の「フレックスコンテンツ フィルタの追加」を参照してください。

次の例は、パケットのペイロードに特定のパターンが含まれているパケットをドロップする方法を示しています。この例のパターンは、Slammer ワームから抽出されたものです。プロトコル、ポート、および tcpdump 式は特定のものでなくてもかまいません。

```
user@DETECTOR-conf-zone-scannet# flex-content-filter enabled drop * *
expression " " pattern
\x89\xE5Qh\.dllhel32hkernQhounthickChGetTf\xB9ll
Qh32\.dhws2_f\xB9etQhsockf\xB9toQhsend\xBE\x18\x10\xAEB
```

フレックスコンテンツ フィルタの表示

フレックスコンテンツ フィルタを表示するには、ゾーン設定モードで次のコマンドを入力します。

show flex-content-filters

表 6-8 に、**show flex-content-filters** コマンド出力のフィールドを示します。

表 6-8　show flex-content-filters コマンドのフィールドの説明

フィールド	説明
Row	フレックスコンテンツ フィルタの優先順位を示します。
State	フィルタの状態（イネーブルまたはディセーブル）を示します。
Action	フィルタが特定のトラフィック タイプに対して実行するアクションを示します。
Protocol	フィルタが処理するトラフィックのプロトコル番号を指定します。
Port	フィルタが処理するトラフィックの宛先ポートを指定します。
Start	パケット ペイロードの先頭から、パターン マッチングを開始する位置までのオフセットを指定します（バイト単位）。このオフセットは、*pattern* フィールドに適用されます。

表 6-8　show flex-content-filters コマンドのフィールドの説明（続き）

フィールド	説明
End	パケット ペイロードの先頭から、パターン マッチングを終了する位置までのオフセットを指定します（バイト単位）。このオフセットは、*pattern* フィールドに適用されます。
Match-case	フィルタが一致するパターンで大文字と小文字が区別されるかどうかを示します（この引数は、*pattern* フィールドに適用されます）。 yes の場合は大文字と小文字が区別され、no の場合は区別されません。
TCPDump-expression	パケットと照合する式をバークリー パケット フィルタ形式で指定します。TCPDump 式の構文については、P.6-11 の「TCPDump 式の構文について」を参照してください。
Pattern-filter	パケット ペイロードと照合する正規表現のデータ パターンを指定します。パターン フィルタの構文については、P.6-14 の「パターン式の構文について」を参照してください。
RxRate (pps)	このフィルタについて測定されている現在のトラフィック レートを pps で示します。

フレックスコンテンツ フィルタの削除

フレックスコンテンツ フィルタを削除できます。また、フレックスコンテンツ フィルタをディセーブルにすることで、Detector モジュールがフィルタの式に基づいてパケットをフィルタリングしないようにすることもできます。詳細については、P.6-18 の「フレックスコンテンツ フィルタの状態の変更」を参照してください。

フレックスコンテンツ フィルタを削除するには、次の手順を実行します。

ステップ 1　フレックスコンテンツ　フィルタのリストを表示し、削除するフレックスコンテンツ フィルタの行番号を確認します。詳細については、P.6-15 の「フレックスコンテンツ フィルタの表示」を参照してください。

ステップ 2　フレックスコンテンツ フィルタを削除します。次のコマンドを入力します。

```
no flex-content-filter row-num
```

row-num 引数には、フレックスコンテンツ フィルタの行番号を指定します。すべてのフレックスコンテンツ フィルタを削除するには、アスタリスク（*）を入力します。

次の例を参考にしてください。

```
user@DETECTOR-conf-zone-scannet# no flex-content-filters 5
```

フレックスコンテンツ フィルタの状態の変更

フレックスコンテンツ フィルタの状態を変更できます。フレックスコンテンツ フィルタをディセーブルにすることで、Detector モジュールがフィルタの式に基づいてパケットをフィルタリングしないようにできます。フィルタは Detector モジュールのフレックスコンテンツ フィルタ リストに残ります。この方法で、Detector モジュールが特定の種類のトラフィックをフィルタリングしないようにできます。その後、特定のトラフィックが再びフィルタリングされるように Detector モジュールを設定できます。このとき、フィルタを設定し直す必要はありません。また、フレックスコンテンツ フィルタを削除することもできます。詳細については、P.6-17 の「フレックスコンテンツ フィルタの削除」を参照してください。

