メッセージ フィルタを使用した電子メール ポリシーの適用

Cisco アプライアンスは、詳細なコンテンツ スキャンおよびメッセージ フィルタリング テクノロジーを備えているため、会社のネットワークに参加または退出するときに、会社のポリシーを適用して、特定のメッセージを処理することができます。

この章では、ポリシーの適用のために使用可能な機能（コンテンツ スキャン エンジン、メッセージ フィルタ、添付ファイル フィルタ、コンテンツ ディクショナリ）の強力な組み合わせ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 （1 ページ）
- メッセージ フィルタのコンポーネント （2 ページ）
- メッセージ フィルタの処理 （4 ページ）
- メッセージ フィルタ ルール （11 ページ）
- メッセージ フィルタ アクション （67 ページ）
- 添付ファイルのスキャン （105 ページ）
- メッセージフィルタを使用した、メッセージの添付ファイルの悪意のあるファイルの検出 （117 ページ）
- CLI を使用したメッセージ フィルタの管理 （117 ページ）
- メッセージ フィルタの例 （133 ページ）
- スキャン動作の設定 （141 ページ）

概要

メッセージ フィルタにより、Cisco アプライアンスでメッセージを受信したときに、それらを処理する方法を記述した特別なルールを作成できます。メッセージフィルタは、特定の種類の電子メール メッセージに指定の特別な処理を施す必要があることを指定します。Cisco メッセージフィルタは、指定の単語に対してメッセージ内容をスキャンすることによって社内メール ポリシーを適用することができます。この章は、次の項で構成されています。

- **メッセージ フィルタのコンポーネント**。メッセージ フィルタにより、メッセージの受信時にそれらを処理する方法を記述した特別なルールを作成できます。フィルタ ルールでは、メッセージまたは添付ファイルの内容、ネットワークに関する情報、メッセージエンベロープ、メッセージヘッダー、またはメッセージ本文に基づいてメッセージを識別します。フィルタアクションにより、通知を生成したり、メッセージのドロップ、バウンス、アーカイブ、ブラインド カーボン コピー、変更を行ったりすることができます。詳細については、メッセージ フィルタのコンポーネント （2 ページ）を参照してください。
- **メッセージ フィルタの処理**。AsyncOS がメッセージ フィルタを処理する場合、AsyncOS がスキャンする内容、処理の順番、実行されるアクションは、メッセージ フィルタの順番、メッセージの内容を変更した可能性のある事前の処理、メッセージの MIME 構造、コンテンツマッチング用に設定されたしきい値スコア、クエリーの構造などのいくつかの要因に基づきます。詳細については、メッセージ フィルタの処理 （4 ページ）を参照してください。
- **メッセージ フィルタ ルール**。各フィルタには、フィルタで処理できる一連のメッセージを定義するルールがあります。メッセージフィルタを作成する場合、それらのルールを定義します。詳細については、メッセージ フィルタ ルール （3 ページ）を参照してください。
- **メッセージ フィルタ アクション**。各フィルタには、ルールで true に評価された場合に、メッセージに対して実行するアクションがあります。実行できるアクションには、最終アクション（メッセージの配信、ドロップ、バウンスなど）、またはメッセージをさらに処理できる非最終アクション（ヘッダーの除去や挿入など）の 2 つのタイプのアクションがあります。詳細については、メッセージ フィルタ アクション （3 ページ）を参照してください。
- **添付ファイル スキャン メッセージ フィルタ**。添付ファイル スキャン メッセージ フィルタを使用して、会社のポリシーと整合しないメッセージから添付ファイルを除去できます。元のメッセージはそのまま配信することができます。添付ファイルは、それらの特定のタイプ、フィンガープリント、内容に基づいてフィルタできます。イメージアナライザを使用して、イメージ添付ファイルをスキャンすることもできます。イメージアナライザは、肌の色、本文サイズ、曲率を測定して、グラフィックに不適切な内容が含まれている可能性を判断するアルゴリズムを作成します。詳細については、添付ファイルのスキャン （105 ページ）を参照してください。
- **CLI を使用したメッセージ フィルタの管理**。CLI は、メッセージ フィルタを操作するためのコマンドを受け入れます。たとえば、メッセージ フィルタのリストを表示、並び替え、インポート、エクスポートする必要がある場合があります。詳細については、CLI を使用したメッセージ フィルタの管理 （117 ページ）を参照してください。
- **メッセージフィルタの例**。この項では、実際のフィルタの例を示し、各フィルタについて簡単に説明します。詳細については、メッセージフィルタの例 （133 ページ）を参照してください。

メッセージ フィルタのコンポーネント

メッセージ フィルタにより、メッセージの受信時にそれらを処理する方法を記述した特別なルールを作成できます。メッセージ フィルタは、メッセージ フィルタ ルールとメッセージ フィルタ アクションから構成されます。

関連項目

- メッセージ フィルタ ルール （3 ページ）
- メッセージ フィルタ アクション （3 ページ）
- メッセージ フィルタの構文例 （3 ページ）

メッセージ フィルタ ルール

メッセージ フィルタ ルールによって、フィルタで処理するメッセージを判断します。ルールは、論理結合子 AND、OR、NOT を使用して組み合わせることで、複雑なテストを作成できます。ルール式は、かっこを使用してグループ化することもできます。

メッセージ フィルタ アクション

メッセージ フィルタの目的は、選択されたメッセージに対してアクションを実行することです。

アクションには、次の 2 つのタイプがあります。

- 最終アクション（deliver、drop、bounce など）はメッセージの処理を終了し、後続のフィルタによるさらなる処理を許可しません。
- 非最終アクションは、メッセージをさらに処理することを許可するアクションを実行します。

（注） 非最終メッセージ フィルタ アクションは、累積的です。各フィルタが異なるアクションを指定する複数のフィルタにメッセージが一致する場合、すべてのアクションが累積され、適用されます。ただし、同じアクションを指定する複数のフィルタにメッセージが一致する場合、前のアクションが上書きされ、最後のフィルタ アクションが適用されます。

関連項目

- フィルタ アクション一覧表 （68 ページ）
- アクション変数 （80 ページ）
- 一致した内容の表示 （83 ページ）
- メッセージ フィルタ アクションの説明と例 （84 ページ）

メッセージ フィルタの構文例

フィルタ仕様の直観的な意味は次のようになります。

メッセージがルールに一致する場合、順番にアクションが適用されます。else 句が存在する場合、メッセージがルールに一致しない場合に else 句内のアクションが実行されます。

指定したフィルタ名によって、フィルタをアクティブ、非アクティブ、削除する場合に、フィルタが管理しやすくなります。

メッセージ フィルタでは次の構文を使用します。

構文例	目的
expedite:	フィルタ名
if (recv-listener == 'InboundMail' or recv-int == 'notmain')	ルールの指定
`{ alt-src-host('outbound1'); skip-filters(); }`	アクションの指定
`else { alt-src-host('outbound2'); }`	（任意）代替アクションの指定

代替アクションは省略できることに注意してください。

構文例	目的
expedite2:	フィルタ名
`if ((not (recv-listener == 'InboundMail')) and` (not (recv-int == 'notmain')))	ルールの指定
`{ alt-src-host('outbound2'); skip-filters();` }	アクションの指定

複数のフィルタを順番に1つずつ並べて1つのテキストファイルにまとめることができます。

単一引用符または二重引用符で、フィルタの値を囲む必要があります。単一引用符または二重引用符は、値の両側に等しく組み合わせる必要があります。たとえば、式 `notify('customercare@example.com')` と `notify("customercare@example.com")` はどちらも有効ですが、式 `notify("customercare@example.com')` は構文エラーが発生します。

「#」文字で始まる行はコメントと見なされ、無視されます。ただし、それらは `filters -> detail` によってフィルタを表示して確認できるため、AsyncOS では保持されません。

メッセージ フィルタの処理

AsyncOSはメッセージフィルタを処理する場合、AsyncOSがスキャンする内容、処理の順番、実行するアクションは、次のいくつかの要因に基づきます。

- **メッセージ フィルタの順番。**メッセージ フィルタは、順序付けられたリストで維持されます。メッセージの処理時に、AsyncOS は各メッセージ フィルタをそれらがリストに表示されている順番で適用します。最終アクションが行われた場合、そのメッセージに対して、それ以上のアクションは実行されません。詳細については、メッセージフィルタの順番 （5 ページ）を参照してください。
- **事前処理。**メッセージ フィルタが評価される前に、AsyncOS メッセージに対して実行されるアクションによって、ヘッダーが追加または削除されることがあります。AsyncOS は、処理時にメッセージに存在するヘッダーに対してメッセージ フィルタ プロセスを実行します。詳細については、メッセージ ヘッダー ルールおよび評価 （6 ページ）を参照してください。
- **メッセージの MIME 構造。**メッセージの MIME 構造によって、「本文」として扱われるメッセージの部分と「添付ファイル」として扱われるメッセージの部分が判断されます。多くのメッセージフィルタは、メッセージの本文部分のみに、または添付ファイル部分のみに作用するように設定されます。詳細については、メッセージ本文とメッセージ添付ファイル （6 ページ）を参照してください。
- **正規表現に設定されるしきい値スコア。**正規表現に一致させる場合、フィルタ アクションが実行されるまでに、一致が発生しなければならない回数を集計する「スコア」を設定します。これにより、さまざまな用語に対する応答の重み付けをすることができます。詳細については、コンテンツ スキャンの一致のしきい値 （7 ページ）を参照してください。
- **クエリーの構造。**メッセージ フィルタ内で、AND または OR テストを評価する場合、AsyncOS は不要なテストを評価しません。さらに、システムは左から右にテストを評価しないことに注意することが重要です。代わりに、AND および OR テストが評価される場合、最も価値の低いテストが最初に評価されます。詳細については、メッセージフィルタ内の AND テストと OR テスト （10 ページ）を参照してください。

関連項目

- メッセージ フィルタの順番 （5 ページ）
- メッセージ ヘッダー ルールおよび評価 （6 ページ）
- メッセージ本文とメッセージ添付ファイル （6 ページ）
- コンテンツ スキャンの一致のしきい値 （7 ページ）
- メッセージ フィルタ内の AND テストと OR テスト （10 ページ）

メッセージ フィルタの順番

メッセージフィルタは順序付けられたリストに維持され、リスト内のそれらの位置によって番号付けされます。メッセージの処理時に、メッセージフィルタが割り振られた番号順で適用されます。そのため、9 番のフィルタがメッセージに対してすでに最終アクション（バウンスなど）を実行した場合、30 番のフィルタは、メッセージの送信元ホストを変更する機会がありません。リストのフィルタの位置は、システムユーザインターフェイスによって変更できます。ファイルからインポートされたフィルタは、インポートされたファイル内のそれらの相対的順序に基づきます。

最終アクション後、そのメッセージに対して、それ以上のアクションは実行されません。

メッセージがフィルタ ルールに一致していても、次のいずれかの理由で、フィルタがそのメッセージに対して作用しないことがあります。

- フィルタが非アクティブである。
- フィルタが無効である。
- フィルタが、メッセージの最終アクションを実行した前のフィルタに取って代わられた。

メッセージ ヘッダー ルールおよび評価

フィルタは、ヘッダー ルールを適用する場合に、元のメッセージのヘッダーではなく、「処理済み」ヘッダーを評価します。つまり、

- 前に実行されたアクションによって、ヘッダーが追加された場合、後続のすべてのヘッダー ルールによって、それを照合できるようになります。
- 前に実行されたアクションによって、ヘッダーが取り除かれた場合、後続のすべてのヘッダー ルールで、それを照合できなくなります。
- 前に実行されたアクションによって、ヘッダーが変更された場合、後続のすべてのヘッダー ルールで、元のメッセージ ヘッダーではなく、変更済みのヘッダーが評価されます。

この動作は、メッセージ フィルタとコンテンツ フィルタの両方に共通です。

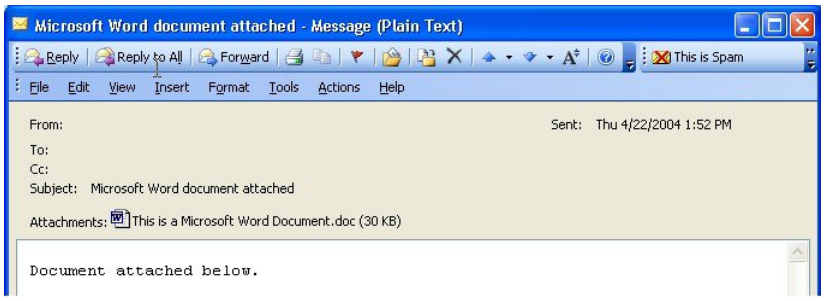
メッセージ本文とメッセージ添付ファイル

電子メール メッセージは、複数の部分から構成されます。RFC では、メッセージのヘッダーの後に続くすべてのものをマルチパート「メッセージ本文」として規定していますが、多くのユーザはまだメッセージの「本文」と「添付ファイル」を別々のものと捉えています。body-variable または `attachment`-variable という Cisco メッセージ フィルタを使用する場合、Cisco アプライアンスは、ほとんどのユーザが「本文」と「添付ファイル」として考える部分を、多くの MUA がそれらを別々にレンダリングしようと試みるのと同じように区別しようとします。

body-*variable* または `attachment`-*variable* メッセージ フィルタ ルールを書く目的では、メッセージ ヘッダーの後のすべてのものがメッセージ本文と見なされ、その内容は本文内にある MIME 部分の最初のテキスト部分と見なされます。そのコンテンツの後のすべてのもの（つまり、追加の MIME 部分）は添付ファイルと見なされます。AsyncOS はメッセージのさまざまな MIME 部分を評価し、添付ファイルとして処理されるファイルの部分を識別します。

たとえば、 以下の図は、Microsoft Outlook MUA のメッセージを示しています。ここでは「`Document attached below.`」という言葉がプレーン テキストのメッセージ本文として表示され、ドキュメント「`This is a Microsoft Word document.doc`」が添付ファイルとして表示されています。多くのユーザが電子メールをこのように捉えている（最初の部分がプレーン テキストで 2 番めの部分がバイナリ ファイルであるマルチパート メッセージとしてではなく）ため、Cisco は、メッセージの「本文」（最初のプレーン テキスト部分）と対照的に、.doc ファイル部分（実質的に 2 番めの MIME 部分）を区別して処理するルールを作成するために、メッセージ フィルタで「添付ファイル」という用語を使用しています。ただし、RFCS 1521 および 1522 で使われている用語によると、メッセージの本文はすべての MIME 部分から構成されます。

図 *1:*「添付ファイル」を含むメッセージ



Cisco アプライアンスは、マルチパート メッセージの本文と添付ファイルを区別しているため、想定される動作をするためには、body-variable または `attachment`-variable メッセージ フィルタ ルールを使用する場合に、いくつかのケースで注意が必要です。

- テキスト部分が 1 つのメッセージ（つまり、「Content-Type: text/plain」または「Content-Type: text/html」のヘッダーを含むメッセージ）がある場合、Cisco アプライアンスはメッセージ全体を本文と見なします。コンテンツ タイプが異なる場合、Cisco アプライアンスは、それを単一の添付ファイルと見なします。
- エンコードされたファイル（uuencoded など）は電子メール メッセージの本文に含まれます。これが発生した場合、エンコードされたファイルは添付ファイルとして扱われ、抽出およびスキャンされ、残りのテキストがテキスト本文として見なされます。
- 単一のテキスト以外の部分は常に添付ファイルと見なされます。 たとえば、.zip ファイルのみで構成されるメッセージは、添付ファイルと見なされます。

コンテンツ スキャンの一致のしきい値

メッセージ本文または添付ファイル内のパターンを検索するフィルタ ルールを追加する場合、パターンが見つかる必要がある回数の最初のしきい値を指定できます。AsyncOS はメッセージをスキャンすると、メッセージおよび添付ファイルに見つかった一致の数の「スコア」を集計します。最小しきい値に満たない場合、正規表現は true と評価されません。このしきい値は次のフィルタ ルールに指定できます。

- body-contains
- only-body-contains
- attachment-contains
- every-attachment-contains
- dictionary-match
- attachment-dictionary-match

`drop-attachments-where-contains` アクションにしきい値を指定することもできます。

（注） ヘッダーまたはエンベロープの受信者と送信者をスキャンするフィルタ ルールにしきい値を指定できません。

関連項目

- しきい値の構文 （8 ページ）
- メッセージ本文と添付ファイルのしきい値スコア （8 ページ）
- しきい値スコアリング マルチパート/代替 MIME 部分 （8 ページ）
- コンテンツ ディクショナリを使用したしきい値のスコアリング （9 ページ）

しきい値の構文

出現最小回数のしきい値を指定するには、パターンと、true と評価するために必要な一致の最小数を指定します。

```
if(<filter rule>('<pattern>',<minimum threshold>)){
```

たとえば、body-contains フィルタ ルールで、値「Company Confidential」が少なくとも 2 回見つかる必要があることを指定するには、次の構文を使用します。

```
if(body-contains('Company Confidential',2)){
```

デフォルトで、AsyncOS がコンテンツ スキャン フィルタを保存する場合、フィルタをコンパイルし、しきい値が割り当てられていない場合、1 のしきい値を割り当てます。

コンテンツ ディクショナリの値に対して、パターン マッチの最小数を指定することもできます。コンテンツ ディクショナリの詳細については、「テキスト リソース」の章を参照してください。

メッセージ本文と添付ファイルのしきい値スコア

電子メール メッセージは、複数の部分から構成されることがあります。メッセージ本文または添付ファイル内のパターンを検索するフィルタ ルールのしきい値を指定すると、AsyncOS は、メッセージ部分と添付ファイルの一致の数をカウントして、しきい値「スコア」を判断します。メッセージ フィルタで特定の MIME 部分を指定しない限り（`attachment-contains` フィルタ ルールなど）、AsyncOS はメッセージのすべての部分で見つかった一致を合計し、一致の合計がしきい値に達しているかどうかを判断します。たとえば、しきい値が 2 の `body-contains` メッセージ フィルタがあるとします。本文に 1 つの一致があり、添付ファイルに 1 つの一致があるメッセージを受信します。AsyncOS がこのメッセージを採点した場合、合計が 2 つの一致になり、しきい値スコアを満たしていると判断します。

同様に、複数の添付ファイルがある場合、AsyncOS は添付ファイルごとにスコアを合計して、一致のスコアを判断します。たとえば、しきい値が 3 の `attachment-contains` フィルタ ルールがあるとします。2 つの添付ファイルがあるメッセージを受信し、各添付ファイルに 2 つの一致が含まれます。AsyncOS はこのメッセージを 4 つの一致と採点し、しきい値スコアが満たされていると判断します。

しきい値スコアリング マルチパート/代替 MIME 部分

カウントの重複を避けるため、同じコンテンツの 2 つの表現（プレーン テキストと HTML）がある場合、AsyncOS は重複した部分からの一致を合計しません。代わりに、各部分の一致を

比較して、最高値を選択します。AsyncOS はこの値をマルチパート メッセージの他の部分からのスコアに追加して、合計スコアを作成します。

たとえば、`body-contains` フィルタ ルールを設定し、しきい値を 4 に設定します。プレーン テキスト、HTML、および 2 つの添付ファイルを含むメッセージを受信します。メッセージは次のような構造を使用します。

```
multipart/mixed

        multipart/alternative

                text/plain

                text/html

        application/octet-stream

        application/octet-stream
```

`body-contains` フィルタ ルールは、メッセージの text/plain および text/html 部分を最初に採点して、このメッセージのスコアを判断します。次に、これらのスコアの結果を比較し、結果から最高のスコアを選択します。さらに、この結果を各添付ファイルからのスコアに追加して、最終スコアを判断します。メッセージに次の数の一致があるとします。

```
multipart/mixed

        multipart/alternative

                text/plain (2 matches)

                text/html (2 matches)

        application/octet-stream (1 match)

        application/octet-stream
```

AsyncOS は text/plain と text/html 部分の一致を比較するため、スコア 3 を返します。これは、フィルタ ルールをトリガーする最小しきい値を満たしていません。

コンテンツ ディクショナリを使用したしきい値のスコアリング

コンテンツディクショナリを使用すると、用語の「重み」を設定して、より簡単に特定の用語でフィルタ アクションをトリガーできます。たとえば、「bank」という用語ではメッセージ フィルタをトリガーせず、「bank」の後に「account」という用語があり、さらに ABA ルーティング番号が含まれていれば、フィルタ アクションをトリガーする必要があるとします。これを実現するには、重みを設定したディクショナリを使用して、特定の用語または用語の組み合わせの重要度を高くします。コンテンツ ディクショナリを使うメッセージ フィルタがフィルタ ルールの一致を評価する場合、コンテンツディクショナリの重みを使用して最終的なスコアを

決定します。たとえば、次のコンテンツと重みを指定してコンテンツディクショナリを作成したとします。

表 *1*:コンテンツ ディクショナリの例

用語/スマート ID	Weight
ABA 送金番号	3
アカウント（Account）	2
バンク	1

このコンテンツ ディクショナリを `dictionary-match` または `attachment-dictionary-match` メッセージ フィルタ ルールに関連付けると、AsyncOS はメッセージ内で検出された一致する用語の各インスタンスの合計「スコア」に、この用語の重みを追加します。たとえば、メッセージ本文に用語「account」のインスタンスが 3 つ含まれているメッセージの合計スコアに、値 6 が追加されます。メッセージ フィルタのしきい値が 6 に設定されている場合、AsyncOS はこのしきい値スコアが満たされたと判断します。または、各用語のインスタンスが 1 つずつ含まれている場合も合計値は 6 になり、このスコアによってフィルタ アクションがトリガーされます。

メッセージ フィルタ内の AND テストと OR テスト

メッセージ フィルタ内で、AND または OR テストを評価する場合、AsyncOS は不要なテストを評価しません。したがって、たとえば、一方の AND テストが false の場合、もう一方のテストは評価されません。テストは左から右に評価されるわけではないため、注意してください。代わりに、AND および OR テストが評価される場合、最も価値の低いテストが最初に評価されます。たとえば、次のフィルタでは、`rcpt-to-group` テストよりも消費リソースの少ない `remote-ip` テストが必ず最初に評価されます（一般に、LDAP テストの方が消費リソースは高くなります）。

```
andTestFilter:

if (remote-ip == "192.168.100.100" AND rcpt-to-group == "GROUP")

  { ... }
```

最もコストの低いテストが最初に実行されるため、項目の順序を入れ替えても影響はありません。テストの実行順序を保証する必要がある場合は、if 文をネストさせてください。この方法は、できる限りコストの高いテストを避けるためにも推奨します。

```
expensiveAvoid:

if (<simple tests>)

  { if (<expensive test>)
```

```
    { <action> }

  }
```

次に、もう少し複雑な例で説明します。

```
if (test1 AND test2 AND test3) { ... }
```

システムは左から右に式をグループ化するため、次のようになります。

```
if ((test1 AND test2) AND test3) { ... }
```

この場合、システムが最初に行うのは、(`test1` AND `test2`) のコストと `test3` のコストの比較です。最初に 2 番めの AND を評価します。3 つのテストすべてで同じコストがかかる場合、`test3` が最初に実行されます。これは、(`test1` AND `test2`) のコストが 2 倍になるためです。

メッセージ フィルタ ルール

各メッセージフィルタには、フィルタを適用できるメッセージのコレクションを定義するルールが含まれています。フィルタ ルールを定義して、true を返すメッセージへのフィルタ アクションを定義します。

関連項目

- フィルタ ルールの概要の表 （11 ページ）
- ルールで使用する正規表現 （27 ページ）
- スマート ID （32 ページ）
- メッセージ フィルタ アクションの説明と例 （84 ページ）

フィルタ ルールの概要の表

次の表 に、メッセージ フィルタで使用できるルールをまとめます。

表 *2:* メッセージ フィルタ ルール

ルール（Rule）	構文	説明
件名ヘッダー（Subject Header）	`subject`	件名ヘッダーが特定のパターンと一致しているか。subject ルール （35 ページ） を参照してください。
本文サイズ（Body Size）	`body-size`	本文のサイズは一定の範囲内か。本文サイズ ルール （38 ページ） を参照してください。

ルール（Rule）	構文	説明
エンベロープ送信者（Envelope Sender）	`mail-from`	エンベロープ送信者（Envelope From, <MAIL FROM>）が指定したパターンと一致しているか。エンベロープ送信者ルール （37ページ）を参照してください。
グループ内のエンベロープ送信者（Envelope Sender in Group）	`mail-from-group`	エンベロープ送信者（Envelope From <MAIL FROM>）が、指定した LDAP グループ内に存在するか。グループ内エンベロープ送信者ルール （37ページ）を参照してください。
送信者グループ（Sender Group）	`sendergroup`	どの送信者グループが、リスナーのホストアクセステーブル（HAT）に一致するか。送信者グループ ルール （37ページ）を参照してください。
グループ内エンベロープ（Envelope Recipient）	`rcpt-to`	エンベロープ受信者（Envelope To, <RCPT TO>）が指定したパターンと一致しているか。エンベロープ受信者ルール （36ページ）を参照してください。 （注） `rcpt-to` ルールはメッセージに基づいています。メッセージに複数の受信者が設定されている場合、いずれか 1 人の受信者がルールと一致していれば、指定した処理がすべての受信者に対するメッセージに適用されます。

ルール（Rule）	構文	説明
グループ内エンベロープ受信者（Envelope Recipient in Group）	`rcpt-to-group`	エンベロープ受信者（Envelope To, <RCPT TO>）が、指定したLDAP グループ内に存在するか。グループ内エンベロープ受信者ルール （36ページ）を参照してください。 （注） `rcpt-to-group` ルールはメッセージに基づいています。メッセージに複数の受信者がある場合、グループの受信者が 1 人でも検出されれば、rcptthe により指定されたアクションがメッセージのすべての受信者に適用されます。
リモートIP	`remote-ip`	リモート ホストから送信されたメッセージは、指定した IP アドレスまたは IP ブロックに一致しているか。リモート IP ルール （39ページ）を参照してください。
受信インターフェイス（Receiving Interface）	`recv-int`	メッセージは、指定された受信インターフェイス経由で届いたか。参照先受信 IP インターフェイス ルール （39ページ）
受信リスナー（Receiving Listener）	`recv-listener`	メッセージは、指定されたリスナー経由で届いたか。受信リスナー ルール （39ページ）を参照してください。
日付（Date）	`date`	現在時刻は特定の日時の前か後か。日付ルール （40ページ）を参照してください。

