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。

この章は、次の項で構成されています。

概要

ゾーン フィルタは、Detector モジュールが特定のトラフィック フローを処理する方法を定義します。これらのフィルタを設定して、Detector モジュールがトラフィックの異常検出で使用する、トラフィックの機能をカスタマイズすることができます。

ゾーン フィルタを使用すると、Detector モジュールは次の機能を実行できます。

- 異常がないかゾーン トラフィックを分析する。
- トラフィックを直接ゾーンに転送し、Detector モジュールの異常検出機能をバイパスする。

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：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。

  信頼されたトラフィックが Detector モジュールの異常検出機能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。

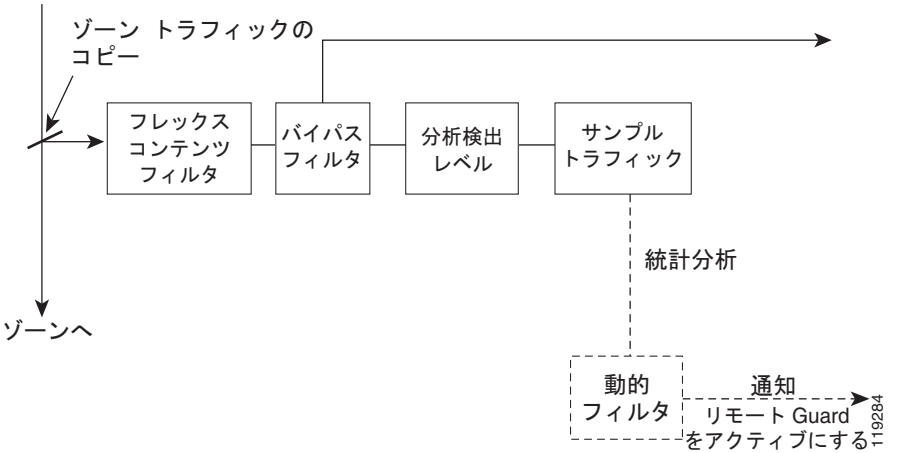
  詳細については、P.6-18 の「バイパス フィルタの設定」を参照してください。
- フレックスコンテンツ フィルタ：特定のトラフィック フローをカウントします。IP ヘッダーや TCP ヘッダー内のフィールドに基づいたフィルタリング、ペイロード コンテンツに基づいたフィルタリング、複雑なブール式に基づいたフィルタリングなどの非常に柔軟なフィルタリング機能があります。

  詳細については、P.6-4 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：必要な保護レベルを指定されたトラフィック　フローに適用する。Detector モジュールは、トラフィック フロー分析に基づいて動的フィルタを作成し、ゾーン トラフィックや DDoS 攻撃（分散型サービス拒絶攻撃）のタイプに合せて常に動的フィルタ セットを修正しています。動的フィルタは有効期間が限定されており、攻撃が終了すると削除されます。

  詳細については、P.6-22 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 モジュールのフィルタ システム



Detector モジュールは、この分析検出レベルをトラフィック フローに適用して、ゾーン トラフィックを分析します。バイパス フィルタを設定すると、Detector モジュールの検出機能をバイパスするように特定のフローを誘導できます。

トラフィック フローの統計分析を行うために、Detector モジュールには特定のタイプのトラフィックを処理する定義があります。この定義を、ゾーン ポリシーといいます。ゾーン ポリシーは、常にトラフィック フローを測定し、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（トラフィック フローがポリシーのしきい値を超えた）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。Detector モジュールがゾーンのトラフィックに異常を発見すると、イベントを自身の syslog に記録するか、新しいフィルタ（動的フィルタ）を作成します。動的フィルタが作成されると、リモート Guard リストに定義されている、ゾーンを保護する Cisco Anomaly Guard Module をアクティブにできます。

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 フィルタは、パケット ヘッダー内のフィールドまたはパケット ペイロードのパターンに基づいて、ゾーン トラフィックをフィルタリングします。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。これらのパターンでは、既知のワームまたは一定のパターンを持つフラッド攻撃が識別可能です。