フレックスコンテンツ フィルタの状態を変更するには、次の手順を実行します。

ステップ 1　フレックスコンテンツ フィルタのリストを表示し、状態を変更するフレックスコンテンツ フィルタの行番号を確認します。詳細については、P.6-15 の「フレックスコンテンツ フィルタの表示」を参照してください。

ステップ 2　フレックスコンテンツ フィルタの状態を変更します。次のコマンドを入力します。

flex-content-filter *row-num* {**disabled** | **enabled**}

row-num 引数には、フレックスコンテンツ フィルタの行番号を指定します。

次の例を参考にしてください。

```
user@DETECTOR-conf-zone-scannet# flex-content-filters 5 disabled
```

バイパス フィルタの設定

バイパス フィルタは、Detector モジュールが特定のトラフィック フローを処理しないようにするために使用します。バイパス フィルタを設定すると、信頼されたトラフィックが Detector モジュールの検出メカニズムを通らないように誘導して、そのトラフィックを直接送信することができます。

この項では、次のトピックについて取り上げます。

バイパス フィルタの追加

バイパス フィルタを追加するには、関係するゾーンの設定モードで次のコマンドを入力します。

bypass-filter *row-num* ***src-ip*** [***ip-mask***] ***protocol dest-port*** [***fragments-type***]

表 6-9 に、**bypass-filter** コマンドの引数を示します。

表 6-9　bypass-filter コマンドの引数

パラメータ	説明
row-num	1 ～ 9999 の固有な番号を割り当てます。行番号はフィルタの ID で、これによって複数のバイパス フィルタの優先順位が定義されます。Detector モジュールは、行番号の昇順でフィルタを操作します。
フローの引数とキーワード	フィルタリングを実行する対象のフロー。*src-ip*、*ip-mask*、*protocol*、*dest-port*、および *fragments-type* の詳細については、表 6-1 を参照してください。

（注）　fragments-type と dest-port を両方指定することはできません。fragments-type を設定する場合は、dest-port に * を入力してください。

バイパス フィルタの表示

バイパス フィルタを表示するには、関連するゾーン設定モードで次のコマンドを入力します。

show bypass-filters

表 6-10 に、**show bypass-filters** コマンド出力のフィールドを示します。

表 6-10　show bypass-filters コマンドのフィールドの説明

フィールド	説明
Row	バイパス フィルタの優先順位を示します。
Filter flow	フィルタリングを実行する対象のフローを示します。Source IP、Source Mask、Proto、DPort、Frg の詳細については、表 6-2 を参照してください。
RxRate (pps)	このフィルタについて測定されている現在のトラフィック レートを pps で示します。

バイパス フィルタの削除

バイパス フィルタを削除するには、次の手順を実行します。

ステップ 1　バイパス フィルタのリストを表示し、削除するバイパス フィルタの行番号を確認します。詳細については、前の項、「バイパス フィルタの表示」を参照してください。

ステップ 2　バイパス フィルタを削除します。次のコマンドを入力します。

```
no bypass-filter row-num
```

row-num 引数には、バイパス フィルタの行番号を指定します。すべてのバイパス フィルタを削除するには、アスタリスク（*）を入力します。

次の例を参考にしてください。

```
user@DETECTOR-conf-zone-scannet# no bypass-filter 10
```

動的フィルタの設定

Detector は、トラフィック フローの分析結果として動的フィルタを作成します。このフィルタは、特定のトラフィック フローを関連する検出モジュールに誘導します。Detector モジュールは、この一連のフィルタを、ゾーンのトラフィックおよび特定の DDoS 攻撃に合せて継続的に調整します。動的フィルタは有効期間が限定されており、攻撃が終了すると消去されます。

Detector は、トラフィックの異常を検出する中で、ゾーンを宛先とするトラフィックを継続的に分析します。フローがポリシーのしきい値を超過すると、異常が発見されます。異常なトラフィック パターンを検出すると、攻撃の処理方法を定義する動的フィルタの作成を開始します。

動的フィルタは、Detector モジュールの syslog に通知レコードを作成するか、リモートの Guard をアクティブにしてゾーンを保護します。

ユーザは、動的フィルタにアクセスし、独自のニーズに合うように設定することができます。

この項では、次のトピックについて取り上げます。

動的フィルタの表示

Detector モジュールによって作成された動的フィルタを表示するには、**show dynamic-filters** コマンドを使用します。このコマンドには、次のオプションが用意されています。