ルール（**Rule**）	構文	説明
ヘッダー（Header）	`header(<string>)`	メッセージに特定のヘッダーが含まれているか。ヘッダーの値が特定のパターンと一致しているか。ヘッダー ルール（40 ページ）を参照してください。
ランダム（Random）	`random(<integer>)`	ランダム番号は一定の範囲内か。乱数ルール（41 ページ）を参照してください。
受信者数（Recipient Count）	`rcpt-count`	この電子メールの受信者の人数。受信者数ルール（42 ページ）を参照してください。
アドレス数（Address Count）	`addr-count()`	受信者の累積数。 このフィルタは、エンベロープの受信者ではなくメッセージ本文のヘッダーに対して機能する点が rcpt-count フィルタ ルールと異なります。アドレス数ルール（42 ページ）を参照してください。
SPFステータス（SPF Status）	`spf-status`	SPF 検証ステータスを判別します。このフィルタ ルールでは、さまざまな SPF 検証結果をクエリできます。有効な SPF/SIDF 戻り値ごとに異なるアクションを入力できます。SPF-Status ルール（49 ページ）を参照してください。
SPF合格（SPF Passed）	`spf-passed`	SPF/SIDF 検証に合格したか。このフィルタ ルールは SPF/SIDF 結果をブール値として一般化します。SPF-Passed ルール（50 ページ）を参照してください。

ルール（Rule）	構文	説明
S/MIME ゲートウェイ メッセージ（S/MIME Gateway Message）	`smime-gateway`	メッセージは S/MIME 署名されているか、暗号化されているか、または署名および暗号化されているか。S/MIME ゲートウェイ メッセージ ルール（51 ページ）を参照してください
S/MIME ゲートウェイ検証済（S/MIME Gateway Verified）	`smime-gateway-verified`	S/MIME メッセージは正常に検証されているか、復号されているか、または復号および検証されているか。S/MIME ゲートウェイ検証済みルール（51 ページ）を参照してください。
イメージ評価（Image verdict）	`image-verdict`	イメージ スキャンの評価の結果。このフィルタ ルールを使用して、さまざまなイメージ分析の評価について問い合わせることができます。イメージ分析（108 ページ）を参照してください。
ワークキュー数（Workqueue count）	`workqueue-count`	ワーク キュー数と指定した値の比較結果（等しい、多い、少ない）。workqueue-count ルール（51 ページ）を参照してください。
本文スキャン（Body Scanning）	`body-contains(<regular expression>)`	指定したパターンと一致するテキストまたは添付ファイルがメッセージに含まれているか。パターンの発生回数が、しきい値で指定した最小回数以上である必要があります。 エンジンは、配信ステータス部分と関連する添付ファイルをスキャンします。 本文スキャン （43 ページ）を参照してください。

ルール（**Rule**）	構文	説明
本文スキャン（Body Scanning）	`only-body-contains (<regular expression>)`	指定したパターンと一致するテキストがメッセージ本文に含まれているか。パターンの発生回数が、しきい値で指定した最小回数以上である必要があります。添付ファイルはスキャンされません。本文スキャン ルール （42 ページ）を参照してください。
暗号化検出（Encryption Detection）	`encrypted`	メッセージは暗号化されているか。暗号化検出ルール （43 ページ）を参照してください。
添付ファイル名（Attachment Filename）	`attachment-filename`	指定したパターンと一致するファイル名の添付ファイルがメッセージに含まれているか。添付ファイル名ルール （45 ページ）を参照してください。
添付タイプ（Attachment Type）	`attachment-type`	特定の MIME タイプの添付ファイルがメッセージに含まれているか。添付ファイル タイプ ルール （44 ページ）を参照してください。

ルール（**Rule**）	構文	説明
添付ファイル タイプ（Attachment File Type）	`attachment-filetype`	フィンガープリントに基づく特定のパターンと一致するファイル タイプの添付ファイルがメッセージに含まれているか（UNIX の `file` コマンドと同様）。添付ファイルが Excel または Word ドキュメントである場合、埋め込みファイル タイプの .exe、.dll、.bmp、.tiff、.pcx、.gif、.jpeg、.png、および Photoshop イメージを検索することもできます。 有効なフィルタを作成するには、ファイル タイプを引用符で囲む必要があります。一重引用符または二重引用符を使用できます。たとえば、.exe 添付ファイルを検索するには、次の構文を使用します。 `if (attachment-filetype == "exe")` 詳細については、添付ファイル名とアーカイブ ファイル内の単独の圧縮ファイル （45 ページ）を参照してください。

ルール（**Rule**）	構文	説明
Attachment MIME Type	`attachment-mimetype`	特定の MIME タイプの添付ファイルがメッセージに含まれているか。このルールは `attachment-type` ルールに似ていますが、MIME 添付ファイルで指定された MIME タイプのみが評価される点が異なります。（アプライアンスは、タイプが明示的に指定されていない場合、拡張子からファイルのタイプを「予測」することはありません）。添付ファイルのスキャン メッセージフィルタの例 （113ページ）を参照してください。
保護された添付ファイル（Attachment Protected）	`attachment-protected`	パスワード保護された添付ファイルがメッセージに含まれているか。保護された添付ファイルの隔離 （116ページ）を参照してください。

ルール（Rule）	構文	説明
保護されていない添付ファイル（Attachment Protected）	`attachment-unprotected`	attachment-unprotected フィルタ条件は、保護されていない添付ファイルをスキャン エンジンが検出した場合に true を返します。スキャン エンジンが添付ファイルを読み取ることができた場合、そのファイルは保護されていないと見なされます。zip ファイルに保護されていないメンバが含まれている場合、その zip ファイルは保護されていないと見なされます。 **注**：attachment-unprotected フィルタ条件と attachment-protected フィルタ条件は、相互に排他的ではありません。同じ添付ファイルをスキャンすると、両方のフィルタ条件で true が返される場合があります。これは、たとえば、zip ファイルに保護されたメンバと保護されていないメンバの両方が含まれている場合に発生します。 保護されていない添付ファイルの検出（116ページ）を参照してください。

ルール（Rule）	構文	説明
添付ファイルのスキャン（Attachment Scanning）	`attachment-contains (<regular expression>)`	指定したパターンと一致するテキストまたは別の添付ファイルが、メッセージの添付ファイルに含まれているか。パターンの発生回数が、しきい値で指定した最小回数以上である必要があります。 このルールは`body-contains()`ルールと似ていますが、このルールでは、メッセージの全体の「本文」をスキャンしないようにします。つまり、ユーザが添付ファイルとして表示する場合だけスキャンします。添付ファイルのスキャン メッセージフィルタの例（113ページ）を参照してください。
添付ファイルのスキャン（Attachment Scanning ）	`attachment-binary-contains (<regular expression>)`	指定したパターンと一致するバイナリ データが存在する添付ファイルがメッセージに含まれているか。 このルールは`attachment-contains ()`ルールに似ていますが、バイナリ データ内のパターンのみを検索します。

ルール（Rule）	構文	説明
添付ファイルのスキャン（Attachment Scanning ）	`every-attachment-contains (<regular expression>)`	このメッセージのすべての添付ファイルに、特定のパターンと一致するテキストが含まれているか。対象のテキストがすべての添付ファイル内に存在する必要があります。つまり実際に実行されるアクションは、各添付ファイルに対する「`attachment-contains()`」の論理AND演算です。本文はスキャンされません。パターンの発生回数が、しきい値で指定した最小回数以上である必要があります。 添付ファイルのスキャン メッセージフィルタの例（113ページ）を参照してください。
添付ファイルのサイズ（Attachment Size）	`attachment-size`	メッセージに含まれている添付ファイルのサイズが特定の範囲内に収まっているか。このルールは `body-size` ルールと似ていますが、このルールでは、メッセージの全体の「本文」をスキャンしないようにします。つまり、ユーザが添付ファイルとして表示する場合だけスキャンします。このサイズは、デコードする前に評価されます。添付ファイルのスキャン メッセージフィルタの例（113ページ）を参照してください。
公開ブラックリスト（Public Blacklists）	`dnslist(<query server>)`	送信者の IP アドレスがパブリック ブラックリスト サーバ（RBL）内に存在するか。DNS リスト ルール （46 ページ）を参照してください。

ルール（Rule）	構文	説明
SenderBaseレピュテーション（SenderBase Reputation）	`reputation`	送信者の SenderBase レピュテーション スコアの値。SenderBase レピュテーション ルール （46 ページ）を参照してください。
SenderBaseレピュテーションなし（No SenderBase Reputation）	`no-reputation`	SenderBase レピュテーションが「None」の場合に使用します。SenderBase レピュテーション ルール （46 ページ）を参照してください。
ディクショナリ	`dictionary-match (<dictionary_name>)`	メッセージ本文に、*dictionary_name* で指定した名前のコンテンツ ディクショナリの正規表現または用語が含まれているかどうかを判別します。パターンの発生回数が、しきい値で指定した最小回数以上である必要があります。ディクショナリ ルール （47 ページ）を参照してください。
添付ディクショナリ一致（Attachment Dictionary Match）	`attachment-dictionary-match (<dictionary_name>)`	添付ファイルに、*dictionary_name* で指定した名前のコンテンツ ディクショナリの正規表現が含まれているかどうかを判別します。パターンの発生回数が、しきい値で指定した最小回数以上である必要があります。ディクショナリ ルール （47 ページ）を参照してください。
件名ディクショナリ一致（Subject Dictionary Match）	`subject-dictionary-match (<dictionary_name>)`	件名ヘッダーに、*dictionary name* で指定した名前のコンテンツ ディクショナリの正規表現または用語が含まれているかどうかを判別します。ディクショナリ ルール （47 ページ）を参照してください。

ルール（Rule）	構文	説明
ヘッダーディクショナリ一致（Header Dictionary Match）	`header-dictionary-match (<dictionary_name>, <header>)`	指定したヘッダー（大文字と小文字を区別）に、*dictionary name* で指定した名前のコンテンツ ディクショナリの正規表現または用語が含まれているかどうかを判別します。ディクショナリ ルール （47 ページ）を参照してください。
本文ディクショナリ一致（Body Dictionary Match）	`body-dictionary-match (<dictionary_name>)`	このフィルタ条件は、辞書の用語がメッセージ本文に含まれていれば true を返します。このフィルタは、添付ファイルであると判断されないMIME部分の用語に一致します。また、ユーザが定義したしきい値が満たされた場合も true を返します（デフォルトのしきい値は 1 です）。ディクショナリ ルール （47 ページ）を参照してください。
エンベロープ受信者ディクショナリ一致（Envelope Recipient Dictionary Match）	`rcpt-to-dictionary-match (<dictionary_name>)`	エンベロープ受信者に、*dictionary name* で指定した名前のコンテンツ ディクショナリの正規表現または用語が含まれているかどうかを判別します。ディクショナリ ルール （47 ページ）を参照してください。
エンベロープ送信者ディクショナリ一致（Envelope Sender Dictionary Match）	`mail-from-dictionary-match (<dictionary_name>)`	エンベロープ送信者に、*dictionary name* で指定した名前のコンテンツ ディクショナリの正規表現または用語が含まれているかどうかを判別します。ディクショナリ ルール （47 ページ）を参照してください。

ルール（**Rule**）	構文	説明
SMTP認証済みユーザー一致（SMTP Authenticated User Match）	`smtp-auth-id-matches (<target>[, <sieve-char>])`	エンベロープ送信者のアドレスとメッセージ ヘッダーのアドレスが、送信者の認証済み SMTP ユーザ ID と一致するかどうかを判別します。SMTP 認証済みユーザー一致ルール （52 ページ）を参照してください。
[はい（True）]	`true`	すべてのメッセージと一致します。true ルール （34 ページ）を参照してください。
有効（Valid）	`valid`	メッセージに解析不能または無効な MIME 部分がある場合に false を返し、それ以外の場合は true を返します。有効なルール （35 ページ）を参照してください。
署名済み（Signed）	`signed`	メッセージが署名済みであるかどうかを判別します。署名付きルール （54 ページ）を参照してください。
署名証明書（Signed Certificate）	`signed-certificate (<field> [<operator> <regular expression>])`	メッセージ署名者またはX.509証明書発行者が特定のパターンと一致するかどうかを判別します。署名付き証明書ルール （54 ページ）を参照してください。

ルール（Rule）	構文	説明
ヘッダー繰り返し回数（Header Repeats）	`header-repeats (<target>, <threshold> [, <direction>])`	任意の時点で次の条件のメッセージが指定された数だけ検出されると、`true` を戻します。 • 過去 1 時間の同一件名ヘッダーを持つメッセージ • 過去 1 時間の同一のエンベロープ送信者からのメッセージ ヘッダー繰り返し回数ルール（57 ページ）を参照してください。
URL レピュテーション（URL Reputation）	`url-reputation` `url-no-reputation`	メッセージに含まれている任意の URL のレピュテーションスコアが、指定された範囲内にあるかどうか。 URL のレピュテーション スコアが使用できないかどうか。 URL レピュテーション ルール（59 ページ）および外部脅威フィードを消費する Cisco E メール セキュリティ ゲートウェイの設定を参照してください。
URL のカテゴリ（URL Category）	`url-category`	メッセージに含まれている任意の URL のカテゴリが、指定されたカテゴリに一致するかどうか。 URL カテゴリ ルール （60 ページ）を参照してください。

ルール（Rule）	構文	説明
破損した添付ファイル（Corrupt Attachment）	`attachment-corrupt`	破損した添付ファイルがメッセージに含まれているかどうか。 破損した添付ファイル ルール（60 ページ）を参照してください。
メッセージ言語（Message Language）	`message-language`	メッセージ（件名と本文）は選択したいずれかの言語であるか。 メッセージ言語ルール （61 ページ）を参照してください。
マクロ検出	`macro-detection-rule (['file_type-1', 'file_type-2', …,'file_type-n'])`	受信または送信メッセージにマクロが有効な添付ファイルが含まれているか。 マクロ検出ルール （62 ページ）を参照してください
偽装メールの検出	`forged-email-detection ("<dictionary_name>", <threshold>)`	メッセージの送信元アドレスが偽装されているか。メッセージの From: ヘッダーがコンテンツ辞書のユーザに類似している場合にチェックするルールです。 偽造メールの検出ルール （62 ページ）を参照してください。
重複境界検証	`duplicate_boundaries`	そのメッセージに、重複する MIME 境界が含まれるか。 重複境界検証ルール （63 ページ）を参照してください。
不正な形式の MIME ヘッダーの検出（Malformed MIME Header Detection）	`malformed-header`	メッセージに不正な形式の MIME ヘッダーが含まれているか。 不正な形式の MIME ヘッダー検出ルール （64 ページ）を参照してください。

ルール（Rule）	構文	説明
位置情報（GeoLocation）	`geolocation-rule (['country_name-1', 'country_name-2', 'country_name-n'])`	受信メッセージは、選択した国から発信されましたか。 （注） 位置情報メッセージ フィルタ ルールを使用する前に、アプライアンス上でスパム対策エンジンを有効にします。 地理位置情報ルール （64 ページ）を参照してください。

ドメインのレピュテーション	`Sender Domain Reputation: - sdr-reputation (<'sdr_verdict_range'>, <'domain_exception _list'>) - sdr-age (<'unit'>, <'operator'> <'actual value'>) - sdr-unscannable (<'domain_exception _list'>)`
	
	`External Threat Feeds: domain-external- threat-feeds (<'external_threat_ feed_source_name'>, <'header'> , <'domain _exception_list'>)`
		送信者ドメインは、指定された基準と一致していますか?
		• 送信者ドメインのレピュテーション
		• 外部脅威フィード
		ETF のドメイン レピュテーション ルール （65 ページ）またはSDR のドメイン レピュテーション ルール （65 ページ）を参照してください。
		詳細については、外部脅威フィードを消費する Cisco E メール セキュリティ ゲートウェイの設定 または 送信者ドメイン レピュテーション フィルタリング を参照してください。

Cisco アプライアンスに送信されるメッセージはいずれも、すべてのメッセージ フィルタで順番に処理されますが、最終アクションを指定した場合はそのアクションによりメッセージに対する以降の処理が停止されます。（メッセージ フィルタ アクション （3 ページ） を参照）。フィルタはすべてのメッセージに適用することもできます。また、ルールは論理接続子（AND、OR、NOT）を使用して結合することもできます。

ルールで使用する正規表現

ルールの定義に使用するアトミック テストの一部では、正規表現照合を行います。正規表現は複雑になる場合があります。次の表は、メッセージ フィルタ ルールで正規表現を適用する場合の目安として使用してください。

表 *3:* ルールで使用する正規表現

正規表現（abc）	フィルタ ルールの正規表現が文字列と一致すると判断されるのは、正規表現の一連の指示が文字列のいずれかの部分と一致する場合です。 たとえば、正規表現「Georg」は「George Of The Jungle」、「Georgy Porgy」、「La Meson Georgette as well as Georg」の各文字列と一致します。
カレット (^) ドル記号（$）	ドル記号（$）を含むルールは文字列の末尾のみと一致し、キャラット（^）を含むルールは文字列の先頭のみと一致します。 たとえば、正規表現「^Georg$」は文字列「Georg」のみと一致します。 空のヘッダーを検索するには、「"^$"」と指定します。
文字、空白、アットマーク（@）	文字、空白、アットマーク（@）を含むルールは、当該の文字自体と完全に一致します。 たとえば、正規表現「^George@admin$」は文字列「George@admin」のみと一致します。
ピリオド（. ）	ピリオド（.）を含むルールは任意の 1 文字（改行を除く）と一致します。 たとえば、「^...admin$」という正規表現は「macadmin」および「sunadmin」の各文字列とは一致しますが、「win32admin」とは一致しません。
アスタリスク（*）命令	アスタリスク（*）を含むルールは、「直前に指定されている文字が 0 回を含む任意の回数繰り返されている文字」と一致します。ピリオドとアスタリスクが続く場合（.*）は、任意の文字列（改行を除く）と一致します。 たとえば、「^P.*Piper$」という正規表現は、「PPiper」、「Peter Piper」、「P.Piper」、「Penelope Penny Piper」のどの文字列とも一致します。
バックスラッシュ特殊文字（\）	円記号は特殊文字のエスケープに使用します。シーケンス「\.」はピリオドそのもののみに一致し、「\$」はドル記号のみに一致し、「\^」はキャレット記号のみに一致します。たとえば、「^ik\.ac\.uk$」は「ik.ac.uk」という文字列のみと一致します。 **重要**：円記号はパーサーでも特殊なエスケープ文字として使用します。そのため、正規表現で円記号を使用する場合、*2* つの円記号が必要です。解析後には「実際に」使用される円記号 1 つのみが残り、正規表現システムに渡されます。上記の例を照合する場合は「^ik\\.ac\\.uk$」と入力することになります。

大文字と小文字を区別しない（`(?i)`）	トークン (?i) は、正規表現の残りの部分で大文字と小文字が区別されないことを表します。このトークンを、大文字と小文字を区別する正規表現の先頭に配置すると、大文字と小文字が一切区別されない照合が行われます。 たとえば、「`(?i)viagra`」という正規表現は、「`Viagra`」、「`vIaGrA`」、「`VIAGRA`」と一致します。
繰り返し回数 `{min,max}`	1 つ前のトークンの繰り返し回数を指定する正規表現表記がサポートされています。 たとえば、「`fo{2,3}`」は「`foo`」および「`fooo`」とは一致しますが、「fo」や「fofo」とは一致しません。 `if(header('To') == "^.{500,}")` というステートメントは、500 文字以上が使用されている「To」ヘッダーを検索します。
または (|)	代替、つまり「or」演算子に相当します。「A および B」が正規表現である場合、「`A|B`」は「A」と「B」のいずれかに一致する文字列と一致します。 たとえば、「`foo|bar`」という表現は「`foo`」や「`bar`」とは一致しますが、「`foobar`」とは一致しません。

関連項目

- メッセージのフィルタリングでの正規表現の使用 （29 ページ）
- 正規表現の使用に関するガイドライン （30 ページ）
- 正規表現と非 ASCII 文字セット （30 ページ）
- n テスト （30 ページ）
- 大文字と小文字の区別 （30 ページ）
- 効率的なフィルタの作成 （31 ページ）
- PDF と正規表現 （31 ページ）

メッセージのフィルタリングでの正規表現の使用

フィルタを使用して、ASCII 以外の形式でエンコードされているメッセージの内容（ヘッダーと本文）の文字列とパターンを検索できます。具体的には、本システムでは次の場所にある非 ASCII 文字を検索する正規表現（regex）を使用できます。

- メッセージ ヘッダー
- MIME 添付ファイル名の文字列
- メッセージ本文：
  - MIME ヘッダーがない本文（従来の形式の電子メール）
  - エンコードを示す MIME ヘッダーがあり、MIME 部分がない本文
  - エンコードが指定されているマルチパート MIME メッセージ
  - 上記の本文のうち、MIME ヘッダーでエンコードが指定されていないもの

メッセージまたは本文の任意の部分（添付ファイルを含む）の照合に正規表現を使用できます。添付ファイルのタイプとして HTML、MS Word、Excel など多数のタイプを対象にできます。対象となる文字セットとして、gb2312、HZ、EUC、JIS、Shift-JIS、Big5、Unicode などがあります。正規表現のメッセージ フィルタ ルールを作成するには、コンテンツ フィルタ GUI を使用するか、テキストエディタでファイルを作成してからシステムにインポートします。詳細については、CLI を使用したメッセージ フィルタの管理 （117 ページ）およびスキャン動作の設定 （141 ページ）を参照してください。

正規表現の使用に関するガイドライン

プレフィックスではなく文字列全体を照合する場合は、正規表現の先頭にキャレット（^）、末尾にドル記号（$）をそれぞれ配置する必要があります。

（注） 空の文字列を照合する場合に「」を使用すると、実際にはすべての文字列が一致します。代わりに、「^$」を使用してください。たとえば、subject ルール （35 ページ）の 2 番めの例がこれに該当します。

また、文字としてのピリオドを照合するには、正規表現でピリオドをエスケープする必要があります。たとえば、sun.com という正規表現は「thegodsunocommando」という文字列と一致しますが、^sun\.com$ という正規表現は「sun.com」という文字列のみと一致します。

技術的には、ここで使用する正規表現のスタイルは **Python re Module** モジュール スタイルの正規表現です。Python スタイルの正規表現の詳細については、http://www.python.org/doc/howto/ からアクセスできる「Python Regular Expression HOWTO」を参考にしてください。

正規表現と非 ASCII 文字セット

一部の言語では、「単語」や「単語境界」、「大文字と小文字」という概念が存在しません。

単語を構成する文字（正規表現で「\w」と表される文字）の識別などが必要になる複雑な正規表現では、ロケールが不明な場合、またはエンコードが不明な場合、問題が発生します。

n テスト

正規表現の照合テストは、シーケンス == とシーケンス != を使用して行うことができます。次に例を示します。

```
rcpt-to ==
 "^goober@dev\\.null\\....$" (matching)


rcpt-to != "^goober@dev\\.null\\....$" (non-matching)
```

大文字と小文字の区別

特に明記されている場合を除き、正規表現では大文字と小文字が区別されます。正規表現で foo を検索する場合、FOO や Foo は一致しません。

効率的なフィルタの作成

次の例は、同じ処理を行う 2 つのフィルタですが、最初の例の方が CPU の使用率が高くなります。2 番めのフィルタの方が効率的な正規表現を使用しています。

```
attachment-filter: if ((recv-listener == "Inbound") AND
((((((((((((((((((((((((((((((((((((((((((((((((((((attachment-filename ==

"\\.386$") OR (attachment-filename == "\\.exe$")) OR (attachment-filename == "\\.ad$"))
 OR
(attachment-filename == "\\.ade$")) OR (attachment-filename == "\\.adp$")) OR
(attachment-filename == "\\.asp$")) OR (attachment-filename == "\\.bas$")) OR
(attachment-filename == "\\.bat$")) OR (attachment-filename == "\\.chm$")) OR
(attachment-filename == "\\.cmd$")) OR (attachment-filename == "\\.com$")) OR
(attachment-filename == "\\.cpl$")) OR (attachment-filename == "\\.crt$")) OR
(attachment-filename == "\\.exe$")) OR (attachment-filename == "\\.hlp$")) OR
(attachment-filename == "\\.hta$")) OR (attachment-filename == "\\.inf$")) OR
(attachment-filename == "\\.ins$")) OR (attachment- filename == "\\.isp$")) OR
(attachment-filename == "\\.js$")) OR (attachment-filename == "\\.jse$")) OR
(attachment- filename == "\\.lnk$")) OR (attachment-filename == "\\.mdb$")) OR
(attachment-filename == "\\.mde$")) OR (attachment-filename == "\\.msc$")) OR
(attachment-filename == "\\.msi$")) OR (attachment-filename == "\\.msp$")) OR
(attachment-filename == "\\.mst$")) OR (attachment-filename == "\\.pcd$")) OR
(attachment-filename == "\\.pif$")) OR (attachment-filename == "\\.reg$")) OR
(attachment-filename == "\\.scr$")) OR (attachment-filename == "\\.sct$")) OR
(attachment-filename == "\\.shb$")) OR (attachment-filename == "\\.shs$")) OR
(attachment-filename == "\\.url$")) OR (attachment-filename == "\\.vb$")) OR
(attachment-filename == "\\.vbe$")) OR (attachment-filename == "\\.vbs$")) OR
(attachment-filename == "\\.vss$")) OR (attachment-filename == "\\.vst$")) OR
(attachment-filename == "\\.vsw$")) OR (attachment-filename == "\\.ws$")) OR
(attachment-filename == "\\.wsc$")) OR (attachment-filename == "\\.wsf$")) OR
(attachment-filename == "\\.wsh$"))) { bounce(); }
```

この例では、AsyncOS は正規表現エンジンを 30 回（添付ファイルタイプと recv-listener のそれぞれに 1 回ずつ）起動する必要があります。

かわりに、次のようなフィルタを作成します。

```
attachment-filter: if (recv-listener == "Inbound") AND (attachment-filename == "\\.
(386|exe|ad|ade|adp|asp|bas|bat|chm|cmd|com|cpl|crt|exe|hlp|hta|inf|ins|isp|js|jse|l
nk|mdb|mde|msc|msi|msp|mst|pcd|pif|reg|scr|sct|shb|shs|
url|vb|vbe|vbs|vss|vst|vsw|ws|wsc|wsf|wsh)$") {
```

正規表現エンジンの起動回数は 2 回だけで、「()」の追加やスペルの誤りについて心配する必要がなくなるためフィルタの管理も大幅に簡単になります。また、最初の例に比べて CPU オーバーヘッドが低下します。

PDF と正規表現

PDF の生成方法によっては、スペースや改行がないことがあります。このような場合、スキャン エンジンは、ページ内の単語の位置に基づき、論理的なスペースと改行の挿入を試みます。たとえば、1 つの単語の中に複数のフォントやフォント サイズが混在する場合、生成される PDF コードからスキャン エンジンが単語と改行を判別するのが難しくなります。このように生成された PDF ファイルで正規表現による照合を行うと、スキャン エンジンは予期しない結果を返す場合があります。

たとえば、PowerPoint 文書に挿入した単語の中に、単語内の文字ごとに異なるフォントやフォントサイズが設定されているものがあるとします。このアプリケーションから生成された PDF をスキャン エンジンが読み取ると、論理的なスペースと改行が挿入されます。PDF の構造が原因で、「callout」という単語が「call out」または「c a l lout」と解釈されることがあります。このいずれかの表現を正規表現「callout」と照合しようとすると、一致なしという結果になります。