（注） フレックスコンテンツ フィルタは大量の CPU リソースを消費します。フレックスコンテンツ フィルタは Detector モジュールのパフォーマンスに影響を及ぼす可能性があるため、使用を制限することをお勧めします。指定のポートに送信される TCP トラフィックなど、動的フィルタによって識別できる特定の攻撃からの検出にフレックスコンテンツ フィルタを使用する場合は、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、次の順序でフィルタリング基準を適用します。

1. プロトコルとポートのパラメータ値に基づいてパケットをフィルタリングする。
2. tcpdump 式の値に基づいてパケットをフィルタリングする。
3. 残りのパケットに対して pattern-expression の値を使用してパターン マッチングを実行する。

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。新しいフレックスコンテンツ フィルタを追加する場合は、リストの適切な位置に配置してください。

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　現在の行番号が連番の場合は、ゾーン設定モードで次のコマンドを入力して、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。

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

表 6-1 に、**flex-content-filter renumber** コマンドの引数を示します。

表 6-1　flex-content-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 ～ 9,999 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）進行中の攻撃や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には、**show packet-dump signatures** コマンドを使用して、Detector モジュールをアクティブにして攻撃のシグニチャを生成します。

詳細については、P.12-31 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　次のコマンドを入力して、新しいフレックスコンテンツ フィルタを追加します。

```
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 6-2 に、**flex-content-filter** コマンドの引数とキーワードを示します。

表 6-2　flex-content-filter コマンドの引数とキーワード

パラメータ	説明
row-num	1 ～ 9,999 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まります。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を監視しません。
enabled	フィルタの状態をイネーブルに設定します。Guard モジュールはトラフィックを監視し、フィルタに一致するフロー上でアクション（ドロップまたはカウント）を実行します。 これがデフォルトの状態です。
drop	フィルタに一致するフローをドロップします。Guard ゾーン テンプレートからゾーンを作成している場合は、Guard 設定モードにアクションのドロップを設定することができます。ドロップ アクションは Cisco Anomaly Guard Module にのみ適用されます。
count	フィルタに一致するフローをカウントします。
protocol	特定のプロトコルからのトラフィック。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 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

表 6-2　flex-content-filter コマンドの引数とキーワード（続き）

パラメータ	説明
port	特定の宛先ポート宛てのトラフィック。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）に設定する場合に、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 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/port-numbers
start-offset	パケット ペイロードの先頭から、*pattern-expression* 引数のパターン 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。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、*pattern-expression* 引数のパターン マッチングを終了する位置までのオフセット（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。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。

表 6-2　flex-content-filter コマンドの引数とキーワード（続き）

パラメータ	説明
tcpdump-expression	パケットと照合する式。式はバークリー パケット フィルタの形式です。詳細および設定例については、P.6-9 の「tcpdump 式の構文の設定」を参照してください。 式にスペースを使用する場合は、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。 式で引用符を使用するには、引用符（\”）の前にバックスラッシュ（\）をエスケープ文字として使用します。 **（注）** tcpdump 式の構文については、ヘルプを使用できません。
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。詳細については、P.6-13 の「パターン式構文の設定」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.12-31 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。 式にスペースを使用する場合は、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。 式で引用符を使用するには、引用符（\”）の前にバックスラッシュ（\）をエスケープ文字として使用します。 **（注）** 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.6-17 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。

次の例は、フレックスコンテンツ フィルタを設定する方法を示しています。