- **show dynamic-filters** [details] : すべての動的フィルタのリストを表示します。
- **show dynamic-filters** *dynamic-filter-id* [details] : 特定の動的フィルタを表示します。
- **show dynamic-filters sort** {action | exp-time | id} : すべての動的フィルタのソートされたリストを表示します。

（注）　保留動的フィルタを表示するには、**show recommendations** コマンドを使用します。詳細については、第 8 章「インタラクティブ検出モード」を参照してください。

表 6-11 に、**show dynamic-filters** コマンドの引数を示します。

表 6-11　show dynamic-filters コマンドの引数

パラメータ	説明
dynamic-filter-id	表示する特定の動的フィルタの識別番号（ID）。この整数は Detector モジュールによって割り当てられます。フィルタの ID を確認するには、すべての動的フィルタのリストを表示します。
details	動的フィルタの詳細情報を表示します。詳細情報には、攻撃フローに関する追加情報、トリガーとなるレート、およびそのフィルタを作成したポリシーなどがあります。
action	動的フィルタをアクション別に表示します。
exp-time	動的フィルタを有効期限の昇順で表示します。
id	動的フィルタを ID 番号の昇順で表示します。

（注）　Detector モジュールは、最大 1,000 個の動的フィルタを表示します。1000 を越える動的フィルタがアクティブになっている場合は、ログ ファイルまたはゾーンのレポートで、動的フィルタに関するすべてのリストを確認してください。

次の例を参考にしてください。

```
user@DETECTOR-conf-zone-scannet# show dynamic-filters 876 details
```

表 6-12 に、**show dynamic-filters** コマンド出力のフィールドを示します。

表 6-12　show dynamic-filters コマンド出力のフィールドの説明

フィールド	説明
ID	フィルタの識別番号を示します。
Action	フィルタがトラフィック フローに対して実行するアクションを示します。
Exp Time	フィルタがアクティブになっている時間を示します。この時間が経過すると、フィルタは削除されます。
Filter flow	フィルタリングを実行する対象のフローを示します。Source IP、Source Mask、Proto、DPort、Frg の詳細については、表 6-2 を参照してください。
RxRate (pps)	このフィルタについて測定されている現在のトラフィック レートを pps で示します。

表 6-13 に、**show dynamic-filters details** コマンド出力の追加フィールドを示します。

表 6-13　show dynamic-filters details コマンドのフィールドの説明

フィールド	説明
Attack flow	攻撃フローの特性を示します。フロー フィールドの詳細については、表 6-2 を参照してください。
Triggering Rate	ポリシーのしきい値を超過した攻撃フローのレートを示します。
Threshold	攻撃フローによって超過したポリシーのしきい値を示します。
Policy	特定の動的フィルタを作成したポリシーを示します。詳細については、第 7 章「ポリシー テンプレートとポリシーの設定」を参照してください。

動的フィルタの削除

動的フィルタを削除することができます。ただし、削除が有効である期間は限られています。ゾーンの検出がイネーブルである限り、Detector モジュールが新しい動的フィルタを設定し続けるためです。

動的フィルタを削除するには、次の手順を実行します。

ステップ 1　動的フィルタのリストを表示し、削除する動的フィルタの ID を確認します。詳細については、前の項、「動的フィルタの表示」を参照してください。

ステップ 2　関連する動的フィルタを削除します。次のコマンドを入力します。

```
no dynamic-filter dynamic-filter-id
```

dynamic-filter-id 引数には、動的フィルタの ID を指定します。すべての動的フィルタを削除するには、アスタリスク（*）を入力します。

次の例を参考にしてください。

```
user@DETECTOR-conf-zone-scannet# no dynamic-filter 876
```

必要のない動的フィルタが再作成されないようにするには、次のアクションのいずれかを実行します。

- 動的フィルタを作成するポリシーを非アクティブにします（詳細については、P.7-22 の「ポリシーの状態の変更」を参照）。不要な動的フィルタを作成したポリシーを発見するには、P.6-21 の「動的フィルタの表示」を参照してください。
- 目的のトラフィック フロー用のバイパス フィルタを設定します（詳細については、P.6-19 の「バイパス フィルタの設定」を参照）。
- 不要な動的フィルタを作成したポリシーのしきい値を大きくします（詳細については、P.7-24 の「ポリシーのしきい値の設定」を参照）。