スマート ID

メッセージの内容をスキャンするメッセージ ルールを使用する場合、スマート ID を使用するとデータ内の特定のパターンを検出できます。

スマート ID で、データ内の次のパターンを検出できます。

- クレジット カード番号
- 米国社会保障番号
- CUSIP ナンバー
- ABA ナンバー

フィルタでスマート ID を使用するには、本文または添付ファイルのコンテンツをスキャンするフィルタ ルールで次のキーワードを使用します。

表 *4:* メッセージ フィルタのスマート *ID*

キーワード	スマート ID	説明
*credit	クレジットカード番号	14、15、および16桁のクレジットカード番号を識別します。 注意：スマート ID は enRoute カードを識別しません。
*aba	ABA 送金番号	ABA 送金番号を識別します。
*ssn	社会保障番号	米国社会保障番号を識別します。*ssn スマート ID はダッシュ、ピリオド、スペースがある社会保障番号を識別します。
*cusip	CUSIP 番号	CUSIP 番号を識別します。

関連項目

- スマート ID の構文 （32 ページ）

スマート ID の構文

フィルタ ルールでスマート ID を使用する場合、次の例のように、本文または添付ファイルをスキャンするフィルタ ルールの中でスマート ID キーワードを引用符で囲みます。

```
ID_Credit_Cards:

if(body-contains("*credit")){

notify("legaldept@example.com");

}
.
```

また、コンテンツ ディクショナリの一部としてコンテンツ フィルタ内でスマート ID を使用することもできます。

（注） スマート ID キーワードは通常の正規表現や他のキーワードと組み合わせて使用できません。たとえば、「`*credit|*ssn`」というパターンは有効ではありません。

（注） *ssn スマート ID による誤判定を防ぐため、*ssn スマート ID は他のフィルタ条件とあわせて使用すると有用な場合があります。たとえば、「only-body-contains」フィルタ条件を使用することができます。この場合、検索文字列がメッセージ本文のすべての MIME 部分に存在する場合のみ式が true であると判定されます。たとえば、次のようなフィルタを作成できます。

```
SSN-nohtml: if only-body-contains("*ssn") { duplicate-quarantine("Policy");}
```

メッセージ フィルタ ルールの説明と例

次のセクションでは、使用されるさまざまなメッセージ フィルタ ルールについて説明し、その例を示します。

関連項目

- true ルール （34 ページ）
- 有効なルール （35 ページ）
- subject ルール （35 ページ）
- エンベロープ受信者ルール （36 ページ）
- グループ内エンベロープ受信者ルール （36 ページ）
- エンベロープ送信者ルール （37 ページ）
- グループ内エンベロープ送信者ルール （37 ページ）
- 送信者グループ ルール （37 ページ）
- 本文サイズ ルール （38 ページ）
- リモート IP ルール （39 ページ）
- 受信リスナー ルール （39 ページ）
- 受信 IP インターフェイス ルール （39 ページ）
- 日付ルール （40 ページ）

- ヘッダー ルール （40 ページ）
- 乱数ルール （41 ページ）
- 受信者数ルール （42 ページ）
- アドレス数ルール （42 ページ）
- 本文スキャン ルール （42 ページ）
- 本文スキャン （43 ページ）
- 暗号化検出ルール （43 ページ）
- 添付ファイル タイプ ルール （44 ページ）
- 添付ファイル名ルール （45 ページ）
- DNS リスト ルール （46 ページ）
- SenderBase レピュテーション ルール （46 ページ）
- ディクショナリ ルール （47 ページ）
- SPF-Status ルール （49 ページ）
- SPF-Passed ルール （50 ページ）
- S/MIME ゲートウェイ メッセージ ルール （51 ページ）
- S/MIME ゲートウェイ検証済みルール （51 ページ）
- workqueue-count ルール （51 ページ）
- SMTP 認証済みユーザー一致ルール （52 ページ）
- 署名付きルール （54 ページ）
- ヘッダー繰り返し回数ルール （57 ページ）
- URL レピュテーション ルール （59 ページ）
- URL カテゴリ ルール （60 ページ）
- 破損した添付ファイル ルール （60 ページ）
- メッセージ言語ルール （61 ページ）
- マクロ検出ルール （62 ページ）
- 偽造メールの検出ルール （62 ページ）
- 重複境界検証ルール （63 ページ）
- 不正な形式の MIME ヘッダー検出ルール （64 ページ）
- 地理位置情報ルール （64 ページ）
- ETF のドメイン レピュテーション ルール （65 ページ）
- SDR のドメイン レピュテーション ルール （65 ページ）

true ルール

true ルールはすべてのメッセージと一致します。たとえば、次のルールはテスト対象となるすべてのメッセージについて、IP インターフェイスを external に変更します。

```
externalFilter:

   if (true)

   {
```

```
        alt-src-host('external');

    }
```

有効なルール

valid ルールは、メッセージに解析不能または無効な MIME 部分が含まれている場合に false を返し、それ以外の場合は true を返します。たとえば、次のルールはテスト対象のメッセージのうち解析不能なメッセージをすべてドロップします。

```
not-valid-mime:

if not valid

{

drop();

}
```

subject ルール

subject ルールは、件名ヘッダーの値が指定した正規表現と一致するメッセージを選択します。

たとえば、次のフィルタは、件名が「Make Money...」という語句で始まるすべてのメッセージを廃棄します。

```
not-valid-mime:

if not valid

{

drop();

}
```

ヘッダーの値で検索する非 ASCII 文字を指定することができます。

ヘッダーに関する操作を行う場合、ヘッダーの現在の値には処理中に行われた変更（メッセージのヘッダーの追加、削除、変更を行うフィルタ処理など）が含まれている点に注意してください。詳細については、メッセージ ヘッダー ルールおよび評価 （6 ページ）を参照してください。

次のフィルタは、ヘッダーが空の場合、またはメッセージにヘッダーがない場合に true を返します。

```
EmptySubject_To_filter:

if (header('Subject') != ".") OR

(header('To') != ".") {

drop();

}
```

（注） このフィルタはSubjectヘッダーとToヘッダーが空の場合にtrueを返しますが、ヘッダーがない場合もtrueを返します。指定したヘッダーがメッセージ内にない場合でも、このフィルタはtrueを返します。

エンベロープ受信者ルール

rcpt-toルールは、いずれかのエンベロープ受信者が指定した正規表現と一致するメッセージを選択します。たとえば、次のフィルタは「scarface」という文字列を含む電子メールアドレス宛てに送信されたすべてのメッセージをドロップします。

（注） rcpt-toルールで使用する正規表現では、大文字と小文字は区別されません。

```
scarfaceFilter:

if (rcpt-to == 'scarface')

{

drop();

}
```

（注） rcpt-toルールはメッセージに基づいています。メッセージに複数の受信者が設定されている場合、いずれか1人の受信者がルールと一致していれば、指定した処理がすべての受信者に対するメッセージに適用されます。

グループ内エンベロープ受信者ルール

rcpt-to-groupルールは、いずれかのエンベロープ受信者が指定したLDAPグループのメンバであるメッセージを選択します。たとえば、次のフィルタは「ExpiredAccounts」というLDAPグループ内の電子メールアドレス宛てに送信されたすべてのメッセージをドロップします。

```
expiredFilter:

if (rcpt-to-group == 'ExpiredAccounts')

{

drop();

}
```

（注） rcpt-to-group ルールはメッセージに基づいています。メッセージに複数の受信者が設定されている場合、いずれか1人の受信者がルールと一致していれば、指定した処理がすべての受信者に対するメッセージに適用されます。

エンベロープ送信者ルール

mail-from ルールは、エンベロープ送信者が指定した正規表現と一致するメッセージを選択します。たとえば、次のフィルタを実行すると admin@yourdomain.com により送信されたすべてのメッセージがただちに出力されます。

（注） mail-from ルールで使用する正規表現では、大文字と小文字は区別されません。次の例では、ピリオドがエスケープ処理されています。

```
kremFilter:

if (mail-from == '^admin@yourdomain\\.com$')

{

skip-filters();

}
```

グループ内エンベロープ送信者ルール

mail-from-group ルールは、エンベロープ送信者が演算子の右辺で指定した LDAP グループに属している（不一致を検索する場合は、送信者の電子メールアドレスが指定したLDAP グループに属していない）メッセージを選択します。たとえば、次のフィルタを実行すると、「KnownSenders」というLDAP グループの電子メールアドレスにより送信されたすべてのメッセージがただちに出力されます。

```
SenderLDAPGroupFilter:

if (mail-from-group == 'KnownSenders')

{

skip-filters();

}
```

送信者グループ ルール

sendergroup メッセージ フィルタは、リスナーのホスト アクセス テーブル（HAT）でどの送信者グループが一致するかに基づいて、メッセージを選択します。このルールは「==」（一致を検索する場合）または「!=」（不一致を検索する場合）を使用して、指定した正規表現（式の右辺）との一致をテストします。たとえば、次のメッセージ フィルタ ルールは、メッセー

ジの送信者グループが正規表現「Internal」と一致する場合に`true`を返し、その場合はメッセージを代替メール ホストに送信します。

```
senderGroupFilter:

if (sendergroup == "Internal")

{

alt-mailhost("[172.17.0.1]");

}
```

本文サイズ ルール

本文サイズとはメッセージのサイズのことで、ヘッダーと添付ファイルも含みます。`body-size`ルールは、本文サイズを指定された数値と比較し、条件に一致するメッセージを選択します。たとえば、次のフィルタは本文サイズが5メガバイトを超えるすべてのメッセージをバウンスします。

```
BigFilter:

if (body-size > 5M)

{

bounce();

}
```

`body-size`を使用すると次のような比較ができます。

例	比較の種類
`body-size < 10M`	より少ない
`body-size <= 10M`	以下
`body-size > 10M`	右辺と比較して大きい
`body-size >= 10M`	以上
`body-size == 10M`	等しい
`body-size != 10M`	等しくない

サイズ指定にはサフィクスを使用すると便利です。

数量	説明
`10b`	10バイト（「10」に同じ）

数量	説明
13k	13 キロバイト
5M	5 メガバイト
40G	40 ギガバイト（注：Cisco アプライアンスでは 100 メガバイトを超えるメッセージを処理できません）

リモート IP ルール

remote-ip ルールは、メッセージを送信したホストの IP アドレスが特定のパターンと一致するかどうかを確認するためのテストを実行します。IP アドレスは、インターネット プロトコル バージョン 4（IPv4）またはインターネット プロトコル バージョン 6（IPv6）を指定できます。IP アドレス パターンは、「送信者グループの構文」に記載されている allowed hosts 表記を使用して指定されます。ただし、SBO、SBRS、dnslist 表記および特殊キーワード ALL を除きます。

allowed hosts 表記では、IP アドレス（ホスト名ではない）の順序と数値での範囲のみを指定できます。たとえば、次のフィルタは 10.1.1.x（X は 50、51、52、53、54、55 のいずれか）の形式の IP アドレスから送信されていないすべてのメッセージをバウンスします。

```
notMineFilter:

if (remote-ip != '10.1.1.50-55')

{

bounce();

}
```

受信リスナー ルール

recv-listener ルールは、名前付きリスナーで受信したメッセージを選択します。リスナー名は、現在システム上で設定されているリスナーのいずれかのニックネームである必要があります。たとえば、次のフィルタを実行すると、expedite という名前のリスナーから受信したすべてのメッセージがただちに出力されます。

```
expediteFilter:

if (recv-listener == 'expedite')

{

skip-filters();

}
```

受信 IP インターフェイス ルール

recv-int ルールは、名前付きインターフェイス経由で受信したメッセージを選択します。インターフェイス名は、現在システムに設定されているインターフェイスのいずれかのニックネー

ムである必要があります。たとえば、次のフィルタは、outside という名前のインターフェイスから受信したすべてのメッセージをバウンスします。

```
outsideFilter:

if (recv-int == 'outside')

{

bounce();

}
```

日付ルール

date ルールは、現在の日時と指定した時刻を照合します。日付ルールは、*MM/DD/YYYY hh:mm:ss* という形式のタイムスタンプを含む文字列と比較されます。このルールは、特定の日時（米国形式）の前または後に実行する処理を指定する場合に便利です。（米国以外の日付形式を使用しているメッセージを検索する場合は問題が発生することがあります）。次のフィルタは、2003 年 7 月 28 日の午後 1 時より後に campaign1@yourdomain.com から送信されたすべてのメッセージをバウンスします。

```
TimeOutFilter:

if ((date > '07/28/2003 13:00:00') and (mail-from ==

'campaign1@yourdomain\\.com'))

{

bounce();

}
```

（注） date ルールを $Date メッセージ フィルタ処理変数と混同しないようにしてください。

ヘッダー ルール

header() ルールは、メッセージ ヘッダーがかっこ内で引用されている特定のヘッダー（“ヘッダー名”）と一致するかどうかを確認します。このルールは subject ルールと同様に正規表現と比較することもできますが、比較を行わずに使用することもできます。この場合、メッセージにそのヘッダーがあれば「true」、なければ「false」となります。たとえば、次の例ではヘッダー X-Sample の有無、およびこのヘッダーの値に「sample text」という文字列が含まれているかどうかを確認しています。一致する場合は、メッセージがバウンスされます。

```
FooHeaderFilter:

if (header('X-Sample') == 'sample text')

{

bounce();

}
```

ヘッダーの値で検索する非 ASCII 文字を指定することができます。

次の例では、比較を行わずにヘッダールールを適用しています。この場合、ヘッダー `X-DeleteMe` が見つかると、そのヘッダーがメッセージから削除されます。

```
DeleteMeHeaderFilter:

if header('X-DeleteMe')

{

strip-header('X-DeleteMe');

}
```

ヘッダーに関する操作を行う場合、ヘッダーの現在の値には処理中に行われた変更（メッセージのヘッダーの追加、削除、変更を行うフィルタ処理など）が含まれている点に注意してください。詳細については、メッセージ ヘッダー ルールおよび評価 （6 ページ）を参照してください。

乱数ルール

`random` ルールは、0 から N-1（N はルール名の後のかっこで指定される整数値）までの乱数を生成します。このルールでは `header()` ルールと同様に比較を行うこともできますが、「単項」形式で単独使用することもできます。単項形式では、生成された乱数が 0 でない場合に `true` と評価されます。たとえば、次のフィルタはいずれも内容としては同じもので、2 分の 1 の確率で Virtual Gateway アドレス A が選択され、残り 2 分の 1 の確率で Virtual Gateway アドレス B が選択されます。

```
load_balance_a:

if (random(10) < 5)
{

alt-src-host('interface_a');
}

else

{

alt-src-host('interface_b');

}

load_balance_b:

if (random(2))

{

alt-src-host('interface_a');

}

else

{

alt-src-host('interface_b');
```

```
}
```

受信者数ルール

rcpt-count ルールは、body-size ルールと同様に、メッセージの受信者の数を整数値と比較します。このルールを使用すると、ユーザが一度に多数のユーザに電子メールを送信することを防止でき、また大規模なメール送信キャンペーンが特定の Virtual Gateway アドレス経由で行われるようにすることができます。次の例では、受信者数が 100 件を超える電子メールが特定の Virtual Gateway アドレスを経由して送信されます。

```
large_list_filter:

if (rcpt-count > 100) {

alt-src-host('mass_mailing_interface');

}
```

アドレス数ルール

addr-count() メッセージ フィルタ ルールは、1 つ以上のヘッダー文字列を対象に、各行の受信者数を計算し、受信者の累積数をレポートします。このフィルタは、エンベロープの受信者ではなくメッセージ本文のヘッダーに対して機能する点が rcpt-count フィルタ ルールと異なります。次の例では、このフィルタ ルールにより長い受信者リストが「undisclosed-recipients」というエイリアスに置き換えられています。

```
large_list_filter:


if (rcpt-count > 100) {

alt-src-host('mass_mailing_interface');

}
```

本文スキャン ルール

body-contains() ルールは、受信する電子メールとその添付ファイルをスキャンし、パラメータで定義された特定のパターンの有無を確認します。これには、配信ステータス部および関連付けられている添付ファイルが含まれます。body-contains() ルールでは複数行を対象とした照合は行われません。スキャンのロジックを [スキャン動作（Scan Behavior）] ページまたは CLI の scanconfig コマンドで変更することにより、スキャンの対象となる、またはスキャンの対象から除外する MIME タイプを定義できます。また、スキャン結果を true と評価するために検出する必要がある一致の最小数を指定することもできます。

デフォルトでは、MIME タイプが video/*、audio/*、image/* 以外であるすべての添付ファイルがスキャンされます。複数のファイルが含まれている .zip、.bzip、.compress、.tar、.gzip の各アーカイブ添付ファイルがスキャンされます。スキャン対象となる、「ネストされた」アーカイブ添付ファイル（.zip に格納されている .zip など）の数を設定できます。

詳細については、スキャン動作の設定 （141 ページ）を参照してください。

本文スキャン

AsyncOSが本文スキャンを実行する場合、正規表現を使用して本文のテキストと添付ファイルをスキャンします。式には最小しきい値を指定することができ、スキャンエンジンがこの最小回数だけ正規表現との一致を検出すると、この式は `true` と評価されます。

AsyncOS はメッセージの各種の MIME 部分を評価し、テキスト形式になっているすべての MIME部分をスキャンします。最初の部分でMIMEタイプがテキストに指定されている場合、AsyncOSはテキスト部分を識別します。AsyncOSはメッセージで指定されたエンコードに基づいてエンコードを決定し、テキストを Unicode に変換します。その後、Unicode 領域で正規表現を検索します。メッセージでエンコードが指定されていない場合は、[スキャン動作（Scan Behavior）] ページまたは `scanconfig` コマンドで指定されたエンコードが使用されます。

メッセージのスキャン時にAsyncOSがMIME部分を評価する方法の詳細については、メッセージ本文とメッセージ添付ファイル （6 ページ）を参照してください。

MIME 部分がテキストでない場合、AsyncOS は .zip または .tar からファイルを抽出するか、圧縮されたファイルを抽出します。データを抽出した後、スキャン エンジンはファイルのエンコードを識別し、ファイルのデータを Unicode 形式で返します。その後、AsyncOS は Unicode 領域で正規表現を検索します。

次の例では、本文のテキストと添付ファイルで「Company Confidential」という文字列を検索します。この例では、最小しきい値が2件に設定されているため、スキャンエンジンがこの文字列を2件以上検出すると、該当するメッセージをすべてバウンスし、法務部門に通知します。

```
ConfidentialFilter:

if (body-contains('Company Confidential',2)) {

notify ('legaldept@example.domain');

bounce();

}
```

メッセージの本文のみをスキャンする場合は、`only-body-contains` を使用します。

```
disclaimer:

if (not only-body-contains('[dD]isclaimer',1) ) {

notify('hresource@example.com');

}
```

暗号化検出ルール

`encrypted` ルールは、メッセージの内容に暗号化データが存在するかどうかを調査します。このルールは暗号化データの復号は行わず、メッセージの内容に暗号化データが存在するかどうかのみを調査します。このルールは、ユーザが暗号化された電子メールを送信できないようにする場合に便利です。

（注） 暗号化されたルールは、メッセージの内容の暗号化されたデータのみを検出できます。暗号化された添付ファイルは検出しません。

encrypted は true ルールと同様に、パラメータを使用せず、比較も行いません。暗号化されたデータが検出された場合に true、検出されなかった場合に false を返します。この機能を実行するにはメッセージのスキャンが必要になるため、[スキャン動作（Scan Behavior）] ページまたは scanconfig コマンドで定義されたスキャン設定が使用されます。オプションの設定の詳細については、スキャン動作の設定 （141 ページ）を参照してください。

次のフィルタは、リスナー経由で送信されたすべての電子メールを確認し、メッセージに暗号化されたデータが含まれる場合は、該当するメッセージが BCC で法務部門宛てに送信され、バウンスされます。

```
prevent_encrypted_data:

if (encrypted) {

bcc ('legaldept@example.domain');

bounce();

}
```

添付ファイル タイプ ルール

attachment-type ルールはメッセージ内の各添付ファイルの MIME タイプを確認し、指定されたパターンと一致するかどうかを判別します。このパターンは[スキャン動作（Scan Behavior）] ページまたは scanconfig コマンドで使用する形式（スキャン動作の設定 （141 ページ）を参照）と同じ形式である必要があり、スラッシュ（/）の左右の一方でアスタリスクをワイルドカードとして使用できます。メッセージの添付ファイルがここで指定した MIME タイプと一致する場合、このルールは「true」を返します。

この機能を実行するにはメッセージのスキャンが必要となるため、スキャン動作の設定 （141 ページ）で説明されているすべてのオプションが適用されます。

メッセージの添付ファイルを操作するために使用できるメッセージ フィルタ ルールの詳細については、添付ファイルのスキャン （105 ページ）を参照してください。

次のフィルタは、リスナー経由で送信されたすべての電子メールを確認し、MIME タイプが video/* である添付ファイルがメッセージに含まれる場合は、該当するメッセージがバウンスされます。

```
bounce_video_clips:

if (attachment-type == 'video/*') {

bounce();

}
```

添付ファイル名ルール

attachment-filename ルールはメッセージ内の各添付ファイルの名前を確認し、指定されたパターンと一致するかどうかを判別します。この比較では大文字と小文字は区別されます。この比較ではスペースの有無も区別されるため、ファイル名の末尾にスペースがある状態でエンコードされていると、フィルタはその添付ファイルをスキップします。メッセージの添付ファイルのいずれかが指定したファイル名と一致すると、このルールは true を返します。

次の点に注意してください。

- 各添付ファイルの名前はMIMEヘッダーからキャプチャされます。MIMEヘッダーにあるファイル名の末尾にはスペースがある場合があります。
- 添付ファイルがアーカイブの場合、Cisco アプライアンスはアーカイブの内部からファイル名を取得し、スキャン設定ルール（スキャン動作の設定 （141 ページ）を参照）を適用します。
  - 添付ファイルが 1 個の圧縮ファイル（拡張子を問わず）である場合、アーカイブであるとは見なされず、この圧縮ファイルの名前は取得されません。つまり、このファイルは attachment-filename ルールでは処理されません。このようなファイルの例としては、gzip で圧縮された実行可能ファイル（.exe）などがあります。
  - 添付ファイルが単独の圧縮ファイルである場合（foo.exe.gz など）、正規表現を使用して圧縮ファイル内の特定のファイルタイプを検索します。添付ファイル名とアーカイブ ファイル内の単独の圧縮ファイル （45 ページ）を参照してください。

メッセージの添付ファイルを操作するために使用できるメッセージ フィルタ ルールの詳細については、添付ファイルのスキャン （105 ページ）を参照してください。

次のフィルタは、リスナー経由で送信されたすべての電子メールを確認し、ファイル名が *.mp3 である添付ファイルがメッセージに含まれる場合は、該当するメッセージがバウンスされます。

```
block_mp3s:

if (attachment-filename == '(?i)\\.mp3$') {

bounce();

}
```

関連項目

- 添付ファイル名とアーカイブ ファイル内の単独の圧縮ファイル （45 ページ）

添付ファイル名とアーカイブ ファイル内の単独の圧縮ファイル

次に、アーカイブ（gzip で作成したものなど）にある単独の圧縮ファイルを照合する例を示します。

```
quarantine_gzipped_exe_or_pif:

if (attachment-filename == '(?i)\\.(exe|pif)($|.gz$)') {

quarantine("Policy");
```

```
}
```

DNS リスト ルール

`dnslist()` ルールは、クエリの実行にDNSBL方式（「ip4rルックアップ」とも呼ばれます）を使用するパブリック DNS リスト サーバを照会します。着信接続の IP アドレスは反転され（IP が 1.2.3.4 の場合は 4.3.2.1 になり）、かっこ内のサーバ名にプレフィックスとして追加されます（サーバ名の先頭がピリオドでない場合は、サーバ名とプレフィックスを区切るためのピリオドが追加されます）。DNS クエリーが生成され、システムには DNS 失敗応答（接続の IP アドレスがサーバのリストにないことを示す）または IP アドレス（アドレスが見つかったことを示す）が返されます。返される IP アドレスは通常、`127.0.0.x`（`x` は 0 ～ 255 のうちほぼすべての数）の形式になります（IPアドレス範囲は許可されていません）。一部のサーバは、リスト生成の理由に基づいてそれぞれ異なる数字を返しますが、それ以外のサーバはすべての一致に対して同じ結果を返します。

`dnslist()` は、`header()` ルールと同様に、単項または二項比較で使用できます。単独では、応答を受信すると `true` を返し、応答がない場合（DNS サーバが到達不能の場合など）は `false` を返します。

次のフィルタを実行すると、送信者が Cisco Bonded Sender 情報サービス プログラムにボンドされている場合、そのメッセージがただちに出力されます。

```
whitelist_bondedsender:

if (dnslist('query.bondedsender.org')) {

skip-filters();

}
```

オプションで、等式（`==`）または不等式（`!=`）を使用して結果を文字列と比較することもできます。

次のフィルタは、サーバから「`127.0.0.2`」が返されるメッセージをドロップします。応答がそれ以外の内容であれば、このルールは false を返し、フィルタは無視されます。

```
blacklist:

if (dnslist('dnsbl.example.domain') == '127.0.0.2') {

drop();

}
```

SenderBase レピュテーション ルール

`reputation` ルールは、SenderBase レピュテーション スコアを他の値と比較して確認します。>、==、<= などのすべての比較演算子を使用できます。メッセージに SenderBase レピュテーション スコアがない場合（これまでスコアがまったく確認されていないか、SenderBase レピュテーション サービス クエリー サーバから応答を取得できなかった場合）、レピュテーション スコアとの比較はすべて失敗します（数値がいずれかの値より大きいまたは小さい、いずれか

の値と等しいまたは等しくないという判別ができません）。次に説明する no-reputation ルールを使用すると、SBRS スコアが「none」であるかどうかを確認できます。次の例では、SenderBase レピュテーション サービスから返されるレピュテーション スコアがしきい値の -7.5 を下回る場合に、メッセージの「Subject:」行の先頭に「*** BadRep ***」が付加されます。

```
note_bad_reps:

if (reputation < -7.5) {
strip-header ('Subject');
insert-header ('Subject', '*** BadRep $Reputation *** $Subject');
}
```

詳細については、「送信者レピュテーション フィルタリング」の章を参照してください。アンチスパム システムのバイパス アクション （98 ページ）も参照してください。

SenderBase レピュテーション ルールによる値は -10 ～ 10 ですが、NONE という値が返される場合もあります。NONE について特に確認が必要な場合は、no-reputation ルールを使用します。

```
none_rep:

if (no-reputation) {

strip-header ('Subject');

insert-header ('Subject', '*** Reputation = NONE *** $Subject');

}
```

ディクショナリ ルール

メッセージ本文に、「dictonary_name」という名前のコンテンツ ディクショナリにある正規表現または用語が含まれている場合、dictionary-match (<dictonary_name>) ルールは true と評価されます。該当のディクショナリが存在しない場合は、ルールは false と評価されます。辞書の定義の詳細については（大文字と小文字の区別や単語境界の設定など）、「テキスト リソース」の章を参照してください。