```
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"
```

この項では、次のトピックについて取り上げます。

- tcpdump 式の構文の設定
- パターン式構文の設定

tcpdump 式の構文の設定

tcpdump 式はバークリー パケット フィルタ形式で、パケットと照合する式を指定します。

（注） 宛先ポートとプロトコルに基づいてトラフィックをフィルタリングする場合は、tcpdump 式を使用できます。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。

式は 1 つ以上の要素からなります。通常、要素は ID（名前または番号）と ID の前に付く 1 つ以上の修飾子からなります。

修飾子には次の 3 つがあります。

- タイプ修飾子：ID（名前または番号）を定義します。指定可能なタイプは、**host**、**net**、および **port** です。**host** タイプ修飾子がデフォルトです。
- 方向修飾子：転送の方向を定義します。指定可能な方向は、**src**、**dst**、**src or dst**、および **src and dst** です。方向修飾子は **src or dst** がデフォルトです。

- プロトコル修飾子：特定のプロトコルへの照合を制限します。指定可能なプロトコルは **ether**、**ip**、**arp**、**rarp**、**tcp**、および **udp** です。プロトコル修飾子を指定しない場合、該当するタイプに適用されるすべてのプロトコルが照合されます。たとえば、ポート 53 とは、TCP または UDP のポート 53 を意味します。

表 6-3 に、tcpdump 式の要素を示します。

表 6-3　tcpdump 式の要素

要素	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。
ip multicast	マルチキャスト パケット。

表 6-3　tcpdump 式の要素（続き）

要素	説明
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。プロトコル名もキーワードです。プロトコル名を入力する場合は、プロトコル名の前にバックスラッシュ（\）をエスケープ文字として使用する必要があります。
expr relop expr	特定の式に適合するトラフィック。表 6-4 に、tcpdump 式の規則を示します。

表 6-4 に、tcpdump 式の規則を示します。

表 6-4　フレックスコンテンツ フィルタの式の規則

式の規則	説明
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。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* 値で指定します。 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*] *size* 引数はオプションで、フィールドのバイト数を指定します。この引数に可能な値は 1、2 または 4 です。デフォルトは 1 です。

次の方法により、式の要素を組み合せることができます。

- 要素と演算子の集まりを丸カッコで囲む：演算子は、通常のバイナリ演算子（+、-、*、/、&、|）と長さ演算子です。

（注）　式でカッコを使用するには、カッコの前にバックスラッシュをエスケープ文字として使用します（\()。

- 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、次のサイトにアクセスしてください。

http://www.freesoft.org/CIE/Topics/56.htm.

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 “”
```

パターン式構文の設定

パターン式とは、一連の文字を含んだ文字列を記述した正規表現です。パターン式は、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。パターン式は、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 文字がすべて含まれます。特殊文字は特殊な意味を持つ文字で、Detector モジュールがパターン式に対して実行するマッチングのタイプを指定します。フレックスコンテンツ フィルタは、パターン式とパケットの内容（パケット ペイロード）を照合します。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。

表 6-5 に、使用可能な特殊文字を示します。

表 6-5　パターン式で使用する特殊文字

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照合します。たとえば、goo.*s は、goos、goods、good for ddos などと照合します。
\	特殊文字から特別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、2 つのバックスラッシュ（\\）は 1 つのバックスラッシュ（\）と照合し、1 つのバックスラッシュとピリオド（\.）は 1 つのピリオドと照合します。 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 進値と照合します。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 進数の値は 2 桁である必要があります。たとえば、\x41 は 16 進数の値 A と照合します。

デフォルトでは、パターン式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を区別しないようにするには、**flex-content-filter** コマンドに **ignore-case** キーワードを指定します。詳細については、P.6-5 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 ペイロードに特殊なパターンを持つ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ました。*protocol*、*port*、および *tcpdump-expression* パラメータ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\xB91l
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 6-6 に、**show flex-content-filters** コマンド出力のフィールドを示します。

表 6-6　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指定します。
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（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。
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	フィルタと一致するパターン式で、大文字と小文字が区別されるのか、またはされないのかを指定します。 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 tcpdump 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.6-9 の「tcpdump 式の構文の設定」を参照してください。

表 6-6　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
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。パターン式の構文については、P.6-13 の「パターン式構文の設定」を参照してください。
RxRate (pps)	このフィルタで測定される現在のトラフィック レートをパケット / 秒で指定します。

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 フィルタを削除するか、フレックスコンテンツ フィルタをディセーブルにして Detector モジュールがフィルタ式に基づくパケットのフィルタリングをしないようにすることができます。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　**show flex-content-filters** コマンドを使用して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。

次の例は、フレックスコンテンツ フィルタのリストを表示する方法を示しています。

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

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　**no flex-content-filter** *row-num* コマンドを入力して、フレックスコンテンツ フィルタを削除します。

row-num 引数には、削除する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、*row-num* 引数としてアスタリスク（*****）を入力します。