次のフィルタは、シスコが「secret_words」という辞書にある単語を含むメッセージをスキャンすると、管理者にブラインド カーボン コピーを送信します。

```
copy_codenames:

if (dictionary-match ('secret_words')) {

bcc('administrator@example.com');

}
```

次の例では、メッセージの本文に、「secret_words」という辞書にあるいずれかの単語が含まれていると、そのメッセージが Policy という隔離エリアに送信されます。only-body-contains 条件とは異なり、body-dictionary-match 条件では、すべてのコンテンツ部分がそれぞれ個別に辞書に一致する必要はありません。各コンテンツ部分のスコア（マルチパート/代替部分も考慮されます）は合計されます。

```
quarantine_data_loss_prevention:
```

```
if (body-dictionary-match ('secret_words'))

{

quarantine('Policy');

}
```

次のフィルタでは、件名が指定した辞書にある単語と一致すると隔離されます。

```
quarantine_policy_subject:

if (subject-dictionary-match ('gTest'))

{

quarantine('Policy');

}
```

次の例では、「To」ヘッダーの電子メール アドレスを照合し、管理者にブラインド コピーを送信しています。

```
headerTest:

if (header-dictionary-match ('competitorsList', 'to'))

{

bcc('administrator@example.com');

}
```

`attachment-dictionary-match(<dictonary_name>)` ルールは上記の `dictionary-match` ルールと同様に機能しますが、検索対象は添付ファイルです。

次のフィルタでは、メッセージの添付ファイルに「secret_words」という辞書にあるいずれかの単語が含まれていると、そのメッセージが Policy という隔離エリアに送信されます。

```
quarantine_codenames_attachment:

if (attachment-dictionary-match ('secret_words'))

{

quarantine('Policy');

}
```

`header-dictionary-match(<dictonary_name>,<header>)` ルールは上記の dictionary-match ルールと同様に機能しますが、検索対象は *<header>* で指定したヘッダーです。ヘッダー名の大文字と小文字は区別されないため、たとえば「subject」でも「Subject」でも機能します。

次のフィルタでは、メッセージの「cc」ヘッダーに「ex_employees」という辞書にあるいずれかの単語が含まれていると、そのメッセージが Policy という隔離エリアに送信されます。

```
quarantine_codenames_attachment:

if (header-dictionary-match ('ex_employees', 'cc'))
```

```
{

quarantine('Policy');

}
```

辞書用語内でワイルドカードを使用することができます。電子メールアドレスのピリオドをエスケープする必要はありません。

SPF-Status ルール

SPF/SIDF 検証されたメールを受信する場合、SPF/SIDF 検証の結果によって異なるアクションを実行することが必要になる場合があります。spf-status ルールを使用すると、複数の SPF 検証結果との照合が可能になります。詳細については、検証結果を参照してください。

（注） SPF 識別情報なしで SPF 検証メッセージ フィルタ ルールを設定している場合、メッセージに判定が異なる別の SPF 識別情報が含まれているときは、そのルールは、メッセージ内の判定のいずれかがルールと一致するとトリガーされます。

SPF/SIDF 検証結果との照合を行うには、次の構文を使用します。

```
if (spf-status == "Pass")
```

1 つの条件で複数の状態判定に対してチェックする場合、次の構文を使用できます。

```
if (spf-status == "PermError, TempError")
```

さらに、次の構文を使用して、HELO、MAIL FROM、PRA ID に対して検証結果をチェックすることもできます。

```
if (spf-status("pra") == "Fail")
```

次の例に、spf-status フィルタの使用例を示します。

```
skip-spam-check-for-verified-senders:

if (sendergroup == "TRUSTED" and spf-status == "Pass"){

skip-spamcheck();

}

quarantine-spf-failed-mail:

if (spf-status("pra") == "Fail") {

if (spf-status("mailfrom") == "Fail"){

# completely malicious mail

quarantine("Policy");
```

```
} else {
if(spf-status("mailfrom") == "SoftFail") {
# malicious mail, but tempting
quarantine("Policy");
}
}
} else {
if(spf-status("pra") == "SoftFail"){
if (spf-status("mailfrom") == "Fail"
or spf-status("mailfrom") == "SoftFail"){
# malicious mail, but tempting
quarantine("Policy");
}
}
}
stamp-mail-with-spf-verification-error:
if (spf-status("pra") == "PermError, TempError"

or spf-status("mailfrom") == "PermError, TempError"
or spf-status("helo") == "PermError, TempError"){
# permanent error - stamp message subject
strip-header("Subject");
insert-header("Subject", "[POTENTIAL PHISHING] $Subject"); }
.
```

SPF-Passed ルール

次の例に、spf-passed とマークされていない電子メールを隔離するための `spf-passed` ルールを示します。

```
quarantine-spf-unauthorized-mail:
if (not spf-passed) {
quarantine("Policy");
}
```

（注） spf-status ルールと異なり spf-passed ルールは SPF/SIDF 検証値を簡単なブール値に単純化します。次の検証結果は、spf-passed ルールに合格していないものとして扱われます。None、Neutral、Softfail、TempError、PermError、Fail。より詳細な結果に基づいて、メッセージへのアクションを実行するには、spf-status ルールを使用します。

S/MIME ゲートウェイ メッセージ ルール

S/MIME ゲートウェイ メッセージ ルールでは、メッセージが S/MIME 署名されているか、暗号化されているか、または署名および暗号化されているかを確認します。次のメッセージフィルタでは、メッセージが S/MIME メッセージであるかどうかを確認し、S/MIME を使用した検証または復号に失敗した場合は隔離します。

```
quarantine_smime_messages:
if (smime-gateway-message and not smime-gateway-verified) {
quarantine("Policy");
}
```

詳細については、S/MIME セキュリティ サービスを参照してください。

S/MIME ゲートウェイ検証済みルール

S/MIME ゲートウェイ メッセージ検証済みルールでは、メッセージが正常に検証されているか、復号されているか、または復号および検証されているかを確認します。次のメッセージフィルタでは、メッセージが S/MIME メッセージであるかどうかを確認し、S/MIME を使用した検証または復号に失敗した場合は隔離します。

```
quarantine_smime_messages:
if (smime-gateway-message and not smime-gateway-verified) {
quarantine("Policy");
}
```

詳細については、S/MIME セキュリティ サービスを参照してください。

workqueue-count ルール

workqueue-count ルールは、ワークキュー数を特定の値と照合します。>、==、<= などのすべての比較演算子を使用できます。

次のフィルタは、ワークキュー数を確認し、指定した値より多ければスパムの確認を省略します。

```
wqfull:

if (workqueue-count > 1000) {

skip-spamcheck();

}
```

SPF/SIDF の詳細については、SPF および SIDF 検証の概要を参照してください。

SMTP 認証済みユーザー一致ルール

Cisco アプライアンスがメッセージの送信に SMTP 認証を使用している場合、`smtp-auth-id-matches` (*<target> [, <sieve-char>]*) ルールはメッセージのヘッダーとエンベロープ送信者を送信者の SMTP 認証ユーザ ID と照合し、スプーフィングされたヘッダーを含む送信メッセージを識別します。このフィルタを使用すると、なりすましの可能性のあるメッセージを隔離またはブロックできます。

`smtp-auth-id-matches` ルールは、SMTP 認証 ID を次の比較対象と比較します。

ターゲット（**Target**）	説明
*EnvelopeFrom	SMTP 対話のエンベロープ送信者のアドレス（MAIL FROM）を比較します。
*FromAddress	From ヘッダーから解析されたアドレスを比較します。From ヘッダーには複数のアドレスを使用できるため、そのうち 1 つが一致すれば一致と見なされます。
*Sender	Sender ヘッダーで指定されているアドレスを比較します。
*Any	ID にかかわらず、認証済み SMTP セッション中に作成されたメッセージと一致します。
*None	認証済み SMTP セッション中に作成されなかったメッセージと一致します。認証がオプションの場合に便利です（推奨）。

フィルタによる照合は厳密ではありません。大文字と小文字は区別されません。オプションで *sieve-char* パラメータが指定されている場合、特定の文字の後に続くアドレスの最後の部分は比較時に無視されます。たとえば、パラメータに「+」が含まれている場合、アドレス `joe+folder@example.com` のうち「+」より後の部分がフィルタでは無視されます。アドレスが `joe+smith+folder@example.com` の場合は、「`+folder`」のみが無視されます。SMTP 認証ユーザ ID 文字列が単純なユーザ名で、完全修飾電子メール アドレスでない場合は、比較対象のユーザ名部分のみが照合されます。ドメイン部分は別のルールで検証する必要があります。

また、$SMTPAuthID 変数を使用して SMTP 認証ユーザ ID をヘッダーに挿入することができます。

次の表は、SMTP 認証 ID と電子メールの比較の例で、`smtp-auth-id-matches` フィルタ ルールによる比較で一致するかどうかを示しています。

SMTP 認証 ID	ふるい文字	比較するアドレス	一致の可否
someuser		otheruser@example.com	なし
someuser		someuser@example.com	○
someuser		someuser@another.com	○
SomeUser		someuser@example.com	○

SMTP 認証 ID	ふるい文字	比較するアドレス	一致の可否
someuser		someuser+folder@example.com	なし
someuser	+	someuser+folder@example.com	○
someuser@example.com		someuser@forged.com	なし
someuser@example.com		someuser@example.com	○
SomeUser@example.com		someuser@example.com	○

次のフィルタは、認証済み SMTP セッション中に作成されたすべてのメッセージを確認し、From ヘッダーのアドレスとエンベロープ送信者が SMTP 認証ユーザ ID と一致するか検証します。アドレスと ID が一致すると、フィルタはドメインを許可します。一致しない場合、アプライアンスはメッセージを隔離します。

```
Msg_Authentication:

if (smtp-auth-id-matches("*Any"))

{

# Always include the original authentication credentials in a

# special header.

insert-header("X-Auth-ID","$SMTPAuthID");

if (smtp-auth-id-matches("*FromAddress", "+") and

smtp-auth-id-matches("*EnvelopeFrom", "+"))

{

# Username matches. Verify the domain

if header('from') != "(?i)@(?:example\\.com|alternate\\.com)" or

mail-from != "(?i)@(?:example\\.com|alternate\\.com)"

{

# User has specified a domain which cannot be authenticated

quarantine("forged");

}

} else {

# User claims to be an completely different user

quarantine("forged");

}

}
```

署名付きルール

signedルールはメッセージの署名を確認します。このルールは、メッセージの署名の有無を示すブール値を返します。このルールは、署名が ASN.1 DER エンコーディング ルールに従っているか、およびCMS署名データ型構造（RFC 3852、セクション5.1）に準拠しているかを評価します。署名がコンテンツと一致するかどうかは検証されず、証明書の有効性も確認されません。

次の例では、`signed` ルールを使用してヘッダーを署名済みメッセージに挿入します。

```
signedcheck: if signed { insert-header("X-Signed", "True"); }
```

次の例では、signedルールを使用して、特定の送信者グループから受信した未署名のメッセージの添付ファイルをドロップします。

```
Signed: if ((sendergroup == "NOTTRUSTED") AND NOT signed) {

html-convert();

if (attachment_size > 0)

{

drop_attachments("");

}

}
```

署名付き証明書ルール

signed-certificateルールは、X.509証明書発行者またはメッセージ署名者が、指定した正規表現と一致しているS/MIMEメッセージを選択します。このルールが対応しているのはX.509証明書のみです。

このルールの構文は signed-certificate (<field> [<operator> <regular expression>]) です。各項目の内容は次のとおりです。

- <field>：引用符で囲まれた文字列 “issuer”（発行者）または “signer”（署名者）。
- <operator>：== または !=。
- <regular expression>：発行者または署名者を照合するための値。

メッセージに複数の署名が使用されている場合、いずれかの発行者または署名者が正規表現と一致すると true が返されます。このルールを一番短い形で signed-certificate(“issuer”) および signed-certificate(“signer”) のように指定すると、S/MIMEメッセージに発行者または署名者が設定されている場合に true が返されます。

関連項目

- 署名者 （55 ページ）
- 発行元（Issuer） （55 ページ）
- 正規表現でのエスケープ処理 （55 ページ）

• $CertificateSigners アクション変数 （55 ページ）
• 例 1 （56 ページ）

署名者

メッセージ署名者に関して、このルールはX.509証明書のsubjectAltName拡張からrfc822Name名のシーケンスを抽出します。署名証明書にsubjectAltNameフィールドがない場合、またはこのフィールドにrfc822Name名がない場合、signed-certificate(“signer”) ルールは false を返します。まれではありますが、rfc822Name名が複数使用されている場合、このルールはすべての名前を正規表現と照合しようと試み、最初に一致した時点で true を返します。

発行元（Issuer）

発行者は X.509 証明書の空でない識別名です。AsyncOS は証明書から発行者を取得し、LDAP-UTF8 Unicode 文字列に変換します。次に例を示します。

- C=US,S=CA,O=IronPort
- C=US,CN=Bob Smith

X.509 証明書では発行者フィールドが必要なため、signed-certificate(“issuer”) は S/MIME メッセージに X.509 証明書があるかどうかを評価します。

正規表現でのエスケープ処理

LDAP-UTF8 では、正規表現で使用できるエスケープ方式が定義されています。LDAP-UTF8 での文字のエスケープ処理の詳細については、『Lightweight Directory Access Protocol (LDAP): String Representation of Distinguished Names』（http://www.ietf.org/rfc/rfc4514.txt）を参照してください。

signed-certificate ルールでのエスケープ ルールは、LDAP-UTF8 で定義されたエスケープ ルールとは異なり、エスケープ処理が必要な文字のみをエスケープします。LDAP-UTF8 では、エスケープ処理なしで表示できる文字をオプションでエスケープすることができます。たとえば、次の 2 つの文字列は、LDAP-UTF8 のエスケープ ルールではいずれも「Example, Inc.」を正しく表すものとされます。

- Example\, Inc.
- Example\,\ Inc\.

一方で、signed-certificate ルールでは「Example\, Inc.」のみが一致します。スペースやピリオドのエスケープ処理は LDAP-UTF8 では許可されていますが、必要ではないため、正規表現では許可されません。signed-certificate ルールで使用する正規表現を作成する場合は、エスケープ処理がなくても表示できる文字はエスケープしないでください。

$CertificateSigners アクション変数

アクション変数 `$CertificateSigners` は、署名証明書の `subjectAltName` 要素から取得した、カンマ区切り形式の署名者のリストです。1 人の署名者に複数の電子メールアドレスがある場合、重複を除去した上でリストに収録されます。

たとえば、Alice が自分の 2 つの証明書でメッセージに署名したとします。Bob は自分の 1 つの証明書でメッセージに署名しています。すべての証明書は 1 件の社内機関により発行されて

例 1

います。メッセージが S/MIME スキャンを通過すると、抽出されるデータには 3 つの項目が含まれます。

```
[
{
'issuer': 'CN=Auth,O=Example\, Inc.',
'signer': ['alice@example.com', 'al@private.example.com']
},
{
'issuer': 'CN=Auth,O=Example\, Inc.',
'signer': ['alice@example.com', 'al@private.example.com']
},
{
'issuer': 'CN=Auth,O=Example\, Inc.',
'signer': ['bob@example.com', 'bob@private.example.com']
}
]
```

`$CertificateSigners` 変数は次のように拡張されます。

```
"alice@example.com, al@private.example.com, bob@example.com, bob@private.example.com"
```

例 1

次の例では、証明書発行者が米国にいる場合、新しいヘッダーが挿入されます。

```
Issuer: if signed-certificate("issuer") == "(?i)C=US" {
insert-header("X-Test", "US issuer");
}
```

次の例では、署名者のドメインが example.com でない場合、管理者に通知されます。

```
NotOurSigners: if signed-certificate("signer") AND
signed-certificate("signer") != "example\\.com$" {
notify("admin@example.com");
}
```

次の例では、メッセージに X.509 証明書がある場合、ヘッダーが追加されます。

```
AnyX509: if signed-certificate ("issuer") {
insert-header("X-Test", "X.509 present");
```

```
}
```

次の例では、メッセージの証明書に署名者がない場合、ヘッダーが追加されます。

```
NoSigner: if not signed-certificate ("signer") {

insert-header("X-Test", "Old X.509?");

}
```

ヘッダー繰り返し回数ルール

ヘッダー繰り返し回数ルールは、任意の時点で次の条件のメッセージが指定された数だけ検出されると、true と判断します。

- 過去 1 時間以内に同じ件名のものが検出された。
- 過去 1 時間以内に同じエンベロープ送信者からのものが検出された。

このルールを使用することで、大量送信メールを検出できます。たとえば、特定の Web サイトで行われる政治キャンペーンで、組織に大量の電子メールが送信されることがあります。アンチスパム エンジンはこのような電子メールを正常なメールとして扱い、電子メールの配信は停止されません。

このルールの構文は header-repeats (<target>, <threshold> [, <direction>]) です。各項目の意味は次のとおりです。

- <target> には subject または mail-from を指定します。AsyncOS はターゲットの値の繰り返し回数をカウントします。
- <threshold> は、過去 1 時間に受信した、指定した target に同じ値を持つメッセージの数です。この数を超えると、ルールは true と評価されます。
- <direction> は incoming、outgoing、またはこの両方です。このルールで direction が指定されていない場合、着信メッセージと発信メッセージがルール評価対象としてカウントされます。

ヘッダー繰り返し回数ルールが true と評価されるたびに、システム アラートが送信されます。システム アラートを参照。

（注） ヘッダー フィールドにカンマまたはセミコロンで区切られた値が含まれている場合、ルールは完全な文字列をトラッキング対象とみなします。このルールでは、件名ヘッダーが空白のメッセージは無視されます。

ヘッダー繰り返し回数ルールは、変化するメッセージの合計数を 1 分単位の精度で維持します。このため、設定されているしきい値に達してからこのルールがトリガーされるまでに、1 分の遅れが生じることがあります。

関連項目

- ヘッダー繰り返し回数ルールとその他のルールの併用 （58 ページ）

• 例 （58 ページ）

ヘッダー繰り返し回数ルールとその他のルールの併用

ヘッダー繰り返し回数ルールとその他のルールを組み合わせて使用するには、AND 演算子または OR 演算子を使用します。たとえば、メッセージのサブセットをホワイトリストに追加するには、次のフィルタを使用します。

```
F1: if (recv_listener == 'Gray')  AND (header-repeats('subject', X, 'incoming') { drop();}
```

AND または OR 演算子を使用してヘッダー繰り返し回数ルールとその他のルールを組み合わせて使用する場合は、ヘッダー繰り返し回数ルールが必要な場合にだけ最後に評価されます。特定のメッセージに対してヘッダー繰り返し回数ルールが評価されない場合、subject または mail-from は指定されたしきい値との比較対象としてカウントされません。

ヘッダー繰り返し回数ルールは必要な場合に限り最後に評価されるため、OR 演算子で他のルールと組み合わせて使用される場合はこのルールの動作は異なります。次のフィルタの例では、OR 演算子を使用して署名付きルールとヘッダー繰り返し回数ルールが組み合わせられています。

```
f1: if signed OR (header-repeats('subject', 10)) { drop();}
```

この例では、このフィルタで処理される最初の 9 件のメッセージが同じ件名の署名付きメッセージである場合、ヘッダー繰り返し回数ルールはこれらのメッセージを処理しません。10 番目のメッセージが、9 番目までのメッセージと同じ件名ヘッダーの未署名メッセージである場合、しきい値に達していても、フィルタは設定されたアクションを実行します。

例

次の例では、任意の時点で、フィルタが過去 1 時間において同じ件名の着信メッセージを X 件以上検出した場合に、それ以降受信する同じ件名のメッセージが、ポリシー隔離に送信されます。

```
f1 : if header-repeats('subject', X, 'incoming') { quarantine('Policy');}
```

次の例では、フィルタが任意の時点で、過去 1 時間において同じエンベロープ送信者からの発信メッセージを X 件以上検出した場合に、それ以降同じエンベロープ送信者から送信されるメッセージがドロップされ、破棄されます。

```
f2 :  if header-repeats('mail-from', X, 'outgoing') {drop();}
```

次の例では、フィルタが任意の時点で、過去 1 時間において同じ件名の着信メッセージまたは発信メッセージを X 件以上検出した場合に、それ以降同じ件名を持つすべてのメッセージが管理者に通知されます。

```
f3: if header-repeats('subject', X) {notify('admin@xyz.com');}
```

URL レピュテーション ルール

URL レピュテーション ルールでは、メッセージに含まれている URL のレピュテーション スコアに基づいてメッセージ アクションを定義します。重要な詳細については、悪意のある URL または望ましくない URL からの保護の URL レピュテーションまたは URL カテゴリによるフィルタリング：条件およびルール を参照してください。

このルールの各部分は次のとおりです。

- `msg_filter_name` はこのメッセージ フィルタの名前です。
- `whitelist` は、（urllistconfig コマンドを使用して）定義されている URL リストの名前です。ホワイトリストの指定は任意です。

レピュテーション サービスからスコアが提供される場合にアクションを実行するには

`url-reputation` ルールを使用します。

url-reputation ルールを使用する場合のフィルタの構文を次に示します。

```
<msg_filter_name>:

if url-reputation('<min_score'>, <'max_score'>, '<whitelist>',
'<include_attachments>','<include_message_body_subject>')

{<action>}
```

場所：

- `min_score` および `max_score` は、アクション適用範囲の最小スコアと最大スコアです。指定する値は範囲に含まれます。

最小スコアと最大スコアは `-10.0` から `10.0` までの範囲内の数値である必要があります。

- メッセージの添付ファイル内の URL をスキャンするには、`include_attachments` を指定します。値「1」はメッセージ添付ファイルの URL スキャンが有効であり、値「0」はメッセージ添付ファイルの URL スキャンが有効でないことを示します。
- メッセージの本文と件名内の URL をスキャンするには、`include_message_body_subject` を指定します。値「1」はメッセージ本文と件名の URL スキャンが有効であり、値「0」はメッセージ本文と件名の URL スキャンが有効でないことを示します。

レピュテーション サービスからスコアが提供されない場合にアクションを実行するには

`url-no-reputation` ルールを使用します。

`url-no-reputation` ルールを使用する場合のフィルタの構文を次に示します。

```
<msg_filter_name>:

if url_no_reputation('<whitelist>',
'<include_attachments>','<include_message_body_subject>')

{<action>}
```

URL カテゴリ ルール

メッセージに含まれている URL のカテゴリに基づいてメッセージ アクションを定義するときに、URL カテゴリを使用します。重要な詳細については、悪意のある URL または望ましくない URL からの保護の URL レピュテーションまたは URL カテゴリによるフィルタリング：条件およびルール を参照してください。

`url-category` ルールを使用する場合のフィルタの構文を次に示します。

```
<msg_filter_name>: if url-category (['<category-name1>','<category-name2>',...,
'<category-name3>'],'<url_white_list>','<include_attachments>','<include_message_body_subject>')

<action>
```

場所：

- `msg_filter_name` はこのメッセージ フィルタの名前です。
- `action` はメッセージ フィルタ アクションです。
- `category-name` は URL カテゴリです。複数のカテゴリを指定する場合は、各カテゴリをカンマで区切ります。正しいカテゴリ名を確認するには、コンテンツ フィルタの URL カテゴリ条件またはアクションを確認してください。カテゴリの説明と例については、URL カテゴリについてを参照してください。
- `url_white_list` は、（urllistconfig コマンドを使用して）定義されている URL リストの名前です。
- メッセージの添付ファイル内の URL をスキャンするには、`include_attachments` を指定します。値「1」はメッセージ添付ファイルの URL スキャンが有効であり、値「0」はメッセージ添付ファイルの URL スキャンが有効でないことを示します。
- メッセージの本文と件名内の URL をスキャンするには、`include_message_body_subject` を指定します。値「1」はメッセージ本文と件名の URL スキャンが有効であり、値「0」はメッセージ本文と件名の URL スキャンが有効でないことを示します。

破損した添付ファイル ルール

破損した添付ファイル ルールは、破損している添付ファイルがメッセージに含まれている場合に true と評価します。破損した添付ファイルとは、スキャン エンジンがスキャンできないため破損として識別する添付ファイルのことです。

関連項目

- 例 （60 ページ）

例

次の例では、メッセージに含まれている破損した添付ファイルが検出されると、メッセージは Policy 隔離エリアに隔離されます。

```
quar_corrupt_attach: if (attachment-corrupt) { quarantine("Policy"); }
```

メッセージ言語ルール

メッセージの言語に基づいて異なるメッセージアクションを取る場合があります。たとえば、次のような場合があります。

- ロシアにあるメッセージにロシア語で免責事項を追加します
- 言語が確定できないメッセージをドロップします

メッセージ言語ルールを使用して、メッセージの件名と本文の言語に応じたメッセージ アクションを取ります。

（注） このルールでは、添付ファイルおよびヘッダーの言語は確認しません。

言語の検出動作の仕組み

Cisco E メール セキュリティ アプライアンスは、メッセージの言語を検出するのに組み込みの言語検出エンジンを使用します。アプライアンスは、件名とメッセージ本文を抽出し、言語検出エンジンに渡します。

言語検出エンジンは、抽出されたテキスト内の各言語の確率を決定し、それをアプライアンスに渡します。アプライアンスは、最も高い確率をもつ言語をメッセージの言語とみなします。アプライアンスは、次のシナリオのいずれかで、メッセージ言語を「未定」とみなします。

- 検出された言語が Cisco E メール セキュリティ アプライアンスでサポートされていない場合
- アプライアンスがメッセージの言語を検出できない場合
- 言語検出エンジンに送られた抽出されたテキストの合計サイズが 50 バイト未満の場合。

メッセージ フィルタの構文

```
<msg_filter_name>: if (message-language <operator> "<language1>, <language2>,..., <language n>") {<action>}
```

場所：

- `msg_filter_name` はこのメッセージ フィルタの名前です。
- 演算子は、`==` または `!=` です。
- `language` は、このメッセージフィルタに指定するメッセージ言語の値です。Separate multiple entries with commas. サポートされているメッセージ言語と値のリストについては、コンテンツ フィルタのメッセージ言語の条件を参照してください。値は、角かっこ（[]）で囲まれています。
- `action` はメッセージ フィルタ アクションです。

例

次の例では、言語が特定できなかったメッセージをドロップする方法を示しています。

```
DropMessagesWithUndeterminedLanguage: if (message-language == "unknown") { drop(); }
```

次の例では、ロシア語のメッセージにロシア語の免責事項を追加する方法を示しています。

```
ussianDisclaimerRule: if (message-language == "ru") { add-heading("RussianDisclaimer");
```

マクロ検出ルール

マクロ検出ルールを使用すると、メッセージに添付されたマクロが有効な添付ファイルを、指定したファイル タイプについて検出できます。

（注） アーカイブまたは埋め込みファイルにマクロが含まれている場合、親ファイルはメッセージからドロップされます。

マクロ検出構文

```
<msg_filter_name>: if (macro-detection-rule (['file_type-1', 'file_type-2',...
,'file_type-n'])) {<action>}
```

場所：

- msg_filter_name はこのメッセージ フィルタの名前です。
- file_type には、次のサポートされているファイル タイプのいずれかを指定できます。
  - Adobe Portable Document Format
  - Microsoft Office のファイル
  - OLE ファイル タイプ
- action はメッセージ フィルタ アクションです。

例

次の例は、マクロが有効なMicrosoft Office添付ファイルを含むメッセージをドロップする方法を示しています。

```
Drop_Messages_With_Macro-enabled_Office_Files: if (macro-detection-rule (['Microsoft
Office Files'])) { drop(); }
```

次の例では、マクロが有効な PDF 形式の添付ファイルを含むメッセージが特定のユーザに送信されると、そのメッセージはドロップされます。

```
Strip_Macro_enabled_PDF: if (rcpt-to == "joe@example.com") {
drop-macro-enabled-attachments(['Adobe Portable Document Format']); }
```

偽造メールの検出ルール

偽造された送信者アドレス（From: ヘッダー）を持つ不正なメッセージを検出し、そのようなメッセージに対してアクションを取ることが必要になる場合があります。

そのようなメッセージを検出するには、forged-email-detection ルールを使用します。このルールを設定する際には、コンテンツ ディクショナリと、メッセージに偽造の可能性があると見なすためのしきい値（1 ～ 100）を指定する必要があります。

forged-email-detection ルールは、From: ヘッダーをコンテンツ ディクショナリ内のユーザと比較します。このプロセス中に、類似により、アプライアンスは辞書内の各ユーザに類似性スコアを割り当てます。次に例を示します。

- From: ヘッダーが <j0hn.sim0ns@example.com> で、コンテンツ辞書に「John Simons」が含まれている場合、このユーザに 82 の類似性スコアが割り当てられます。
- From: ヘッダーが <john.simons@diff-example.com> で、コンテンツ辞書に「John Simons」が含まれている場合、このユーザに 100 の類似性スコアが割り当てられます。

類似性スコアが高くなればなるほど、メッセージが偽装されている確立が高くなります。類似性スコアが指定したしきい値以上の場合は、フィルタ アクションがトリガーされます。

詳細については、偽装メールの検出を参照してください。

メッセージ フィルタの構文

```
<filter_name>: if (forged-email-detection("<content_dictionary>", threshold)) {<action>;}
```

ここで、

- filter_name はメッセージ フィルタの名前です
- content_dictionary はコンテンツ ディクショナリの名前です
- threshold は、メッセージに偽造の可能性があるとみなすためのしきい値（1 ～ 100）です

例

次のメッセージフィルタは、メッセージ内の From: ヘッダーをディクショナリ内の用語と比較します。コンテンツディクショナリ内のユーザの類似性スコアが 70 以上である場合、このメッセージ フィルタは From: ヘッダーを削除し、エンベロープ送信者に置き換えます。

```
FED_CF: if (forged-email-detection("Execs", 70)) { fed("from", ""); }
```

重複境界検証ルール

duplicate_boundaries ルールを使用すると、重複する MIME 境界が含まれるメッセージを検出できます。

（注） 添付ファイルベースのルール（attachment-contains など）またはアクション（drop-attachments-where-contains など）は形式異常のメッセージ（重複する MIME 境界を含む）では動作しません。

メッセージ フィルタの構文

```
<filter_name>: if (duplicate_boundaries){<action>;}
```

例

次のメッセージ フィルタは、重複する MIME 境界が含まれるすべてのメッセージを隔離します。

```
DuplicateBoundaries: if (duplicate_boundaries) { quarantine("Policy"); }
```

不正な形式の MIME ヘッダー検出ルール

不正形式ヘッダー ルールを使用して、不正な形式の MIME ヘッダーを含むメッセージを検出できます。

メッセージ フィルタの構文

```
<filter_name>: if (malformed-header){<action>;}
```

例

次の例では、不正な形式の MIME ヘッダーがあるすべてのメッセージを隔離する方法を示しています。

```
quarantine_malformed_headers: if (malformed-header)
{
quarantine("Policy");
}
```

地理位置情報ルール

地理位置情報ルールを使用すると、選択した特定の国からの着信メッセージを処理できます。

地理位置情報構文

```
<msg_filter_name>: if (geolocation-rule (['country_name-1', 'country_name-2',...
,'country_name-n'])) {<action>}
```

場所：

- `msg_filter_name` はこのメッセージ フィルタの名前です。
- `country_name` は選択した国の名前です。
- `action` はメッセージ フィルタ アクションです。

例

次の例は、Country1 と Country2 から受信したメッセージを検疫する方法を示します。

```
Quarantine_Incoming_Messages_from_Country1_and_Country2: if (geolocation-rule
(['Country1', 'Country2'])) {quarantine("Policy");}
```

ETF のドメイン レピュテーション ルール

例として、以下のメッセージ フィルタ ルール構文を使用して、ETF エンジンを使用してメッセージ内の悪意のあるドメインを検出し、そのようなメッセージに対して適切な対応をします。

構文：

```
quarantine_msg_based_on_ETF: if (domain-external-threat-feeds (['etf_source1'],
 ['mail-from', 'from'], <'domain_exception_list'>)) { quarantine("Policy"); }
```

引数の説明

- `'domain-external-threat-feeds'` は、ドメイン レピュテーション メッセージ フィルタのルールです。
- `'etf_source1'`は、メッセージのヘッダーの悪意のあるドメインを検出するために使用される ETF ソースです。
- `'mail-from'`、`'from'` は、ドメインのレピュテーションを確認するために使用される必須ヘッダーです。
- `'domain_exception_list'` は、ドメインの例外リストの名前です。ドメインの例外リストが存在しない場合は「""」と表示されます。

例

以下の例では、‘Errors To:’ カスタム ヘッダーのドメインが ETF によって悪意があるとして検出された場合、メッセージが検疫されます。

```
Quarantining_Messages_with_Malicious_Domains: if domain-external-threat-feeds
(['threat_feed_source'], ['Errors-To'], "")) {quarantine("Policy");}
```

SDR のドメイン レピュテーション ルール

ドメイン レピュテーション ルールを使用して SDR に基づいてメッセージをフィルタ処理し、そのようなメッセージに対して適切な対応ができます。

- 送信者のドメインの判定
- 送信者のドメインの経過時間
- 送信者のドメインがスキャン不可

送信者ドメインの判定に基づいてメッセージをフィルタ処理

（注） 推奨されるブロッキングのしきい値は「Poor」です。SDR の詳細については、Cisco Talos（https://www.talosintelligence.com）にお問い合わせください。

構文：

```
drop_msg_based_on_sdr_verdict:
if sdr-reputation (['awful', 'poor'], "<domain_exception_list>")
{drop();}
```

それぞれの説明は次のとおりです。

- 'drop_msg_based_on_sdr_verdict' は、メッセージ フィルタの名前です。
- 'sdr-reputation' は、ドメイン レピュテーション メッセージ フィルタのルールです。
- 'awful'、'poor' は、SDR に基づいてメッセージをフィルタ処理するための送信者のドメイン判定の範囲です。
- 'domain_exception_list' は、ドメインの例外リストの名前です。ドメインの例外リストが存在しない場合は「""」と表示されます。
- 'drop' は、メッセージに適用されるアクションです。

例

以下のメッセージでは、SDR 判定が 'Unknownr' の場合、メッセージが検疫されます。

```
quarantine_unknown_sdr_verdicts:
if sdr-reputation (['unknown'], "")
{quarantine("Policy")}
```

送信者ドメインの経過時間に基づいてメッセージをフィルタ処理

構文：

```
<msg_filter_name>
if sdr-age (<‘unit'>, <‘operator'> <‘actual value’>)
{<action>}
```

それぞれの説明は次のとおりです。

- ‘sdr-reputation' は、ドメイン レピュテーション メッセージ フィルタのルールです。
- 'sdr_age'は、SDR に基づいてメッセージをフィルタ処理するために使用される送信者ドメインの経過時間です。
- ‘unit' は、送信者ドメインの経過時間に基づいてメッセージをフィルタ処理するための ‘days’、‘years’、‘months’、'weeks' オプションです。
- ‘operator' は、送信者ドメインの経過時間に基づいてメッセージをフィルタ処理するための比較演算子です。
  - – >（次の値より大きい）
  - – >=（次の値以上）
  - – <（次の値より小さい）
  - – <=（次の値以下）
  - – ==（次の値と等しい）

- – !=（次の値と等しくない）
- – Unknown

- 'actual value' は、送信者ドメインの経過時間に基づいてメッセージをフィルタ処理するために使用される数字です。

例

以下のメッセージでは、送信者ドメインの経過時間が不明な場合、メッセージはドロップされます。

```
Drop_Messages_Based_On_SDR_Age: if (sdr-age ("unknown", "")) {drop();}
```

以下のメッセージでは、送信者ドメインの経過時間が 1 ヵ月よりも短い場合、メッセージはドロップされます。

```
Drop_Messages_Based_On_SDR_Age: if (sdr-age ("months", <, 1, "")) { drop(); }
```

送信者ドメインのスキャン不可能性に基づいてメッセージをフィルタ処理

構文：

```
<msg_filter_name>
if sdr-unscannable (<'domain_exception_list'>)
{<action>}
```

それぞれの説明は次のとおりです。

- 'sdr-unscannable' は、ドメイン レピュテーション メッセージ フィルタのルールです。

  'domain_exception_list' は、ドメインの例外リストの名前です。ドメインの例外リストが存在しない場合は「""」と表示されます。

例

以下のメッセージでは、メッセージが SDR チェックに不合格の場合、メッセージが検疫されます。

```
Quarantine_Messages_Based_On_Sender_Domain_Unscannable: if (sdr-unscannable (""))
{quarantine("Policy");}
```

メッセージ フィルタ アクション

メッセージ フィルタの目的は、選択されたメッセージに対してアクションを実行することです。

アクションには、次の 2 つのタイプがあります。

- 最終アクション（deliver、drop、bounce など）はメッセージの処理を終了し、後続のフィルタによるさらなる処理を許可しません。
- 非最終アクションは、メッセージをさらに処理することを許可するアクションを実行します。

（注） 非最終メッセージ フィルタ アクションは、累積的です。各フィルタが異なるアクションを指定する複数のフィルタにメッセージが一致する場合、すべてのアクションが累積され、適用されます。ただし、同じアクションを指定する複数のフィルタにメッセージが一致する場合、前のアクションが上書きされ、最後のフィルタ アクションが適用されます。

関連項目

- フィルタ アクション一覧表 （68 ページ）
- アクション変数 （80 ページ）
- 一致した内容の表示 （83 ページ）
- メッセージ フィルタ アクションの説明と例 （84 ページ）

フィルタ アクション一覧表

メッセージ フィルタは、電子メール メッセージに対し、次の表に示すアクションを適用することができます。

表 *5:* メッセージ フィルタ アクション

操作	構文	説明
送信元ホストの変更	alt-src-host	メッセージの送信に使用する送信元ホスト名とIPインターフェイス（Virtual Gatewayアドレス）を変更します。送信元ホスト（Virtual Gatewayアドレス）変更アクション （93 ページ）を参照してください。
受信者の変更	alt-rcpt-to	メッセージの受信者を変更します。受信者変更アクション （92 ページ）を参照してください。
メールホストの変更	alt-mailhost	メッセージの送信先メール ホストを変更します。配信ホスト変更アクション （92 ページ）を参照してください。
通知	notify	メッセージに関する報告を別の受信者に送信します。通知およびコピー通知アクション （86 ページ）を参照してください。

操作	構文	説明
コピーの通知	notify-copy	notify アクションと同様ですが、bcc-scan アクションのようにコピーを送信します。通知およびコピー通知アクション （86 ページ）を参照してください。
BCC	bcc	メッセージをコピーし（メッセージ レプリケーション）、このコピーを匿名で別の受信者に送信します。ブラインド カーボン コピー アクション （89 ページ）を参照してください。
BCC（スキャン処理あり）	bcc-scan	メッセージを秘密で他の受信者に送信し、そのメッセージを新しいメッセージであるかのようにワーク キューで処理します。ブラインド カーボン コピー アクション （89 ページ）を参照してください。
アーカイブ（Archive）	archive	メッセージを mbox 形式のファイルにアーカイブします。アーカイブ アクション （94 ページ）を参照してください。
検疫（Quarantine）	`quarantine (quarantine_name`	*quarantine_name* で指定した隔離エリアにメッセージを送信するようフラグを設定します。隔離および複製アクション （91 ページ）を参照してください。
複製（隔離）	`duplicate-quarantine (quarantine_name`	指定された隔離エリアにメッセージのコピーを送信します。隔離および複製アクション （91 ページ）を参照してください。
ヘッダーの削除	strip-header	メッセージの配信前に、指定したヘッダーをメッセージから削除します。ヘッダー削除アクション （95 ページ）を参照してください。
ヘッダーの挿入	insert-header	メッセージの配信前に、ヘッダーと値の対をメッセージに挿入します。ヘッダー挿入アクション （95 ページ）を参照してください。

操作	構文	説明
ヘッダーテキストの編集	edit-header-text	指定したヘッダー テキストを、フィルタ条件として指定した文字列に置き換えます。ヘッダー テキスト編集アクション （96 ページ）を参照してください。
本文の編集	edit-body-text()	メッセージ本文から正規表現に一致する部分を削除し、指定したテキストに置き換えます。このフィルタは、メッセージ本文内の URL などの特定のコンテンツを削除および置換する場合に使用できます。本文編集アクション （96 ページ）を参照してください。
HTML の変換	html-convert()	メッセージ本文から HTML タグを削除し、メッセージのプレーン テキスト部分を残します。このフィルタは、メッセージ内のすべての HTML テキストをプレーン テキストに変換する場合に使用します。HTML 変換アクション （97 ページ）。
バウンス プロファイルの割り当て	bounce-profile	特定のバウンス プロファイルをメッセージに割り当てます。バウンス プロファイル アクション （98 ページ）を参照してください。
アンチスパム システムのバイパス	skip-spamcheck	Cisco システムのアンチスパム システムがメッセージに適用されないようにします。アンチスパム システムのバイパス アクション （98 ページ）を参照してください。

操作	構文	説明
グレイメール アクションのバイパス	skip-marketingcheck	マーケティング メールに対するアクションのバイパス。グレイメール アクションのバイパス （99 ページ）を参照してください。
	skip-socialcheck	ソーシャル ネットワーク メールに対するアクションのバイパス。グレイメール アクションのバイパス （99 ページ）を参照してください。
	skip-bulkcheck	バルク メールに対するアクションのバイパス。グレイメール アクションのバイパス （99 ページ）を参照してください。
アンチウイルス システムのバイパス	`skip-viruscheck`	Cisco システムのアンチウイルス システムがメッセージに適用されないようにします。アンチウイルス システムのバイパス アクション （99 ページ）を参照してください。
ファイル レピュテーション フィルタリングおよびファイル分析のバイパス	`skip-ampcheck`	このメッセージにファイル レピュテーション フィルタリングおよびファイル分析が適用されていないことを確認します。ファイル レピュテーション フィルタリングおよびファイル分析システムのバイパス アクション （100 ページ）を参照してください。
ウイルス アウトブレイク フィルタのスキャニング処理のスキップ	`skip-vofcheck`	このメッセージがウイルス アウトブレイク フィルタでスキャニング処理されないようにします。アンチウイルス システムのバイパス アクション （99 ページ）を参照してください。

操作	構文	説明
添付ファイルのドロップ（名前別）	`drop-attachments-by-name`	メッセージの添付ファイルのうち、指定した正規表現と一致する名前のファイルをすべてドロップします。一致するファイルが含まれている場合、アーカイブ形式の添付ファイル（zip、tar）、Microsoft Office の添付ファイル（doc、docx）、電子メールの添付ファイル（winmail.dat）はドロップされます。添付ファイルのスキャン メッセージ フィルタの例（113 ページ）を参照してください。
添付ファイルのドロップ（タイプ別）	`drop-attachments-by-type`	メッセージの添付ファイルのうち、指定した MIME タイプまたはファイル拡張子に該当する MIME タイプのファイルをすべてドロップします。アーカイブ形式の添付ファイル（zip、tar）内に該当するファイルがある場合、この添付ファイルはドロップされます。添付ファイルのスキャン メッセージ フィルタの例（113 ページ）を参照してください。
添付ファイルのドロップ（ファイル タイプ別）	`drop-attachments-by-filetype`	メッセージの添付ファイルのうち、指定したファイルの「フィンガープリント」と一致するファイルをすべてドロップします。アーカイブ形式の添付ファイル（zip、tar）内に該当するファイルがある場合、この添付ファイルはドロップされます。詳細については、添付ファイルのスキャン メッセージ フィルタの例（113 ページ）を参照してください。
添付ファイルのドロップ（MIME タイプ別）	`drop-attachments-by-mimetype`	メッセージの添付ファイルのうち、特定の MIME タイプのファイルをすべてドロップします。このアクションではファイル拡張子による MIME タイプの判別は行われず、アーカイブの内容の確認もされません。添付ファイルのスキャン メッセージ フィルタの例（113 ページ）を参照してください。

操作	構文	説明
添付ファイルのドロップ（サイズ別）	`drop-attachments-by-size`	メッセージの添付ファイルのうち、ロー エンコード形式で指定したサイズ（バイト単位）以上のサイズであるファイルをすべてドロップします。アーカイブや圧縮ファイルの場合、このアクションでは非圧縮状態でのサイズは計測されず、デコードを行う前の実際の添付ファイルのサイズが計測されます。添付ファイルのスキャン メッセージ フィルタの例（113 ページ）を参照してください。
添付ファイルのドロップ（内容別）	`drop-attachments-where-contains`	メッセージの添付ファイルのうち、指定した正規表現を含むファイルをすべてドロップします。パターンの発生回数が、しきい値で指定した最小回数以上である必要があります。アーカイブ ファイル（zip、tar）は、中に含まれているファイルのいずれかが正規表現と一致する場合にドロップされます。添付ファイルのスキャン メッセージ フィルタの例（113 ページ）を参照してください。 オプション コメントは、ドロップされた添付ファイルの置換に使用されるテキストを変更します。添付ファイルのフッターは、単純にメッセージに追加されるだけです。

操作	構文	説明
マクロが含まれる添付ファイルのドロップ	drop-macro-enabled-attachments	指定したファイル タイプのマクロが有効になった添付ファイルをすべてドロップします。 （注） アーカイブまたは埋め込みファイルにマクロが含まれている場合、親ファイルはメッセージからドロップされます。 **構文** drop-macro-enabled-attachments (['file_type-1', 'file_type-2', …,'file_type-n'], "custom_replacement_message") ここで、 • file_type には、次のサポートされているファイル タイプのいずれかを指定できます。 • Adobe Portable Document Format • Microsoft Office のファイル • OLE ファイル タイプ • **カスタム差し替え**メッセージとは、添付ファイルが削除されるときはメッセージ本文の一番下に既定のシステム生成メッセージが追加されますが、それに代わって追加される任意のメッセージです。 マクロ検出ルール （62 ページ）を参照してください
添付ファイルのドロップ（辞書との一致別）	drop-attachments-where-dictionary-match	辞書の用語との一致に基づいて添付ファイルを削除します。添付ファイルであると判断される MIME 部分の用語が辞書の用語と一致する場合（かつ、ユーザ定義のしきい値に達している場合）、添付ファイルが電子メールから削除されます。添付ファイルのスキャン メッセージ フィルタの例 （113 ページ）を参照してください。

操作	構文	説明
フッターの追加	`add-footer(footer-name)`	メッセージのフッターとして免責条項を追加します。詳細については、「テキスト リソース」の章の「メッセージ免責事項スタンプ」を参照してください。
見出しの追加	`add-heading(heading-name)`	メッセージの見出しとして免責条項を追加します。詳細については、「テキスト リソース」の章の「メッセージ免責事項スタンプ」を参照してください。
配信時の暗号化	`encrypt-deferred`	配信時にメッセージを暗号化します。メッセージはそのまま次の処理に進み、すべての処理が完了した時点で暗号化され、配信されます。
配信時の S/MIME 署名/暗号化	`smime-gateway-deferred ("sending_profile")`	配信時に、指定された送信プロファイルを使用して、メッセージの S/MIME 署名または暗号化を実行します配信時の S/MIME 署名/暗号化アクション （86 ページ）を参照してください。
S/MIME 署名/暗号化	`smime-gateway("sending_profile")`	指定された送信プロファイルを使用して S/MIME 署名または暗号化を実行してメッセージを配信し、その後の処理はスキップします。S/MIME 署名または暗号化アクション （86 ページ）を参照してください。
メッセージ タグの追加	`tag-message(tag-name)`	DLP ポリシー フィルタリングで使用するカスタム用語をメッセージに追加します。DLP ポリシーを設定して、スキャン対象をメッセージ タグがあるメッセージに限定することができます。メッセージ タグは受信者側では表示されません。メッセージ タグ追加アクション （101 ページ）と「データ消失防止」の章を参照してください。

操作	構文	説明
ログエントリの追加	`log-entry`	カスタマイズしたテキストを、テキスト メール ログに INFO レベルで追加します。このテキストにはアクション変数を使用することができます。ログエントリはメッセージ トラッキングに表示されます。ログ エントリ追加アクション （101 ページ）を参照してください。
URL レピュテーションに基づき URL をテキストに置換	• `url-reputation-replace` • `url-no-reputation-replace`	URL またはその動作を URL のレピュテーションに基づいて変更します。 レピュテーション サービスから URL のスコアが提供されない状況を処理するには、個別のアクションを使用します。 URL レピュテーション アクション （101 ページ）を参照してください。
URL レピュテーションに基づき URL の危険を取り除く	• `url-reputation-defang` • `url-no-reputation-defang`	
URL レピュテーションに基づいてシスコのセキュリティ プロキシに URL をリダイレクト	• `url-reputation-proxy-redirect` • `url-no-reputation-proxy-redirect`	
URL カテゴリに基づき URL をテキストに置換	`url-category-replace`	URL またはその動作を URL のカテゴリに基づいて変更します。 URL カテゴリ アクション （104 ページ）を参照してください。
URL カテゴリに基づき URL の危険を取り除く	`url-category-defang`	
URL カテゴリに基づき Cisco セキュリティ プロキシに URL をリダイレクトする	`url-category-proxy-redirect`	
偽装メールの検出	`fed`	偽装されたメッセージから From: ヘッダーを削除し、エンベロープ送信者で置き換えます。偽造メールの検出アクション （105 ページ）を参照してください。

操作	構文	説明
オペレーションなし	`no-op`	操作は実行されません。オペレーションなし （105 ページ）を参照してください。
*残りのメッセージ フィルタをスキップ	`skip-filters`	メッセージに対して他のメッセージ フィルタによる処理は行われず、メッセージは電子メール パイプラインをそのまま通過します。「残りのメッセージフィルタをスキップ」アクション （84 ページ）を参照してください。
*メッセージのドロップ	`drop`	メッセージをドロップし、廃棄します。ドロップ アクション （85 ページ）を参照してください。
*メッセージのバウンス	`bounce`	メッセージを送信者に戻します。バウンス アクション （85 ページ）を参照してください。
*すぐに暗号化して配信	`encrypt`	Cisco Email Encryption を使用して、送信メッセージを暗号化します。暗号化アクション （86 ページ）を参照してください。
* 最終アクション		

関連項目

- 添付ファイル グループ （77 ページ）

添付ファイル グループ

特定のファイル タイプ（「exe」など）や一般的な添付ファイルのグループを `attachment-filetype` ルールや `drop-attachments-by-filetype rules` ルールで指定できます。AsyncOS は添付ファイルを以下の表に記載されているグループに分類します。

特定のファイルタイプの添付ファイルを含まないメッセージと照合させる `!=`演算子を使うメッセージ フィルタを作成する場合は、フィルタで除外するファイル タイプの添付ファイルが少なくとも 1 つあると、フィルタはメッセージへのアクションを実行しません。たとえば、次のフィルタは **`.exe`** ファイル タイプではない添付ファイルを含むメッセージをドロップします。

```
exe_check: if (attachment-filetype != "exe") {

drop();

}
```

メッセージに複数の添付ファイルがある場合、E メール セキュリティ アプライアンスは他の添付ファイルが .exe ファイルでない場合でも、添付ファイルの少なくとも 1 つが .exe ファイルの場合はメッセージをドロップしません。

表 *6:* 添付ファイル グループ

添付ファイルグループ名	スキャン対象のファイル タイプ
マニュアル	• doc • docx • mdb • mpp • ole • pdf • ppt • pptx • rtf • wps • x-wmf • xls • xlsx
実行可能ファイル	• exe • java • msi • pif （注） Executable グループをフィルタリングすると、.dll ファイルと .scr ファイルもスキャンされます。これらのファイル タイプは個別にスキャンできません。

添付ファイルグループ名	スキャン対象のファイル タイプ
圧縮	• ace（ACE アーカイバ圧縮ファイル） • arc（SQUASH 圧縮アーカイブ） • arj（Robert Jung ARJ 圧縮アーカイブ） • binhex • bz（Bzip 圧縮ファイル） • bz2（Bzip 圧縮ファイル） • cab（Microsoft キャビネット ファイル） • gzip*（圧縮ファイル - UNIX gzip） • lha（圧縮アーカイブ [LHA/LHARC/LZH]） • rar（圧縮アーカイブ） • sit（圧縮アーカイブ - Macintosh ファイル [Stuffit]） • tar*（圧縮アーカイブ） • unix（UNIX 圧縮アーカイブ） • zip*（圧縮アーカイブ - Windows） • zoo（ZOO 圧縮アーカイブ ファイル） * これらのファイルは「本文スキャン」の対象にすることができます。
テキスト（Text）	• txt • html • xml
画像	• bmp • cur • gif • ico • jpeg • pcx • png • psd • psp • tga • tiff

添付ファイルグループ名	スキャン対象のファイル タイプ
メディア	• aac • aiff • asf • avi • flash • midi • mov • mp3 • mpeg • ogg • ram • snd • wav • wma • wmv

アクション変数

bcc()、bcc-scan()、notify()、notify-copy()、add-footer()、add-heading()、insert-headers() の各アクションには、アクションの実行時に元のメッセージの情報に自動的に置き換えられる所定の変数を使用しているパラメータがあります。これらの特殊な変数はアクション変数と呼ばれます。Cisco アプライアンスでは次のアクション変数がサポートされています。

表 7: メッセージ フィルタ アクション変数

変数	構文	説明
すべてのヘッダー（All Headers）	`$AllHeaders`	メッセージのヘッダーを返します。
本文サイズ（Body Size）	`$BodySize`	メッセージのサイズをバイト単位で返します。
証明書の署名者（Certificate Signers）	`$CertificateSigners`	署名付き証明書の subjectAltName 要素から取得した署名者を返します。詳細については、$CertificateSigners アクション変数 （55 ページ）を参照してください。
日付（Date）	`$Date`	現在の日付を MM/DD/YYYY 形式で返します。
ドロップされたファイル名（Dropped File Name）	`$dropped_filename`	直近にドロップされたファイル名のみを返します。