次の例は、フレックスコンテンツ フィルタを削除する方法を示しています。

```
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 フィルタをディセーブルにすると、Detector モジュールはフィルタ式に基づくパケットのフィルタリングと、特定の種類のトラフィックのフィルタリングを実行しなくなります（フィルタはフレックスコンテンツ フィルタのリストに保持されます）。

その後、Detector モジュールが指定されたトラフィックをフィルタリングするように再設定できます（フィルタの再設定は不要）。あるいは、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.6-16 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　次のコマンドを入力して、フレックスコンテンツ フィルタの状態を変更します。

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。

次の例は、フレックスコンテンツ フィルタをディセーブルにする方法を示しています。

```
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、特定のトラフィック フローを Detector モジュールが処理しないようにするためのものです。バイパス フィルタを設定して、信頼できるトラフィックを誘導して Detector モジュールの検出機能を通らないように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 フィルタの表示
- バイパス フィルタの削除

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、ゾーン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 6-7 に、**bypass-filter** コマンドの引数を示します。

表 6-7　bypass-filter コマンドの引数

パラメータ	説明
row-num	1 ～ 9,999 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。
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。

表 6-7　bypass-filter コマンドの引数（続き）

パラメータ	説明
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 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/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。断片化には、次の 3 つのタイプがあります。 • **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。

（注）　fragments-type と dest-port を両方指定することはできません。fragments-type を設定するには、dest-port にアスタリスク（*）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、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 6-8 に、**show bypass-filters** コマンド出力のフィールドを示します。

表 6-8　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。
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。
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サブネット マスク。
Proto	フィルタが処理するトラフィックのプロトコル番号。
DPort	フィルタが処理するトラフィックの宛先ポート。
Frg	フィルタが処理する断片化の設定。 • 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：トラフィック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を処理します。
RxRate (pps)	このフィルタが測定する現在のトラフィック レート（パケット / 秒）。

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。

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　ゾーン設定モードで次のコマンドを入力して、バイパス フィルタを削除します。

no bypass-filter *row-num*

row-num 引数には、削除するバイパス 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。

次の例は、バイパス フィルタを削除する方法を示しています。

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

動的フィルタの設定

動的フィルタは必要な保護レベルをトラフィック フローに適用し、攻撃の処理方法を定義するものです。Detector モジュールは、ゾーン トラフィックに異常があると判断すると（フローがゾーン ポリシーのしきい値を超えたときに発生する）、動的フィルタを作成し、この動的フィルタ セットを常にゾーン トラフィックや DDoS 攻撃のタイプに適合させます。動的フィルタは有効期間が限定されており、攻撃が終了すると削除されます。Detector モジュールは、すべてのゾーンで同時にアクティブな動的フィルタを最大 150,000 個サポートします。

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。

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 モジュールが作成した動的フィルタを表示することができます。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 1 つ表示します。
- **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。

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。保留動的フィルタの詳細については、第 10 章「インタラクティブ検出モードの使用方法」を参照してください。

表 6-9 に、**show dynamic-filters** コマンドの引数とキーワードを示します。

表 6-9　show dynamic-filters コマンドの引数とキーワード

パラメータ	説明
dynamic-filter-id	表示する特定の動的フィルタの ID[1]。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、動的フィルタの完全なリストを表示します。
details	動的フィルタを詳細に表示します。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。

1. ID = 識別番号

（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。1,000 を超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。

次の例は、動的フィルタを詳細に表示する方法を示しています。

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

表 6-10 に、**show dynamic-filters** コマンドの出力フィールドを示します。

表 6-10　show dynamic-filters コマンドの出力フィールドの説明

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。

表 6-10　show dynamic-filters コマンドの出力フィールドの説明（続き）

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。
RxRate (pps)	このフィルタで測定される現在のトラフィック レートを pps で指定します。
Destination IP	フィルタが処理するトラフィックの宛先 IP アドレス。 Detector モジュールは、宛先 IP アドレスおよびゾーンに設定されている **protect-ip-state** の値に基いて Cisco Anomaly Guard Module の保護をアクティブにします。