変数	構文	説明
ドロップされたファイル名（Dropped File Names）	`$dropped_filenames`	ドロップされたファイルのリストを表示します（$filenames と同様です）。
ドロップされたファイル タイプ（Dropped File Types）	`$dropped_filetypes`	ドロップされたファイルのタイプを表示します（$filetypes と同様です）。
エンベロープ送信者（Envelope Sender）	`$EnvelopeFrom`	メッセージのエンベロープ送信者（Envelope From、<MAIL FROM>）を返します。
エンベロープ受信者（Envelope Recipients）	`$EnvelopeRecipients`	メッセージのすべてのエンベロープ受信者（Envelope To、<RCPT TO>）を返します。
ファイル名（File Names）	`$filenames`	メッセージの添付ファイルの名前のリストをカンマ区切りで返します。
ファイルサイズ（File Sizes）	`$filesizes`	メッセージの添付ファイルのサイズのリストをカンマ区切りで返します。
ファイルタイプ（File Types）	`$filetypes`	メッセージの添付ファイルのタイプのリストをカンマ区切りで返します。
フィルタ名（Filter Name）	`$FilterName`	処理中のフィルタの名前を返します。
GMT 日時（GMTimeStamp）	`$GMTimeStamp`	メッセージの Received: 行に表示される現在の日時を GMT 形式で返します。
HATグループ名（HAT Group Name）	`$Group`	メッセージの送信時に送信者が属していた送信者グループの名前を返します。送信者グループに名前がない場合は、文字列「>Unknown<」が挿入されます。
一致した内容（Matched Content）	`$MatchedContent`	スキャン フィルタ ルール（body-contains などのフィルタ ルールやコンテンツ ディクショナリを含む）をトリガーした内容を返します。
メールフローポリシー（Mail Flow Policy）	`$Policy`	メッセージの送信時に送信者に適用された HAT ポリシーの名前を返します。事前に定義されているポリシー名が使用されていない場合、文字列「>Unknown<」が挿入されます。

変数	構文	説明
ヘッダー（Header）	`$Header['string']`	引用符で囲まれたヘッダーの値を返します（元のメッセージに該当するヘッダーがある場合）。二重引用符が使用される場合もあります。
ホストネーム	`$Hostname`	Ciscoアプライアンスのホスト名を返します。
内部メッセージID（Internal Message ID）	`$MID`	内部でメッセージを識別するため使用されているメッセージ ID（MID）を返します。RFC822「Message-Id」の値とは異なるため注意してください（「Message-Id」を取得するには $Header を使用します）。
受信リスナー（Receiving Listener）	`$RecvListener`	メッセージを受信したリスナーのニックネームに置き換えられます。
受信インターフェイス（Receiving Interface）	`$RecvInt`	メッセージを受信したインターフェイスのニックネームを返します。
リモート IP アドレス（Remote IP Address）	`$RemoteIP`	Cisco アプライアンスにメッセージを送信したシステムの IP アドレスを返します。
リモートホストアドレス（Remote Host Address）	`$remotehost`	Cisco アプライアンスにメッセージを送信したシステムのホスト名を返します。
SenderBase レピュテーション スコア	`$Reputation`	送信者の SenderBase レピュテーション スコアを返します。レピュテーション スコアがない場合は「None」に置き換えられます。
Subject	`$Subject`	メッセージの件名を返します。
時刻（Time）	`$Time`	現在地の時間帯での現在時刻を返します。
Timestamp	`$Timestamp`	メッセージの Received: 行に表示される現在の日時を現在地の時間帯に従って返します。

関連項目

- 非 ASCII 文字セットとメッセージ フィルタ アクション変数 （82 ページ）

非 ASCII 文字セットとメッセージ フィルタ アクション変数

このシステムでは、ISO-2022 スタイル文字コード（ヘッダー値で使用されるエンコードのスタイル）を含むアクション変数の拡張をサポートしています。また、通知内で多言語テキストを

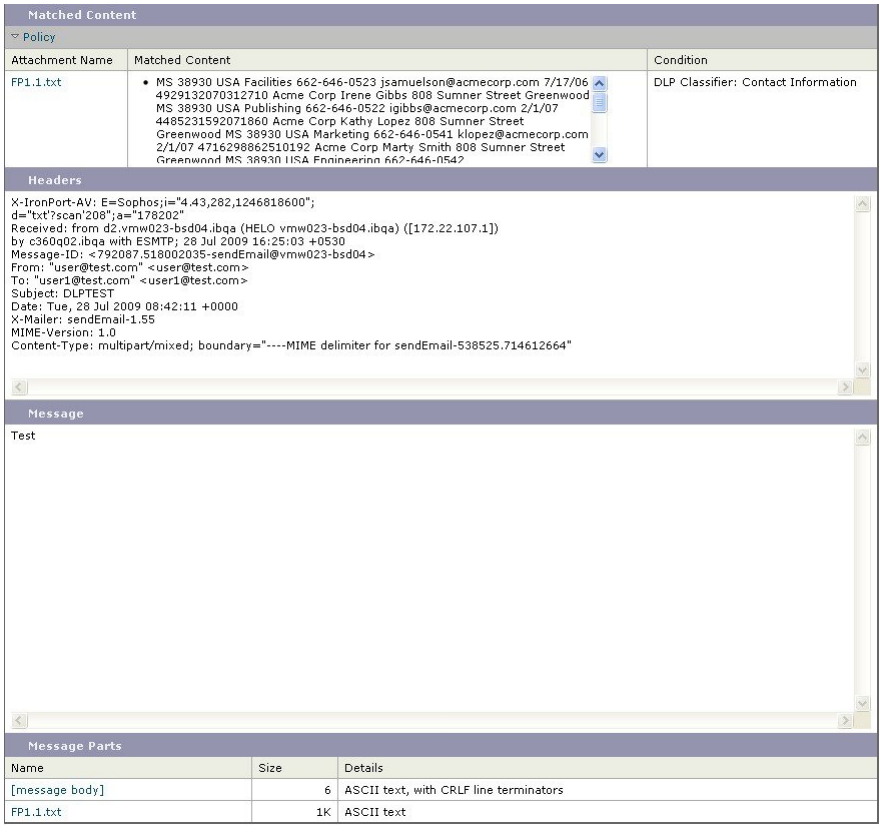
使用できます。これらの内容が統合されて通知が生成され、UTF-8 形式の、引用符で囲まれた印刷可能なメッセージとして送信されます。

一致した内容の表示

Attachment Content 条件、Message Body または Attachment 条件、Message 本文条件、または Attachment 内容条件と一致するメッセージに対して隔離アクションを設定した場合、隔離されたメッセージ内の一致した内容を表示できます。メッセージ本文を表示すると、一致した内容が黄色で強調表示されます。また、$MatchedContent アクション変数を使用して、一致した内容をメッセージの件名に含めることができます。

メッセージフィルタまたはコンテンツ フィルタのルールをトリガーしたローカル隔離内のメッセージを表示すると、フィルタ アクションを実際にはトリガーしなかった内容が（フィルタ アクションをトリガーした内容と共に）GUI で表示されることがあります。GUI の表示は、該当コンテンツを特定するための目安として使用するもので、該当コンテンツの完全なリストであるとは限りません。これは、GUI で使用される内容一致ロジックが、フィルタで使用されるものほど厳密ではないため起こります。この問題はメッセージ本文での検索についてのみ発生します。メッセージの各パート内の一致文字列をそれに対応するフィルタ ルールと共に一覧表示するテーブルは正しく表示されます。

図 *2 : Policy* 検疫エリア内で表示された一致内容



メッセージ フィルタ アクションの説明と例

次のセクションでは、使用されるさまざまなメッセージフィルタアクションについて説明し、その例を示します。

- 「残りのメッセージ フィルタをスキップ」アクション （84 ページ）
- ドロップ アクション （85 ページ）
- バウンス アクション （85 ページ）
- 暗号化アクション （86 ページ）
- 通知およびコピー通知アクション （86 ページ）
- ブラインド カーボン コピー アクション （89 ページ）
- 隔離および複製アクション （91 ページ）
- 受信者変更アクション （92 ページ）
- 配信ホスト変更アクション （92 ページ）
- 送信元ホスト（Virtual Gateway アドレス）変更アクション （93 ページ）
- アーカイブ アクション （94 ページ）
- ヘッダー削除アクション （95 ページ）
- ヘッダー挿入アクション （95 ページ）
- ヘッダー テキスト編集アクション （96 ページ）
- 本文編集アクション （96 ページ）
- HTML 変換アクション （97 ページ）
- バウンス プロファイル アクション （98 ページ）
- アンチスパム システムのバイパス アクション （98 ページ）
- グレイメール アクションのバイパス （99 ページ）
- アンチウイルス システムのバイパス アクション （99 ページ）
- ファイル レピュテーション フィルタリングおよびファイル分析システムのバイパス アクション （100 ページ）
- アンチウイルス システムのバイパス アクション （99 ページ）
- メッセージ タグ追加アクション （101 ページ）
- ログ エントリ追加アクション （101 ページ）
- URL レピュテーション アクション （101 ページ）
- URL カテゴリ アクション （104 ページ）
- オペレーションなし （105 ページ）
- 偽造メールの検出アクション （105 ページ）

「残りのメッセージ フィルタをスキップ」アクション

`skip-filters` アクションを実行すると、メッセージフィルタによるメッセージの処理がスキップされ、メッセージは電子メールパイプラインを通過します。アプライアンスでアンチスパムスキャンとアンチウイルス スキャンが使用できる場合、`skip-filters` アクションを実行したメッセージはこれらのスキャンの対象となります。`skip-filters` アクションは、メッセージフィルタのデフォルトの最終アクションです。

次のフィルタは、customercare@example.com に通知を送信し、boss@admin 宛てのメッセージをただちに送信します。

```
bossFilter:

if(rcpt-to == 'boss@admin$')

{

notify('customercare@example.com');

skip-filters();

}
```

ドロップ アクション

drop アクションを実行すると、メッセージは送信されずに破棄されます。メッセージは送信者には戻されず、メッセージの本来の宛先にも送信されず、それ以外の処理も一切行われません。

次のフィルタは、まず george@whitehouse.gov に通知を送信し、その後件名が「SPAM」で始まるメッセージを破棄します。

```
spamFilter:

if(subject == '^SPAM.*')

{

notify('george@whitehouse.gov');

drop();

}
```

バウンス アクション

bounce アクションは、メッセージを送信者（エンベロープ送信者）に戻し、それ以降の処理は行いません。

次のフィルタは、@yahoo\\.com で終わる電子メールアドレスから送信されたすべてのメッセージを戻します（バウンスします）。

```
yahooFilter:

if(mail-from == '@yahoo\\.com$')

{

bounce();

}
```

暗号化アクション

encrypt アクションは、設定された暗号化プロファイルを使用して、電子メール受信者に暗号化されたメッセージを送信します。

次のフィルタは、メッセージの件名に [encrypt] という語句が含まれている場合に、そのメッセージを暗号化します。

```
Encrypt_Filter:

if ( subject == '\\[encrypt\\]' )

{

encrypt('My_Encryption_Profile');

}
```

（注） このフィルタ アクションを使用するには、ネットワークにCisco暗号化アプライアンスがあるか、ホスト キー サービスが設定されている必要があります。また、このフィルタ アクションを使用するには、暗号化プロファイルの設定が必要です。

配信時の S/MIME 署名/暗号化アクション

smime-gateway-deferredアクションでは、配信時に、指定された送信プロファイルを使用して、メッセージのS/MIME署名または暗号化を実行しますメッセージは次の処理段階に進み、すべての処理が完了した時点で署名または暗号化されて、配信されます。

次のフィルタでは、配信時に、特定の送信者からのすべての発信メッセージに対して S/MIME 暗号化を実行します。

```
smime-deferred:if(mail-from ==
"user@example.com"){smime-gateway-deferred("smime-encrypt");}
```

S/MIME 署名または暗号化アクション

smime-gateway アクションでは、指定された送信プロファイルを使用して S/MIME 署名または暗号化を実行してメッセージを配信し、その後の処理はスキップします。

次のフィルタでは、特定の送信者からのすべての発信メッセージに対してS/MIME暗号化を実行して、即時に配信します。

```
smime-deliver-now:if(mail-from == "user@example.com"){smime-gateway("smime-sign");}
```

通知およびコピー通知アクション

notify および notify-copy アクションは、指定した電子メールに対して、メッセージの概要を電子メールで送信します。notify-copy アクションは、bcc-scan アクションと同様に、元のメッセージのコピーも送信します。通知概要には次の内容が含まれます。

- メッセージのメール転送プロトコル対話から取得したエンベロープ送信者およびエンベロープ受信者（MAIL FROM および RCPT TO）指定の内容。
- メッセージのヘッダー。
- メッセージを検出したメッセージ フィルタの名前。

受信者、件名行、送信元アドレス、および通知テンプレートを指定できます。次のフィルタは、サイズが 4 MB を超えるメッセージを選択し、一致するメッセージのそれぞれについて通知メッセージを admin@example.com に送信し、最後にメッセージを破棄します。

```
bigFilter:

if(body-size >= 4M)

{

notify('admin@example.com');

drop();

}
```

または

```
bigFilterCopy:

if(body-size >= 4M)

{

notify-copy('admin@example.com');

drop();

}
```

エンベロープ受信者パラメータとして、有効な任意の電子メール アドレス（上の例では admin@example.com）を指定できます。また、メッセージのすべてのエンベロープ受信者を指定するアクション変数 $EnvelopeRecipients（アクション変数 （80 ページ）を参照）を指定することもできます。

```
bigFilter:

if(body-size >= 4M)

{

notify('$EnvelopeRecipients');

drop();

}
```

notify アクションでは最大で 3 つのオプション引数を使用でき、件名ヘッダー、エンベロープ送信者、通知メッセージに使用する定義済みテキスト リソースを指定できます。これらのパラ

メータはこの順序で指定する必要があるため、エンベロープ送信者を設定する場合や通知テンプレートを指定する場合は件名を指定する必要があります。

件名パラメータにはアクション変数（アクション変数（80ページ）を参照）を指定できます。この変数は元のメッセージから取得したデータで置き換えられます。デフォルトでは、件名は「Message Notification」に設定されています。

エンベロープ送信者パラメータとして、有効な任意の電子メールアドレスを指定できます。また、メッセージのリターンパスを元のメッセージと同じに設定する $EnvelopeFrom アクション変数を指定することもできます。

通知テンプレート パラメータは、既存の通知テンプレートの名前になります。詳細については、通知 （113 ページ）を参照してください。

次の例は前の例を拡張したものですが、件名が「[bigFilter] Message too large」となるように変更し、リターンパスを元の送信者に設定し、「message.too.large」テンプレートを使用しています。

```
bigFilter:

if (body-size >= 4M)

{

notify('admin@example.com', '[$FilterName] Message too large',

'$EnvelopeFrom', 'message.too.large');

drop();

}
```

また、$MatchedContent アクション変数を使用して、送信者または管理者にコンテンツ フィルタがトリガーされたことを通知することもできます。$MatchedContent アクション変数は、フィルタをトリガーしたコンテンツを表示します。たとえば、次のフィルタは、電子メールにABAアカウント情報が含まれる場合に、管理者に通知します。

```
ABA_filter:

if (body-contains ('*aba')){

notify('admin@example.com','[$MatchedContent]Account Information Displayed');

}
```

関連項目

- Notification Template （88 ページ）

Notification Template

[テキストリソース（Text Resources）] ページまたは textconfig CLI コマンドを使用して、notify() および notify-copy() アクションで使用するテキストリソースとなるカスタム通知テ

ンプレートを設定できます。カスタム通知テンプレートを作成しない場合、デフォルトのテンプレートが使用されます。デフォルトのテンプレートにはメッセージ ヘッダーが含まれますが、デフォルトではカスタム通知テンプレートにはメッセージヘッダーは含まれません。カスタム通知にメッセージ ヘッダーを含めるには、`$AllHeaders` アクション変数を使用します。

詳細については、「テキスト リソース」の章を参照してください。

次の例では、メッセージのサイズが大きい場合に次のフィルタがトリガーされると、本来の受信者に対して、メッセージが大きすぎることを示す電子メールが送信されます。

```
bigFilter:

if (body-size >= 4M)

{

notify('$EnvelopeRecipients', '[$FilterName] Message too large',

'$EnvelopeFrom', 'message.too.large');

drop();

}
```

ブラインド カーボン コピー アクション

`bcc` アクションは、メッセージの無記名コピーを、指定した受信者に送信します。この処理はメッセージ レプリケーションとも呼ばれています。元のメッセージにはコピーに関する通知は含まれず、無記名コピーが受信者にバウンスされることはないため、メッセージの元の送信者と受信者はコピーが送信されたことを関知しない場合があります。

次のフィルタは、`johnny` から sue に送信されるメッセージのそれぞれについて、ブラインド カーボン コピーを `mom@home.org` に送信します。

```
momFilter:

if ((mail-from == '^johnny$') and (rcpt-to == '^sue$'))

{

bcc('mom@home.org');

}
```

bcc アクションでは最大で 3 つのオプション引数を使用でき、コピーしたメッセージに使用する件名ヘッダーとエンベロープ送信者、およびalt-mailhostを指定できます。これらのパラメータはこの順序で指定する必要があるため、エンベロープ送信者を設定する場合は件名を指定する必要があります。

件名パラメータにはアクション変数（アクション変数（80ページ）を参照）を指定できます。この変数は元のメッセージから取得したデータで置き換えられます。デフォルトでは、元のメッセージの件名（`$Subject` と同じ内容）が設定されます。

エンベロープ送信者パラメータとして、有効な任意の電子メールアドレスを指定できます。また、メッセージのリターンパスを元のメッセージと同じに設定する $EnvelopeFrom アクション変数を指定することもできます。

次の例は前の例を拡張したもので、件名は「[Bcc] <original subject>」に設定され、リターンパスは badbounce@home.org に設定されています。

```
momFilter:

if ((mail-from == '^johnny$') and (rcpt-to == '^sue$'))

{

bcc('mom@home.org', '[Bcc] $Subject', 'badbounce@home.org');

}
```

4 番めのパラメータは alt-mailhost です。

```
momFilterAltM:

if ((mail-from == '^johnny$') and (rcpt-to == '^sue$'))

{

bcc('mom@home.org', '[Bcc] $Subject', '$EnvelopeFrom',

'momaltmailserver.example.com');

}
```

注意 Bcc()、notify()、bounce() の各フィルタ アクションを実行すると、ネットワーク内にウイルスが侵入する場合があります。ブラインド カーボン コピー フィルタ アクションは、元のメッセージの完全なコピーであるメッセージを新規作成します。通知フィルタ アクションは、元のメッセージのヘッダーを含むメッセージを新規作成します。まれにではありますが、ヘッダーにウイルスが含まれている場合があります。バウンス フィルタ アクションは、元のメッセージの最初の 10 キロバイトを含むメッセージを新規作成します。3 つのうちいずれの場合についても、新しいメッセージはアンチウイルス スキャンやアンチスパム スキャンの処理対象とはなりません。

複数のホストに送信する場合は、bcc() アクションを複数回呼び出すことができます。

```
multiplealthosts:

if (recv-listener == "IncomingMail")

{

insert-header('X-ORIGINAL-IP', '$remote_ip');

bcc ('$EnvelopeRecipients', '$Subject', '$EnvelopeFrom', '10.2.3.4');
```

```
bcc ('$EnvelopeRecipients', '$Subject', '$EnvelopeFrom', '10.2.3.5');

bcc ('$EnvelopeRecipients', '$Subject', '$EnvelopeFrom', '10.2.3.6');

}
```

関連項目

- 競合他社に送信されたメールの BCC およびスキャン （134 ページ）

bcc-scan() アクション

bcc-scan アクションは bcc アクションと同様に機能しますが、送信されるメッセージは新しいメッセージとして扱われるため、電子メール パイプライン全体を経由して送信されます。

```
momFilter:

if ((mail-from == '^johnny$') and (rcpt-to == '^sue$'))

{

bcc-scan('mom@home.org');

}
```

隔離および複製アクション

quarantine('quarantine_name') アクションは、隔離エリアと呼ばれるキューに入れるメッセージにフラグを設定します。隔離についての詳細については、「隔離」の章を参照してください。duplicate-quarantine ('quarantine_name') アクションを実行すると、メッセージのコピーが指定されている隔離エリアにただちに配置されます。隔離エリア名の大文字と小文字は区別されます。

隔離フラグの付けられたメッセージは、電子メール パイプラインの残りの処理を継続します。メッセージがパイプラインの末尾に到達すると、メッセージに 1 つ以上の隔離に関するフラグが設定されていれば、該当するキューに入ります。それ以外の場合は配信されます。メッセージがパイプラインの末尾に到達しなければ、隔離エリアには配置されません。

したがって、メッセージ フィルタに quarantine() アクションがあり、その後に bounce() または drop() アクションが続く場合、最後のアクションによりメッセージはパイプラインの末尾に到達しないため、メッセージは隔離エリアに配置されません。メッセージ フィルタに隔離アクションが含まれる場合も同様ですが、メッセージはアンチスパムまたはアンチウイルス スキャン、またはコンテンツ フィルタによりドロップされます。skip-filters() アクションによりメッセージは残りのメッセージ フィルタをとばしますが、コンテンツ フィルタが適用される場合があります。たとえば、メッセージ フィルタがメッセージに隔離フラグを設定し、同時に最後の skip-filters() アクションも設定している場合、電子メール パイプラインの他のアクションによりメッセージがドロップされる場合を除き、メッセージは残りのメッセージ フィルタをすべてスキップした上で隔離されます。

次の例では、メッセージに「secret_word」という辞書にあるいずれかの単語が含まれていると、そのメッセージは Policy 隔離エリアに送信されます。

```
quarantine_codenames:

if (dictionary-match ('secret_words'))

{

quarantine('Policy');

}
```

次の例では、ある会社に.mp3ファイル形式の添付ファイルをすべてドロップする公式ポリシーがあるものと仮定しています。受信メッセージに .mp3 形式の添付ファイルがある場合、この添付ファイルは削除され、残りのメッセージ（本文と他の添付ファイル）は本来の受信者に送信されます。元のメッセージにすべての添付ファイルが添付されているコピーが隔離（Policy 隔離エリアに送信）されます。ブロックされた添付ファイルを受信する必要がある場合、本来の受信者はメッセージを隔離エリアからリリースするよう要求することができます。

```
strip_all_mp3s:

if (attachment-filename == '(?i)\\.mp3$') {

duplicate-quarantine('Policy');

drop-attachments-by-name('(?i)\\.mp3$');

}
```

受信者変更アクション

alt-rcpt-to アクションは、メッセージの配信時にメッセージのすべての受信者を指定した受信者に変更します。

次のフィルタは、エンベロープ受信者のアドレスに .freelist.com が含まれているすべてのメッセージを送信し、そのメッセージのすべての受信者を system-lists@myhost.com に変更します。

```
freelistFilter:

if(rcpt-to == '\\.freelist\\.com$')

{

alt-rcpt-to('system-lists@myhost.com');

}
```

配信ホスト変更アクション

alt-mailhost アクションは、選択したメッセージのすべての受信者の IP アドレスを、指定した数値 IP アドレスまたはホスト名に変更します。

（注） alt-mailhost アクションを実行すると、アンチスパム スキャンによりスパムと分類されたメッセージが隔離されないようにすることができます。alt-mailhost アクションは quarantine アクションに優先して実行され、指定したメール ホストにメッセージを送信します。

次のフィルタは、すべての受信者について、受信者のアドレスをホスト example.com に変更します。

```
localRedirectFilter:

if(true)

{

alt-mailhost('example.com');

}
```

これにより、joe@anywhere.com に送信されるメッセージの Envelope To アドレスが joe@anywhere.com になり、メッセージは example.com のメールホストに送信されます。smtproutes コマンドで指定された追加ルーティング情報は、引き続きメッセージのルーティングに適用されます。（ローカル ドメインの電子メールのルーティング を参照）。

（注） alt-mailhost アクションではポート番号を指定できません。この操作を行うには、かわりに SMTP ルートを追加します。

次のフィルタは、すべてのメッセージを 192.168.12.5 にリダイレクトします。

```
local2Filter:

if(true)

{

alt-mailhost('192.168.12.5');

}
```

送信元ホスト（Virtual Gateway アドレス）変更アクション

alt-src-host アクションは、メッセージの送信元ホストを指定した送信元に変更します。送信元ホストは、メッセージの送信元となる IP インターフェイス、または IP インターフェイスのグループにより構成されます。IP インターフェイスのグループが選択された場合、システムは電子メールの配信時に、グループ内のすべての IP インターフェイスを送信元インターフェイスとして使用する処理を繰り返します。つまり、これにより 1 台の Cisco E メール セキュリティ アプライアンスに複数の仮想ゲートウェイ アドレスを設定できます。詳細については、

Virtual Gateway™ テクノロジーを使用してすべてのホストされたドメインでの構成のメールゲートウェイを参照してください。

IP インターフェイスは、現在システムで設定されている IP インターフェイスまたは IP インターフェイス グループだけに変更できます。次のフィルタは、IP アドレスが `1.2.3.4` であるリモート ホストから受信したすべてのメッセージに対して、発信（配信）IP インターフェイス `outbound2` を使用する仮想ゲートウェイを作成します。

```
externalFilter:
if(remote-ip == '1.2.3.4')
{
alt-src-host('outbound2');
}
```

次のフィルタは、IP アドレスが `1.2.3.4` であるリモート ホストから受信したすべてのメッセージに対して、IP インターフェイスのグループ Group1 を使用します。

```
groupFilter:
if(remote-ip == '1.2.3.4')
{
alt-src-host('Group1');
}
```

アーカイブ アクション

`archive` アクションは、元のメッセージ（すべてのメッセージ ヘッダーと受信者を含む）のコピーを、アプライアンス上の mbox 形式のファイルに保存します。このアクションでは、メッセージを保存するログ ファイルの名前がパラメータとして使用されます。システムはフィルタの作成時に、指定したファイル名で自動的にログ サブスクリプションを作成します。また、既存のフィルタ ログ ファイルを指定することもできます。フィルタとフィルタ ログ ファイルの作成後は、`filters -> logconfig` サブコマンドでフィルタ ログ オプションを編集できます。

（注） `logconfig` コマンドは `filters` のサブコマンドです。このサブコマンドの完全な説明については、CLI を使用したメッセージ フィルタの管理 （117 ページ）を参照してください。

mbox 形式は標準の UNIX メールボックス形式で、メッセージを簡単に表示するためのユーティリティが多数用意されています。ほとんどの UNIX システムでは、「`mail -f` *mbox.filename*」と入力して、ファイルを表示できます。mbox 形式はプレーン テキストであるため、普通のテキスト エディタを使用してメッセージの内容を表示することができます。

次の例では、エンベロープ送信者が joesmith@yourdomain.com と一致する場合に、メッセージのコピーが joesmith というログに保存されます。

```
logJoeSmithFilter:

if(mail-from == '^joesmith@yourdomain\\.com$')

{

archive('joesmith');

}
```

ヘッダー削除アクション

strip-header アクションは、メッセージの特定のヘッダーを調べ、配信する前に該当する行をメッセージから削除します。ヘッダーが複数ある場合は、ヘッダーのすべてのインスタンス（「Received:」ヘッダーなど）が削除されます。

次の例では、すべてのメッセージで送信前に X-DeleteMe ヘッダーが削除されます。

```
stripXDeleteMeFilter:

if (true)

{

strip-header('X-DeleteMe');

}
```

ヘッダーに関する操作を行う場合、ヘッダーの現在の値には処理中に行われた変更（メッセージのヘッダーの追加、削除、変更を行うフィルタ処理など）が含まれている点に注意してください。詳細については、メッセージ ヘッダー ルールおよび評価 （6 ページ）を参照してください。

ヘッダー挿入アクション

insert-header アクションは、メッセージに新しいヘッダーを挿入します。AsyncOS は、挿入したヘッダーが規格を満たしているかどうかを検証しません。生成されるメッセージが電子メールのインターネット規格を満たしているかどうかは、ユーザが自分で確認する必要があります。

次の例では、X-Company というヘッダーがメッセージにない場合に、このヘッダーに My Company Name という値が設定されます。

```
addXCompanyFilter:

if (not header('X-Company'))

{

insert-header('X-Company', 'My Company Name');
```

```
}
```

insert-header() アクションでは、ヘッダーのテキストに非 ASCII 文字を使用できます。ただし、ヘッダー名には（規格遵守のため）ASCII 文字しか使用できません。可読性を最大限に高めるため、トランスポート エンコードは Quoted-Printable となります。

（注） strip-headers アクションと insert-header アクションを組み合わせることにより、元のメッセージにある任意のメッセージヘッダーを書き換えることができます。場合によっては、同じヘッダーを複数回使用することができますが（Received: など）、それ以外の場合は同じヘッダーを複数回使用すると MUA が混乱する場合があります（Subject: ヘッダーを複数回使用する場合など）。

ヘッダーに関する操作を行う場合、ヘッダーの現在の値には処理中に行われた変更（メッセージのヘッダーの追加、削除、変更を行うフィルタ処理など）が含まれている点に注意してください。詳細については、メッセージ ヘッダー ルールおよび評価 （6 ページ）を参照してください。

ヘッダー テキスト編集アクション

edit-header-text アクションを実行すると、正規表現の置換機能を使用して、指定したヘッダーテキストを書き換えることができます。このフィルタはヘッダー内で正規表現と一致するテキストを検索し、指定した正規表現に置き換えます。

たとえば、電子メールに次のような件名ヘッダーがあるものとします。

```
Subject: SCAN Marketing Messages
```

次のフィルタは、「SCAN」というテキストを削除し、「Marketing Messages」というテキストをヘッダー内に残します。

```
Remove_SCAN: if true

{

edit-header-text ('Subject', '^SCAN\\s*','');

}
```

フィルタはメッセージを処理した後、次のヘッダーを返します。

```
Subject: Marketing Messages
```

本文編集アクション

edit-body-text() メッセージ フィルタの機能は Edit-Header-Text() フィルタと同様ですが、メッセージのヘッダーではなく本文が処理対象です。

edit-body-text() メッセージフィルタは次の構文を使用します。最初のパラメータは検索のための正規表現で、2 番めのパラメータは置換のためのテキストです。

```
Example: if true {

edit-body-text("parameter 1","parameter 2");

}
```

`edit-body-text()` メッセージ フィルタはメッセージ本文のみが処理対象です。特定の MIME 部分がメッセージの「本文」と見なされるか「添付ファイル」と見なされるかの詳細については、メッセージ本文とメッセージ添付ファイル （6 ページ）を参照してください。

次の例では、メッセージから URL が削除され、「URL REMOVED」というテキストに置き換えられています。

```
URL_Replaced: if true {

edit-body-text("(?i)(?:https?|ftp)://[^\\s\">]+", "URL REMOVED");

}
```

次の例では、メッセージの本文から社会保障番号が削除され、「XXX-XX-XXXX」というテキストに置き換えられています。

```
ssn: if true {

edit-body-text("(?!000)(?:[0-6]\\d{2}|7(?:[0-6]\\d|7[012]))([
-]?)(?!00)\\d\\d\\1(?!0000)\\d{4}",

"XXX-XX-XXXX");

}
```

（注） 現時点では、`edit-body-text()` フィルタではスマート ID を使用できません。

HTML 変換アクション

RFC 2822 では電子メール メッセージのテキスト形式が規定されていますが、RFC 2822 メッセージ内の他のコンテンツのトランスポートを実現するための拡張機能（MIME など）があります。AsyncOS は `html-convert()` メッセージ フィルタを使用して、次の構文により HTML をプレーン テキストに変換できます。

```
Convert_HTML_Filter:

if (true)

{

html-convert();

}
```

Cisco メッセージ フィルタは、特定の MIME 部分がメッセージの「本文」であるか「添付ファイル」であるかを判別します。`html-convert()` メッセージ フィルタはメッセージ本文のみが処理対象です。メッセージの本文と添付ファイルの詳細については、メッセージ本文とメッセージ添付ファイル （6 ページ）を参照してください。

`html-convert()` フィルタが文書内の HTML を削除する方式は、形式によって異なります。

メッセージがプレーン テキスト（text/plain）である場合、メッセージは変更されずにフィルタを通過します。メッセージが単純な HTML メッセージ（text/html）である場合、すべての HTML タグはメッセージから削除され、残りの本文が HTML メッセージにかわり使用されます。各行の再フォーマットは行われず、HTML がプレーン テキストになることはありません。構造が MIME（multipart/alternative 構造）で、同じコンテンツに text/plain 部分と text/html 部分が含まれている場合、フィルタはメッセージの text/html 部分を削除して text/plain 部分を残します。その他の MIME タイプ（multipart/mixed など）では、すべての HTML 本文部分のタグが削除され、メッセージに再挿入されます。

メッセージ フィルタでは、`html-convert()` フィルタ アクションは処理対象のメッセージにタグを設定するだけで、メッセージ構造の変更はすぐには行われません。メッセージの変更は、すべての処理が完了した後に行われます。これにより、変更前に他のフィルタ アクションが元のメッセージを処理することができます。

バウンス プロファイル アクション

`bounce-profile` アクションは、設定済みのバウンス プロファイルをメッセージに割り当てます。（バウンスした電子メールの処理 を参照）。メッセージを配信できない場合、バウンス プロファイルで設定されたバウンス オプションが使用されます。この機能は、リスナーの設定から割り当てられているバウンスプロファイル（割り当てられている場合）に優先して適用されます。

次のフィルタの例では、送信される電子メールのうち、ヘッダーに「`X-Bounce-Profile: fastbounce`」があるすべての電子メールにバウンス プロファイル「fastbounce」が割り当てられます。

```
fastbounce:

if (header ('X-Bounce-Profile') == 'fastbounce') {

bounce-profile ('fastbounce');

}
```

アンチスパム システムのバイパス アクション

`skip-spamcheck` アクションは、システムに設定されたコンテンツベースのアンチスパム フィルタリングをすべてバイパスするようシステムに指示します。コンテンツベースのアンチスパム フィルタリングが設定されていない場合、またはメッセージがあらかじめスパム スキャンの対象に設定されていない場合は、このアクションを実行してもメッセージに影響はありません。

次の例では、メッセージの SenderBase レピュテーション スコアが高い場合に、メッセージに対するコンテンツベースのアンチスパム フィルタリングがバイパスされます。

```
whitelist_on_reputation:

if (reputation > 7.5)

{

skip-spamcheck();

}
```

関連項目

- 着信リレーが機能にどのように影響するか
- スパム フィルタからのアプライアンス生成メッセージの保護

グレイメール アクションのバイパス

特定のメッセージにグレイメール アクションを適用しない場合、次のメッセージ フィルタ アクションを使用してバイパスできます。

メッセージ フィルタ アクション	説明
skip-marketingcheck	マーケティング メールに対するアクションのバイパス
skip-socialcheck	ソーシャル ネットワーク メールに対するアクションのバイパス
skip-bulkcheck	バルク メールに対するアクションのバイパス

次の例では、リスナー “private_listener” で受信したメッセージは、ソーシャル ネットワーク メールに対するグレイメール アクションをバイパスする必要があること指定しています。

```
internal_mail_is_safe:

if (recv-listener == 'private_listener')

{

skip-socialcheck();

}
```

アンチウイルス システムのバイパス アクション

skip-viruscheck アクションは、システムに設定されたウイルス保護システムをすべてバイパスするようシステムに指示します。アンチウイルス システムが設定されていない場合、またはメッセージがあらかじめウイルス スキャンの対象に設定されていない場合は、このアクションを実行してもメッセージに影響はありません。

次の例では、「private_listener」というリスナーで受信したメッセージに対して、アンチスパム システムとアンチウイルス システムによる処理がバイパスされています。

```
internal_mail_is_safe:

if (recv-listener == 'private_listener')

{

skip-spamcheck();

skip-viruscheck();

}
```

ファイル レピュテーション フィルタリングおよびファイル分析システムのバイパス アクション

skip-ampcheck アクションは、メッセージがシステムで設定されたファイル レピュテーション フィルタリングおよびファイル分析をバイパスすることを許可するよう、システムに指示します。ファイル レピュテーション フィルタリングおよびファイル分析が設定されていない場合、またはメッセージがあらかじめファイル レピュテーション フィルタリングおよびファイル分析スキャンの対象に設定されていない場合は、このアクションを実行してもメッセージに影響はありません。

次の例では、PDF 添付ファイルを含むメッセージがファイル レピュテーション フィルタリングおよびファイル分析をバイパスすることを指定します。

```
skip_amp_scan:
if (attachment-filetype == 'pdf')
{
skip-ampcheck();
}
```

ウイルス アウトブレイク フィルタのスキャニング処理バイパス アクション

skip-vofcheck アクションは、メッセージのウイルス アウトブレイク フィルタによるスキャニング処理がバイパスされるようシステムに指示します。ウイルス アウトブレイク フィルタのスキャニング処理がイネーブルになっていない場合、このアクションを実行してもメッセージに影響はありません。

次の例では、「private_listener」というリスナーで受信したメッセージに対して、ウイルス アウトブレイク フィルタのスキャニング処理がバイパスされています。

```
internal_mail_is_safe:

if (recv-listener == 'private_listener') Outbreak Filters

{

skip-vofcheck();

}
```

メッセージ タグ追加アクション

tag-message アクションは、DLP ポリシー フィルタリングで使用するカスタム用語を送信メッセージに挿入します。DLP ポリシーを設定して、スキャン対象をメッセージ タグがあるメッセージに限定することができます。メッセージタグは受信者側では表示されません。タグ名には、[a-zA-Z0-9_-.] の範囲の文字のうち任意のものを組み合わせて使用できます。.

メッセージのフィルタリングに使用する DLP ポリシーの設定の詳細については、「データ消失防止」の章を参照してください。

次の例では、件名に「[Encrypt]」が含まれるメッセージにメッセージタグを挿入しています。Cisco Email Encryption が使用できる場合は、メッセージの配信前にメッセージをこのメッセージ タグで暗号化する DLP ポリシーを作成できます。

```
Tag_Message:

if (subject == '^\\[Encrypt\\]')

{

tag-message('Encrypt-And-Deliver');

}
```

ログ エントリ追加アクション

log-entry アクションは、カスタマイズしたテキストを、テキスト メール ログに INFO レベルで追加します。このテキストにはアクション変数を使用することができます。このアクションを使用すると、デバッグ時に便利なテキストや、メッセージフィルタがアクションを実行した理由に関する情報を挿入できます。ログ エントリはメッセージ トラッキングにも表示されます。

次の例では、メッセージに会社の機密情報が含まれていると判断されたためメッセージがバウンスされたことを示すログ エントリが挿入されます。

```
CompanyConfidential:

if (body-contains('Company Confidential'))

{

log-entry('Message may have contained confidential information.');

bounce();

}
```

URL レピュテーション アクション

メッセージに含まれる URL のレピュテーション スコアを使用して、URL またはその動作を変更します。重要な詳細と例については、悪意のある URL または望ましくない URL からの保護

のメッセージに含まれる URL の変更：フィルタでの URL レピュテーションまたは URL カテゴリのアクションの使用 を参照してください。

これらのアクションでは、ルールは不要です。

URL レピュテーション アクションの各部分は次のとおりです。

- msg_filter_name はこのメッセージ フィルタの名前です。
- min_score および max_score は、アクション適用範囲の最小スコアと最大スコアです。適用範囲には、指定する値も含まれます。

最小スコアと最大スコアは -10.0 から 10.0 までの範囲内の数値でなければなりません。

- レピュテーション サービスからスコアが提供されない場合のアクションを指定するには、このアクションの「no-reputation」バージョンを使用します。これについては以降の項で説明します。
- whitelist は、（urllistconfig コマンドを使用して）定義されている URL リストの名前です。ホワイトリストの指定は任意です。
- Preserve_signed の位置に 0 または 1 を入力します。
  - 1 - このアクションを未署名のメッセージだけに適用する
  - 0 - このアクションをすべてのメッセージに適用する

preserve_signed 値を指定しないと、アクションは未署名のメッセージだけに適用されます。

関連項目

- URL レピュテーションに基づき URL をテキストに置換する （102 ページ）
- URL レピュテーションに基づき URL の危険を取り除く （103 ページ）
- URL レピュテーションに基づき Cisco セキュリティ プロキシに URL をリダイレクトする （103 ページ）

URL レピュテーションに基づき URL をテキストに置換する

レピュテーション サービスからスコアが提供される場合にアクションを実行するには

url-reputation-replace アクションを使用します。

url-reputation-replace アクションを使用するフィルタの構文を次に示します。

```
<msg_filter_name>:
if <condition>
{url-reputation-replace(<min_score>, <max_score>,'<replace_text>', '< whitelist> ', < Preserve_signed> );}
```

replace_text は、URL を置き換えるテキストです。

レピュテーション サービスからスコアが提供されない場合にアクションを実行するには

url-no-reputation-replace アクションを使用します。

url-no-reputation-replace アクションを使用するフィルタの構文を次に示します。

```
<msg_filter_name>:
if <condition>
{url-no-reputation-replace ('<replace_text>', '<whitelist>', <Preserve_signed>);}
```

replace_text は、URL を置き換えるテキストです。

URL レピュテーションに基づき URL の危険を取り除く

レピュテーション サービスからスコアが提供される場合にアクションを実行するには

url-reputation-defang アクションを使用します。

url-reputation-defang アクションを使用するフィルタの構文を次に示します。

```
<msg_filter_name>:
if <condition>
{url-reputation-defang (<min_score>, <max_score>, '<whitelist>', <Preserve_signed>);}
```

レピュテーション サービスからスコアが提供されない場合にアクションを実行するには

url-no-reputation-defang アクションを使用します。

url-no-reputation-defang アクションを使用するフィルタの構文を次に示します。

```
<msg_filter_name>:
if <condition>
{url-no-reputation-defang ('<whitelist>', <Preserve_signed>);}
```

URL レピュテーションに基づき Cisco セキュリティ プロキシに URL をリダイレクトする

レピュテーション サービスからスコアが提供される場合にアクションを実行するには

url-reputation-proxy-redirect アクションを使用します。

url-reputation-proxy-redirect アクションを使用するフィルタの構文を次に示します。

```
<msg_filter_name>:
if <condition>
{url-reputation-proxy-redirect (<min_score>, <max_score>, '<whitelist>',
<Preserve_signed>);}
```

レピュテーション サービスからスコアが提供されない場合にアクションを実行するには

url-no-reputation-proxy-redirect アクションを使用します。

url-no-reputation-proxy-redirect アクションを使用するフィルタの構文を次に示します。

```
<msg_filter_name>:
if <condition>
```

```
{url-no-reputation-proxy-redirect ('<whitelist>', <Preserve_signed>);}
```

URL カテゴリ アクション

メッセージに含まれる URL のカテゴリを使用して、URL またはその動作を変更します。重要な詳細については、悪意のある URL または望ましくない URL からの保護のメッセージに含まれる URL の変更：フィルタでの URL レピュテーションまたは URL カテゴリのアクションの使用 を参照してください。

これらのアクションでは、ルールは不要です。

すべての URL カテゴリ アクションの各部分は次のとおりです。

- msg_filter_name はメッセージ フィルタの名前です。
- category-name は URL カテゴリです。複数のカテゴリを指定する場合は、各カテゴリをカンマで区切ります。正しいカテゴリ名を確認するには、コンテンツ フィルタの URL カテゴリ条件またはアクションを確認してください。カテゴリの説明と例については、URL カテゴリについてを参照してください。
- url_white_list は、（urllistconfig コマンドを使用して）定義されている URL リストの名前です。
- unsigned-only：0 または 1 を入力します。
  - 1 - このアクションを未署名のメッセージだけに適用する
  - 0 - このアクションをすべてのメッセージに適用する

関連項目

- URL カテゴリに基づき URL をテキストに置換する （104 ページ）
- URL カテゴリに基づき URL の危険を取り除く （104 ページ）
- URL カテゴリに基づき Cisco セキュリティ プロキシに URL をリダイレクトする （105 ページ）

URL カテゴリに基づき URL をテキストに置換する

`url-category-replace` アクションを使用するフィルタの構文を次に示します。

```
<msg_filter_name>:
if <condition>
url-category-replace(['<category-name1>','<category-name2>',...,
'<category-name3>'],'<replacement-text>', '<url_white_list>', <unsigned-only>);
```

`replacement-text` は、URL を置き換えるテキストです。

URL カテゴリに基づき URL の危険を取り除く

`url-category-defang` アクションを使用するフィルタの構文を次に示します。

```
<msg_filter_name>:
if <condition>
```

```
url-category-defang(['<category-name1>','<category-name2>',..., '<category-name3>'],
'<url_white_list>', <unsigned-only>);
```

URL カテゴリに基づき Cisco セキュリティ プロキシに URL をリダイレクトする

`url-category-proxy-redirect` アクションを使用するフィルタの構文を次に示します。

```
<msg_filter_name>:

if <condition>

url-category-proxy-redirect(['<category-name1>','<category-name2>',...,
'<category-name3>'], '<url_white_list>', <unsigned-only>);
```

オペレーションなし

オペレーションなしアクションは、操作を実行しません（no-op）。通知、隔離、ドロップなどその他のアクションを使用しない場合にメッセージフィルタでこのアクションを使用できます。たとえば、作成した新しいメッセージ フィルタの動作を確認する場合に、操作なしアクションを使用できます。メッセージ フィルタが動作したら、[メッセージフィルタ（Message Filters）] レポート ページを使用して新しいメッセージ フィルタの動作をモニタし、要件に対応するようにフィルタを調整できます。

次に、操作なしアクションをメッセージ フィルタで使用する例を示します。

```
new_filter_test: if header-repeats ('subject', X, 'incoming') {no-op();}
```

偽造メールの検出アクション

偽装されたメッセージから From: ヘッダーを削除し、エンベロープ送信者で置き換えます。

次のメッセージフィルタは、メッセージ内の From: ヘッダーと辞書の用語を比較し、コンテンツ辞書の用語のマッチング スコアが 70 以上である場合、メッセージ フィルタは From: ヘッダーを除去し、エンベロープ送信者と置き換えます。

```
FED_CF: if (forged-email-detection("Execs", 70)) { fed("from", ""); }
```

添付ファイルのスキャン

E メール セキュリティ アプライアンスではコンテンツ スキャナを使用して、会社のポリシーと整合しないメッセージから添付ファイルを削除できます。元のメッセージはそのまま配信できます。

添付ファイルのフィルタリングは、特定のファイルタイプ、フィンガープリント、添付ファイルの内容に基づいて行うことができます。フィンガープリントを使用して添付ファイルの正確な種類を判別することにより、ユーザは悪意のある添付ファイルの拡張子（.exe など）を一般的な拡張子（.doc など）に変更して、名前が変更されたファイルが添付ファイル フィルタを通過できるようにすることができなくなります。

添付ファイルのコンテンツをスキャンする際、コンテンツ スキャナは添付ファイルからデータを抽出し、正規表現による検索を実行します。添付ファイルのデータとメタデータの両方が検

査対象となります。Excel または Word 文書をスキャンする場合、添付ファイル スキャン エンジンは .exe、.dll、.bmp、.tiff、.pcx、.gif、.jpeg、.png、Photoshop 画像の各埋め込みファイルも検出できます。

アプライアンスのコンテンツ スキャナでは、次のアーカイブ ファイル形式でコンテンツ スキャンを実行できます。

- ACE アーカイブ
- ALZ アーカイブ
- Apple ディスク イメージ
- ARJ アーカイブ
- bzip2 アーカイブ
- EGG アーカイブ
- GNU Zip
- ISO ディスク イメージ
- Java アーカイブ
- LZH
- Microsoft キャビネット アーカイブ
- RAR マルチパート ファイル
- RedHat パッケージ マネージャ アーカイブ
- Roshal アーカイブ（RAR）
- UNIX AR アーカイブ
- UNIX 圧縮アーカイブ
- UNIX cpio
- UNIX Tar
- XZ アーカイブ
- ZIP アーカイブ
- 7-Zip

（注） コンテンツ スキャナ関連ファイルの詳細を表示するには、Web インターフェイスで [セキュリティ サービス（Security Services）] > [スキャン動作（Scan Behavior）] ページを使用するか、CLI で contentscannerstatus コマンドを使用します。これらのファイルは、アップデート サーバを使用して自動的に更新されます。これらのファイルを手動で更新する場合は、スキャン動作の設定 （141 ページ）を参照してください。

関連項目

- 添付ファイルのスキャンで使用するメッセージ フィルタ （107 ページ）
- イメージ分析 （108 ページ）
- イメージ分析スキャン エンジンの設定 （109 ページ）
- イメージ分析結果に基づいたアクション実行のメッセージ フィルタの構成（111 ページ）
- 通知 （113 ページ）
- 添付ファイルのスキャン メッセージ フィルタの例 （113 ページ）

添付ファイルのスキャンで使用するメッセージ フィルタ

次の表に記載されているメッセージフィルタ アクションは、最終でないアクションです。（添付ファイルはドロップされ、メッセージの処理が続行されます）。

オプションのコメントは、フッターのようにメッセージに追加されるテキストで、メッセージ フィルタ アクション変数（添付ファイルのスキャン メッセージ フィルタの例 （113 ページ）を参照）を使用することもできます。

表 *8*: 添付ファイルのスキャンで使用するメッセージ フィルタ アクション

操作	構文	説明
添付ファイルのドロップ（名前別）	`drop-attachments-by-name (<regular expression >[, <optional comment >])`	メッセージの添付ファイルのうち、指定した正規表現と一致する名前のファイルをすべてドロップします。アーカイブ形式の添付ファイル（zip、tar）内に該当するファイルがある場合、この添付ファイルはドロップされます。添付ファイルのスキャン メッセージ フィルタの例 （113 ページ）を参照してください。
添付ファイルのドロップ（タイプ別）	`drop-attachments-by-type (<MIME type >[, <optional comment >])`	メッセージの添付ファイルのうち、指定したMIMEタイプまたはファイル拡張子に該当するMIMEタイプのファイルをすべてドロップします。アーカイブ形式の添付ファイル（zip、tar）内に該当するファイルがある場合、この添付ファイルはドロップされます。
添付ファイルのドロップ（ファイル タイプ別）	`drop-attachments-by-filetype (<fingerprint name >[, <optional comment >])`	メッセージの添付ファイルのうち、指定したファイルの「フィンガープリント」と一致するファイルをすべてドロップします。アーカイブ形式の添付ファイル（zip、tar）内に該当するファイルがある場合、この添付ファイルはドロップされます。

操作	構文	説明
添付ファイルのドロップ（MIME タイプ別）	drop-attachments-by-mimetype (<*MIME type*>[, <*optional comment*>])	メッセージの添付ファイルのうち、特定の MIME タイプのファイルをすべてドロップします。このアクションではファイル拡張子による MIME タイプの判別は行われず、アーカイブの内容の確認もされません。
添付ファイルのドロップ（サイズ別）	drop-attachments-by-size (<*number*>[, <*optional comment*>])	メッセージの添付ファイルのうち、ローエンコード形式で指定したサイズ（バイト単位）以上のサイズであるファイルをすべてドロップします。アーカイブ ファイルまたは圧縮ファイルの場合、このアクションは、圧縮前のサイズを検証せず、実際の自体のサイズが計測されます。
添付ファイルのスキャン	drop-attachments-where-contains (<*regular expression*>[, <*optional comment*>])	メッセージの添付ファイルのうち、指定した正規表現を含むファイルをすべてドロップします。アーカイブ ファイル（zip、tar）は、中に含まれているファイルのいずれかが正規表現と一致する場合にドロップされます。
添付ファイルのドロップ（辞書との一致別）	drop-attachments-where-dictionary-match(<dictionary name>)	このフィルタ アクションは、辞書の用語との一致に基づいて添付ファイルを削除します。添付ファイルであると判断される MIME 部分の用語が辞書の用語と一致する場合（かつ、ユーザ定義のしきい値に達している場合）、添付ファイルが電子メールから削除されます。添付ファイルのスキャン メッセージ フィルタの例（113 ページ）を参照してください。

イメージ分析

メッセージによってはイメージを含むものがあり、適切でないコンテンツがないかスキャンすることが必要になる場合があります。イメージ分析エンジンを使用すると、電子メール内の適切でないコンテンツを検索できます。イメージ分析は、アンチウイルスおよびアンチスパム スキャン エンジンの補完または代替を目的とするものではありません。この機能は、電子メール内の適切でないコンテンツを特定することにより、許容範囲での使用を促進するためのものです。イメージ分析スキャン エンジンを使用すると、メールの隔離と分析、および傾向の認識ができます。

アプライアンスでイメージ分析を設定すると、イメージ分析フィルタ ルールを使用して、疑わしい電子メールまたは不適切な電子メールに対してアクションを実行できます。イメージ スキャンでは、次のタイプの添付ファイルをスキャンできます：BMP、JPG、TIF、PNG、GIF、TGA、PCX。イメージ アナライザは、スキン カラー、本体サイズ、曲率を測定するアルゴリズムを使用し、画像に適切でないコンテンツが含まれる可能性を判定します。イメージ添付ファイルをスキャンすると、Cisco フィンガープリントによりファイル タイプが特定され、イメージ アナライザはイメージ コンテンツを分析するアルゴリズムを使用します。イメージが他のファイルに埋め込まれている場合、コンテンツ スキャナはファイルを抽出します。イメージ分析の結果は、メッセージ全体で計算されます。メッセージにイメージがない場合、メッセージのスコアは 0 となります。これは分析結果が「Clean」であることを表します。そのため、イメージがないメッセージに対する分析結果は「Clean」となります。