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。

表 6-11 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 6-11　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。攻撃フローは、Source IP、Source Mask、Proto、Dport、および Frg フィールドで構成されています。これらのフィールドについては、表 6-10 で説明しています。
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	動的フィルタを作成したポリシーを指定します。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。

動的フィルタの追加

ゾーンの攻撃中に、動的フィルタを追加してゾーン異常検出を操作することができます。リモートの Guard リスト（リモート Gurard）に定義されている Cisco Anomaly Guard Module をアクティブにしてゾーンを保護するように動的フィルタを設定することができます。動的フィルタの宛先 IP アドレスが、**protect-ip-state** コマンドとゾーンのアドレス範囲を使用してゾーンに定義した Guard 保護アクティベーション方式と一致しないと、リモート アクティベーションは失敗します。リモート Guard のゾーン保護をアクティブ化するように動的フィルタを設定するには、次のいずれかの方法で実行します。

- ゾーン全体に対してリモート Guard のゾーン保護をアクティブ化する : ゾーン全体でゾーン保護をアクティブ化するには、*dst-ip* 引数を入力しないようにします。ゾーンの Guard 保護アクティベーション方式は、**entire-zone** または **policy-type** に設定する必要があります。
- ゾーン IP アドレス範囲にある特定の IP アドレスに対してのみリモート Guard のゾーン保護をアクティブ化する : 特定の IP アドレスのゾーン保護をアクティブ化するには、*dst-ip* 引数を使用して特定の IP アドレスを指定します。ゾーンの Guard 保護アクティベーション方式は、**dst-ip-by-name** に設定する必要があります。

詳細については、P.9-7 の「ゾーンを保護するためのリモート Guard のアクティブ化」および P.9-4 の「Guard 保護のアクティベーション方式の設定」を参照してください。

動的フィルタを追加するには、ゾーン設定モードで次のコマンドを入力します。

dynamic-filter remote-activate {*exp-time* | **forever**} [*dst-ip*]

複数の動的フィルタを追加するには、**dynamic-filter** コマンドを複数使用します。

表 6-12 に、**dynamic-filter** コマンドの引数を示します。

表 6-12　dynamic-filter コマンドの引数とキーワード

パラメータ	説明
remote-activate	ゾーンを保護するためのリモート Guards をアクティブ化します。 引数 *dst-ip* を入力しないと、リモート Guard の保護をアクティブにするために Detector モジュールが使用するアクティベーション方式は **entire-zone** になります。
exp-time	フィルタがアクティブである期間（秒単位）を指定する、1 ～ 3,000,000 の整数。
forever	フィルタを無期限でアクティブにします。保護が終了すると、フィルタは削除されます。
dst-ip	特定の IP アドレスまでのトラフィック。Detector モジュールは特定の IP アドレスに基いてリモート Guard をアクティブにしてゾーンを保護します。IP アドレスをドット区切り 10 進表記で入力します（たとえば 192.168.100.1）。 Detector モジュールは、アクティベーション方式 dst-ip-by-name を使用してリモート Guard の保護をアクティブにします。

次の例では、ゾーン全体のリモート Gurad の保護をアクティブにする動的フィルタの追加方法を示しています。

```
admin@GUARD-conf-zone-scannet# dynamic-filter remote-activate 600
```

動的フィルタの削除

動的フィルタを削除しても、削除が有効になっている期間は限られています。これは、ゾーン異常検出がイネーブルのときは、Detector モジュールが新しい動的フィルタの設定を続行するためです。Detector モジュールが動的フィルタを作成しないようにする方法の詳細については、P.6-28 の「動的フィルタの作成防止」を参照してください。

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　ゾーン設定モードで次のコマンドを入力して、関連する動的フィルタを削除します。

```
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（*）を使用します。

次の例は、動的フィルタを削除する方法を示しています。

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

動的フィルタの作成防止

不要な動的フィルタが作成されないようにするには、次のいずれかのアクションを実行します。

- 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特定するには、P.6-22 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 フローにバイパス フィルタを設定します。詳細については、P.6-18 の「バイパス フィルタの設定」を参照してください。
- 不要な動的フィルタを作成するポリシーのしきい値を増分します。詳細については、P.7-23 の「ポリシーのしきい値の設定」を参照してください。