イメージ分析スキャン エンジンの設定

GUI からイメージ分析をイネーブル化するには、次の手順を実行します。

ステップ 1 [セキュリティサービス（Security Services）] > [IronPortイメージ分析（IronPort Image Analysis）] の順に進みます。

ステップ 2 [有効（Enable）] をクリックします。

成功したことを示すメッセージが表示され、分析結果設定が表示されます。

イメージ分析フィルタ ルールを使用すると、次の各分析結果に基づいてアクションを決定できます。

- [正常（Clean）]：イメージに適切でないコンテンツはありません。イメージ分析の結果はメッセージ全体で計算されるため、イメージがないメッセージをスキャンすると分析結果は [正常（Clean）] となります。
- [疑わしい（Suspect）]：イメージに適切でないコンテンツがある可能性があります。
- [不適切（Inappropriate）]：イメージに適切でないコンテンツがあります。

これらの計算結果には、イメージ アナライザのアルゴリズムにより、適切でないコンテンツがある可能性を示す数値が割り当てられます。

次の値が推奨されます。

- [正常（Clean）]：0 ～ 49
- [疑わしい（Suspect）]：50 ～ 74
- [不適切（Inappropriate）]：75 ～ 100

次のタスク

精度を設定することによりイメージスキャンを微調整できます。これにより、誤判定を減らすことができます。たとえば、誤判定が発生している場合は、精度を低くします。逆に、イメージスキャンで適切でないコンテンツが検出されていない場合は、精度を高く設定します。精度

設定は 0（一切検出しない）と 100（精度が最高である）の間の値です。デフォルトの精度の 65 に設定することを推奨します。

関連項目

- イメージ分析設定の調整 （110 ページ）

イメージ分析設定の調整

ステップ 1 [セキュリティサービス（Security Services）] > [IronPortイメージ分析（IronPort Image Analysis）] の順に進みます。

ステップ 2 [設定の編集（Edit Settings）] をクリックします。

ステップ 3 イメージ分析の精度を設定します。デフォルトの精度の 65 に設定することを推奨します。

ステップ 4 [正常（Clean）]、[疑わしい（Suspect）]、および [不適切（Inappropriate）] の評価を設定します。

値の範囲を設定する場合、値が重ならないようにしてください。また、すべて整数を使用してください。

ステップ 5 任意で、最小サイズの要件を満たさないイメージのスキャンをバイパスするように、AsyncOS を設定します（推奨）。デフォルトで、この設定は 100 ピクセルに設定されています。100 ピクセル未満のイメージをスキャンすると、誤検知が発生する可能性があります。

imageanalysisconfig コマンドを使用して CLI でイメージ分析設定を有効にすることもできます。

次のタスク

関連項目

- 特定のメッセージの判定スコアの表示 （110 ページ）

特定のメッセージの判定スコアの表示

特定のメッセージのレピュテーション スコアを確認するには、メール ログを参照します。メール ログにはイメージ名またはファイル名、特定のメッセージの添付ファイルのスコアが表示されます。また、ログにはファイル内のイメージがスキャン可能かどうかについての情報も表示されます。このログには、各イメージではなく、各メッセージの添付ファイルの結果に関する情報が表示されます。たとえば、メッセージに JPEG イメージを含む zip ファイルが添付されていた場合、ログのエントリには JPEG の名前ではなく、zip ファイルの名前が表示されます。また、zip ファイルに複数のイメージが含まれている場合、ログ エントリにはすべてのイメージの最大スコアが表示されます。「unscannable」の通知は、いずれかのイメージがスキャンできないことを意味します。

ログには、スコアがどのように特定の評価（[正常（clean）]、[疑わしい（suspect）]、または [不適切（inappropriate）]）に反映されるかに関する情報はありません。ただし、メール ログを使用して特定のメッセージの配信を追跡できるため、メッセージに対して実行されたアクションによって、メールに不適切なイメージまたは疑わしいイメージが含まれていたかどうかがわかります。

たとえば、次のメール ログでは、イメージ分析スキャンの結果、メッセージ フィルタ ルールによってドロップされた添付ファイルを示しています。

```
Thu Apr 3 08:17:56 2009 Debug: MID 154 IronPort Image Analysis: image 'Unscannable.jpg'
is unscannable.

Thu Apr 3 08:17:56 2009 Info: MID 154 IronPort Image Analysis: attachment
'Unscannable.jpg' score 0 unscannable

Thu Apr 3 08:17:56 2009 Info: MID 6 rewritten to MID 7 by
drop-attachments-where-image-verdict filter 'f-001'

Thu Apr 3 08:17:56 2009 Info: Message finished MID 6 done
```

イメージ分析結果に基づいたアクション実行のメッセージフィルタの構成

イメージ分析をイネーブルにしたら、メッセージフィルタを作成して、さまざまなメッセージの評価に対してさまざまなアクションを実行する必要があります。たとえば、問題ないと評価されたメッセージを配信し、不適切なコンテンツを含むと判断されたメッセージを隔離する必要があるとします。

（注） シスコでは、不適切または疑わしいと評価されたメッセージをドロップまたはバウンスしないことを推奨します。代わりに、後で確認してトレンド分析について把握するために、違反したメッセージのコピーを隔離します。

次のフィルタは、コンテンツが不適切または疑わしい場合にタグを付けられるメッセージを示しています。

```
image_analysis: if image-verdict == "inappropriate" {

strip-header("Subject");

insert-header("Subject", "[inappropriate image] $Subject");

}

else {

if image-verdict == "suspect" {

strip-header("Subject");


insert-header("Subject", "[suspect image] $Subject");

}

}
```

関連項目

- イメージ分析の評価に基づいて添付ファイルを除去するコンテンツ フィルタの作成 （112 ページ）

イメージ分析の評価に基づいて添付ファイルを除去するコンテンツ フィルタの作成

イメージ分析をイネーブルにすると、コンテンツ フィルタを作成してイメージ分析の評価に基づいて添付ファイルを削除するか、さまざまなメッセージの評価に対してさまざまなアクションを実行するようにフィルタを設定できます。たとえば、不適切なコンテンツを含むメッセージを隔離することに決定したとします。

イメージ分析の評価に基づいて添付ファイルを削除するには、次の手順を実行します。

ステップ 1 [メールポリシー（Mail Policies）] > [受信コンテンツフィルタ（Incoming Content Filters）] をクリックします。

ステップ 2 [フィルタを追加（Add Filter）] をクリックします。

ステップ 3 コンテンツ フィルタの名前を入力します。

ステップ 4 [アクション（Actions）] で、[アクションを追加（Add Action）] をクリックします。

ステップ 5 [ファイル情報によって添付ファイルを除去（Strip Attachment by File Info）] で、[イメージ分析判定（Image Analysis Verdict is）] をクリックします。

ステップ 6 次のイメージ分析の評価から選択します。

- 疑わしい（Suspect）
- 不適切（Inappropriate）
- 不適切もしくは疑わしい（Suspect or Inappropriate）
- スキャン不可（Unscannable）
- 正常（Clean）

イメージ分析判定に基づくアクションの設定

イメージ分析の評価に基づくアクションを設定するには、次の手順を実行します。

ステップ 1 [メールポリシー（Mail Policies）] > [受信コンテンツフィルタ（Incoming Content Filters）] をクリックします。

ステップ 2 [フィルタを追加（Add Filter）] をクリックします。

ステップ 3 コンテンツ フィルタの名前を入力します。

ステップ 4 [条件（Conditions）] で、[条件を追加（Add Condition）] をクリックします。

ステップ 5 [添付ファイルのファイル情報（Attachment File Info）] で、[イメージ分析判定（Image Analysis Verdict）] をクリックします。

ステップ 6 次のいずれかの評価を選択します。

- 疑わしい（Suspect）
- 不適切（Inappropriate）
- 不適切もしくは疑わしい（Suspect or Inappropriate）
- スキャン不可（Unscannable）
- 正常（Clean）

ステップ 7 [アクションを追加（Add Action）] をクリックします。

ステップ 8 イメージ分析の評価に基づいてメッセージに対して実行するアクションを選択します。

ステップ 9 変更を送信し、保存します。

通知

GUI の [テキストリソース（Text Resources）] ページまたは `textconfig` CLI コマンドを使用して、カスタム通知テンプレートをテキストリソースとして設定することもできます。これも、添付ファイルのフィルタ ルールと組み合わせて使用すると便利なツールです。通知テンプレートは非 ASCII 文字をサポートしています（テンプレートを作成するとき、エンコードを選択するように要求されます）。

次の例では、最初に `textconfig` コマンドを使用して、`strip.mp3` という名前の通知テンプレートを作成します。これは、通知メッセージの本文に挿入されます。次に、添付ファイルのフィルタ ルールを作成し、.mp3 ファイルがメッセージから削除された場合、予定していた受信者宛てに .mp3 ファイルが削除されたことを通知する電子メールが送信されるように設定できます。

```
drop-mp3s:

if (attachment-type == '*/mp3')

{ drop-attachments-by-filetype('Media');

notify ('$EnvelopeRecipients', 'Your mp3 has been removed', '$EnvelopeFrom',

'strip.mp3');

}
```

詳細については、通知およびコピー通知アクション （86 ページ）を参照してください。

添付ファイルのスキャン メッセージ フィルタの例

次に、添付ファイルに対して実行されるアクションの例を示します。

- ヘッダーの挿入 （114 ページ）
- ファイル タイプによる添付ファイルのドロップ （114 ページ）
- ディクショナリの一致による添付ファイルのドロップ （116 ページ）
- 保護された添付ファイルの隔離 （116 ページ）

• 保護されていない添付ファイルの検出 （116 ページ）

ヘッダーの挿入

この例では、添付ファイルに指定したコンテンツが含まれている場合に、AsyncOS がヘッダーを挿入します。

次の例では、あるキーワードが含まれるかどうか、メッセージのすべての添付ファイルをスキャンします。すべての添付ファイルにキーワードが存在する場合、カスタムの `X-Header` が挿入されます。

```
attach_disclaim:

if (every-attachment-contains('[dD]isclaimer') ) {

insert-header("X-Example-Approval", "AttachOK");

}
```

次の例では、特定のバイナリ データのパターンがあるかどうか、添付ファイルをスキャンします。フィルタは `attachment-binary-contains` フィルタ ルールを使用して、PDF ドキュメントが暗号化されていることを示すパターンを検索します。バイナリ データ内にそのパターンが存在する場合、カスタム ヘッダーが挿入されます。

```
match_PDF_Encrypt:

if (attachment-filetype == 'pdf' AND

attachment-binary-contains('/Encrypt')){

strip-header (‘Subject’);

insert-header (‘Subject’, ‘[Encrypted] $Subject’);

}
```

ファイル タイプによる添付ファイルのドロップ

次の例では、添付ファイルの「executable」グループ（`.exe`、`.dll`、および `.scr`）がメッセージから削除され、削除されたファイルの名前をリストするテキストがメッセージに追加されます（`$dropped_filename` アクション変数を使用して）。`drop-attachments-by-filetype` アクションは添付ファイルを確認し、3 文字のファイル拡張子だけではなく、ファイルのフィンガープリントに基づいて添付ファイルを削除します。1 つのファイル タイプ（「mpeg」）を指定したり、あるファイル タイプのすべてのメンバ（「Media」）を参照したりできます。

```
strip_all_exes: if (true) {

drop-attachments-by-filetype ('Executable', “Removed attachment:

$dropped_filename”);

}
```

次の例では、エンベロープ送信者がドメイン example.com 内に存在しないメッセージから、同じ「executable」グループの添付ファイル（.exe、.dll、および .scr）が、削除されます。

```
strip_inbound_exes: if (mail-from != "@example\\.com$") {
drop-attachments-by-filetype ('Executable');
}
```

次の例では、エンベロープ送信者がドメイン example.com 内に存在しないメッセージから、ファイル タイプの特定のメンバ（「wmf」）および同じ「executable」グループの添付ファイル（.exe、.dll、および .scr）が削除されます。

```
strip_inbound_exes_and_wmf: if (mail-from != "@example\\.com$") {
drop-attachments-by-filetype ('Executable');
drop-attachments-by-filetype ('x-wmf');
}
```

次の例では、添付ファイルの「executable」事前定義グループが、より多くの添付ファイルの名前を含むように拡張されています（このアクションでは、添付ファイルのファイルタイプは確認されません）。

```
strip_all_dangerous: if (true) {
drop-attachments-by-filetype ('Executable');
drop-attachments-by-name('(?i)\\.(cmd|pif|bat)$');
}
```

drop-attachments-by-name アクションでは、非 ASCII 文字をサポートしています。

（注） drop-attachments-by-name アクションは、MIME ヘッダーでキャプチャされたファイル名に対して正規表現照合を実行します。MIME ヘッダーからキャプチャされたファイル名は、最後にスペースが存在する場合があります。

次の例では、添付ファイルがメッセージに .exe 実行ファイルのファイル タイプでない場合はドロップされます。ただし、フィルタは、除外するファイル タイプを備えた少なくとも 1 つの添付ファイルがあるメッセージへのアクションを実行しません。たとえば、次のフィルタは **.exe** ファイル タイプではない添付ファイルを含むメッセージをドロップします。

```
exe_check: if (attachment-filetype != "exe") {
drop();
}
```

メッセージに複数の添付ファイルがある場合、E メール セキュリティ アプライアンスは他の添付ファイルが .exe ファイルでない場合でも、添付ファイルの少なくとも 1 つが .exe ファイルの場合はメッセージをドロップしません。

ディクショナリの一致による添付ファイルのドロップ

この drop-attachments-where-dictionary-match アクションでは、辞書の用語との一致に基づいて添付ファイルを削除します。添付ファイルであると判断される MIME 部分の用語が辞書の用語と一致する場合（かつ、ユーザ定義のしきい値に達している場合）、添付ファイルが電子メールから削除されます。次の例では、「secret_words」辞書内の単語が添付ファイル内で検出されると、添付ファイルが削除されます。一致のしきい値は 1 に設定されている点に注意してください。

```
Data_Loss_Prevention: if (true) {

drop-attachments-where-dictionary-match("secret_words", 1);

}
```

保護された添付ファイルの隔離

attachment-protected フィルタでは、メッセージ内の添付ファイルがパスワード保護されているかをテストします。受信メールに対してこのフィルタを使用して、添付ファイルがスキャン可能かどうかを確認できます。この定義に従い、1 つの暗号化されたメンバと複数の暗号化されていないメンバーを含む zip ファイルは、保護されていると見なされます。同様に、オープン パスワードが設定されていない PDF ファイルは、コピーや印刷がパスワード保護されていたとしても、保護されているとは見なされません。次の例では、保護された添付ファイルが隔離エリア「Policy」に送信されます。

```
quarantine_protected:

if attachment-protected

{

quarantine("Policy");

}
```

保護されていない添付ファイルの検出

attachment-unprotected フィルタは、メッセージ内の添付ファイルがパスワード保護されていないかをテストします。このメッセージ フィルタは、attachment-protected フィルタと補完関係にあります。このフィルタを送信メールに使用して、保護されていないメールを検出することができます。次の例では、AsyncOS が送信リスナーで保護されていない添付ファイルを検出し、メッセージを隔離しています。

```
quarantine_unprotected:
```

```
if attachment-unprotected

{

quarantine("Policy");

}
```

メッセージフィルタを使用した、メッセージの添付ファイルの悪意のあるファイルの検出

例として、以下のメッセージフィルタルール構文を使用して、ETFエンジンによってメッセージの添付ファイル内で悪意があるとして分類されるファイルを検出し、そのようなメッセージに対して適切な対応をします。

構文：

```
Strip_malicious_files: if (file-hash-etf-rule (['etf_source1'],
<'file_hash_exception_list'>))
{ file-hash-etf-strip-attachment-action (['etf_source1'], <'file_hash_exception_list>,
 "file stripped from message attachment"); }
```

それぞれの説明は次のとおりです。

- `'file-hash-etf-rule'` は、添付ファイル情報のメッセージ フィルタのルールです。
- `'etf_source1'` は、ファイルのハッシュに基づいてメッセージの悪意のあるファイルを検出するために使用される ETF ソースです。
- `'file_hash_exception_list'` は、ファイル ハッシュの例外リストの名前です。ファイル ハッシュの例外リストが存在しない場合は「""」と表示されます。
- `'file-hash-etf-strip-attachment-action'` は、悪意のあるファイルが含まれるメッセージに対して適用するアクションです。

以下の例では、メッセージにETFエンジンによって悪意があるとして検出された添付ファイルが含まれる場合、添付ファイルが除去されます。

```
Strip_Malicious_Attachment: if (true) {file-hash-etf-strip-attachment-action
(['threat_feed_source'], "", "Malicious message attachment has been stripped from
the message.");}
```

CLI を使用したメッセージ フィルタの管理

CLIを使用して、メッセージフィルタの追加、削除、アクティブ化/非アクティブ化、インポート/エクスポート、ログ オプションの設定が可能です。次の表で、コマンドとサブコマンドについてまとめて説明します。次の表で、コマンドとサブコマンドについてまとめて説明します。

表 *9:* メッセージ フィルタ サブコマンド

構文	説明
filters	メイン コマンド。このコマンドは対話形式で、詳細情報を入力するよう要求されます（たとえば、new、delete、import など）。
new	新しいフィルタを作成します。場所を指定しない場合、現在のシーケンスにフィルタが追加されます。場所を指定した場合、シーケンスの特定の場所にフィルタが挿入されます。詳細については、新しいメッセージ フィルタの作成 （119 ページ）を参照してください。
削除	名前またはシーケンス番号を指定して、フィルタを削除します。詳細については、メッセージ フィルタの削除 （120 ページ）を参照してください。
移動	既存のフィルタを並べ替えます。詳細については、新しいメッセージ フィルタの作成 （119 ページ）を参照してください。
設定	フィルタをアクティブまたは非アクティブ状態に設定します。詳細については、新しいメッセージ フィルタの作成 （119 ページ）を参照してください。
import	フィルタの現在のセットを、ファイル（アプライアンスの /configuration ディレクトリ）内に保存されている新しいセットに置き換えます。詳細については、新しいメッセージ フィルタの作成 （119 ページ）を参照してください。
export	フィルタの現在のセットを（アプライアンスの /configuration ディレクトリ内の）ファイルにエクスポートします。詳細については、メッセージ フィルタのエクスポート （124 ページ）を参照してください。
list	1 つ以上のフィルタに関する情報を一覧表示します。詳細については、メッセージ フィルタ リストの表示 （124 ページ）を参照してください。
detail	特定のフィルタに関する詳細情報（フィルタ ルール自体の本文など）を出力します。詳細については、メッセージ フィルタの詳細の表示 （125 ページ）を参照してください。
logconfig	フィルタの logconfig サブメニューを入力すると、archive() フィルタ アクションからログ サブスクリプションを編集できます。詳細については、フィルタ ログ サブスクリプションの構成 （125 ページ）を参照してください。

（注） フィルタを有効にするには、commit コマンドを発行する必要があります。

パラメータには、次の 3 つのタイプがあります。

表 *10* : フィルタ管理パラメータ

seqnum	フィルタのリスト内の位置に基づいてフィルタを表す整数です。たとえば、*seqnum* が 2 の場合、リスト内の 2 番めのフィルタを表します。
filtname	フィルタの表示名。
range	range は、複数のフィルタを表す場合に使用することがあり、「*X*-Y」の形式で表されます。X と Y は、範囲を指定するための最初と最後の *seqnums* です。たとえば、「2-4」は、2、3、4 番めの位置にあるフィルタを表します。X または Y のいずれかを省略すると、無制限のリストを表します。たとえば、「-4」は最初から 4 つのフィルタを表し、「2-」は、先頭以外のすべてのフィルタを表します。キーワード all を使用して、フィルタ リスト内のすべてのフィルタを表すこともできます。

関連項目

- 新しいメッセージ フィルタの作成 （119 ページ）
- メッセージ フィルタの削除 （120 ページ）
- メッセージ フィルタの移動 （120 ページ）
- メッセージ フィルタのアクティベーションとディアクティベーション （120 ページ）
- メッセージ フィルタのインポート （123 ページ）
- メッセージ フィルタのエクスポート （124 ページ）
- 非 ASCII 文字セットの表示 （124 ページ）
- メッセージ フィルタ リストの表示 （124 ページ）
- メッセージ フィルタの詳細の表示 （125 ページ）
- フィルタ ログ サブスクリプションの構成 （125 ページ）
- メッセージのエンコードの変更 （126 ページ）
- サンプル メッセージ フィルタ （128 ページ）

新しいメッセージ フィルタの作成

```
new [seqnum|filtname|last]
```

新しいフィルタを挿入する位置を指定します。省略するか、キーワード last を指定すると、入力されたフィルタがフィルタ リストの最後に追加されます。シーケンス番号は連続させる必要があります。現在のリストの範囲を超える *seqnum* は入力できません。不明な *filtname* を入力すると、有効な *filtname*、*seqnum*、または last を入力するように求められます。

フィルタを入力したら、手動でフィルタ スクリプトを入力する必要があります。入力を終了したら、その行自体にピリオド（.）を入力してエントリを終了します。

次の条件ではエラーが発生します。

- シーケンス番号が現在のシーケンス番号の範囲を超えている。
- フィルタに付けた *filtname* が一意ではない。

- フィルタに付けた *filtname* が予約語である。
- フィルタに構文エラーが発生している。
- インターフェイスなど、存在しないシステム リソースを参照するアクションを実行するフィルタ。

メッセージ フィルタの削除

```
delete [seqnum|filtname|range]
```

指定したフィルタを削除します。

次の条件ではエラーが発生します。

- 指定した名前のフィルタが存在しない。
- 指定したシーケンス番号のフィルタが存在しない。

メッセージ フィルタの移動

```
move [seqnum|filtname|rangeseqnum|last]
```

最初のパラメータで指定したフィルタを、2 番めのパラメータで指定した場所に移動します。2 番めのパラメータがキーワード last である場合、フィルタはフィルタ リストの最後に移動されます。複数のフィルタを移動する場合、それらのフィルタの相対的な順序は変わりません。

次の条件ではエラーが発生します。

- 指定した名前のフィルタが存在しない。
- 指定したシーケンス番号のフィルタが存在しない。
- シーケンス番号が現在のシーケンス番号の範囲を超えている。
- 移動してもシーケンスが変更されない。

メッセージフィルタのアクティベーションとディアクティベーション

指定されるメッセージ フィルタは、*active* または *inactive* のいずれかであり、さらに *valid* または *invalid* のいずれかです。メッセージ フィルタは、*active* と *valid* の両方の状態である場合にのみ処理に使用されます。CLI を使用して、既存のフィルタを active から inactive に変更します（その後、再び戻します）。存在しない（または削除された）リスナーまたはインターフェイスを参照している場合、そのフィルタは invalid です。

（注） フィルタが inactive であるかどうかは、構文から判断できます。AsyncOS では、inactive であるフィルタのフィルタ名に続くコロンが、感嘆符に変更されます。フィルタを入力またはインポートするときにこの構文を使用すると、AsyncOS はフィルタを inactive としてマークします。

たとえば、次のように無害な「filterstatus」という名前のフィルタを入力します。`filter -> set` サブコマンドを使用して、このフィルタを inactive にします。フィルタの詳細が表示され、コロンが感嘆符に変わっている点に注目してください（以下の例で、太字で示されています）。

```
mail3.example.com> filters

Choose the operation you want to perform:

- NEW - Create a new filter.

- IMPORT - Import a filter script from a file.

[]> new

Enter filter script. Enter '.' on its own line to end.

filterstatus: if true{skip-filters();}

.

1 filters added.

Choose the operation you want to perform:

- NEW - Create a new filter.

- DELETE - Remove a filter.

- IMPORT - Import a filter script from a file.

- EXPORT - Export filters to a file

- MOVE - Move a filter to a different position.

- SET - Set a filter attribute.

- LIST - List the filters.

- DETAIL - Get detailed information on the filters.

- LOGCONFIG - Configure log subscriptions used by filters.

- ROLLOVERNOW - Roll over a filter log file.

[]> list

Num Active Valid Name

1 Y Y filterstatus

Choose the operation you want to perform:

- NEW - Create a new filter.

- DELETE - Remove a filter.

- IMPORT - Import a filter script from a file.

- EXPORT - Export filters to a file

- MOVE - Move a filter to a different position.
```

```
- SET - Set a filter attribute.
- LIST - List the filters.
- DETAIL - Get detailed information on the filters.
- LOGCONFIG - Configure log subscriptions used by filters.
- ROLLOVERNOW - Roll over a filter log file.
[]> set
Enter the filter name, number, or range:
[all]> all
Enter the attribute to set:
[active]> inactive
1 filters updated.
Choose the operation you want to perform:
- NEW - Create a new filter.
- DELETE - Remove a filter.
- IMPORT - Import a filter script from a file.
- EXPORT - Export filters to a file
- MOVE - Move a filter to a different position.
- SET - Set a filter attribute.
- LIST - List the filters.
- DETAIL - Get detailed information on the filters.
- LOGCONFIG - Configure log subscriptions used by filters.
- ROLLOVERNOW - Roll over a filter log file.
[]> detail
Enter the filter name, number, or range:
[]> all
Num Active Valid Name
1 N Y filterstatus
filterstatus! if (true) {
skip-filters();
}
Choose the operation you want to perform:
- NEW - Create a new filter.
```

```
- DELETE - Remove a filter.

- IMPORT - Import a filter script from a file.

- EXPORT - Export filters to a file

- MOVE - Move a filter to a different position.

- SET - Set a filter attribute.

- LIST - List the filters.

- DETAIL - Get detailed information on the filters.

- LOGCONFIG - Configure log subscriptions used by filters.

- ROLLOVERNOW - Roll over a filter log file.

[]>
```

関連項目

- メッセージフィルタのアクティベーションまたはディアクティベーション（123ページ）

メッセージ フィルタのアクティベーションまたはディアクティベーション

`set [seqnum|filtname|range] active|inactive`

指定したフィルタを指定した状態に設定します。状態のルールは次のとおりです。

- active：選択したフィルタの状態を active に設定します。
- inactive：選択したフィルタの状態を inactive に設定します。

次の条件ではエラーが発生します。

- 指定した *filtname* のフィルタが存在しない。
- 指定したシーケンス番号のフィルタが存在しない。

（注） inactive であるフィルタは、構文からも判断できます。ラベル（フィルタ名）の後のコロンが、感嘆符（!）に変更されます。CLI から手動で入力された、またはインポートされたフィルタにこの構文が含まれる場合、自動的に inactive とマークされます。たとえば、mailfrompm! が、mailfrompm: の代わりに表示されます。

メッセージ フィルタのインポート

`import filename`

処理されるフィルタを含むファイルの名前です。このファイルは、アプライアンスの FTP/SCP ルートディレクトリの configuration ディレクトリ内に存在する必要があります（`interfaceconfig` コマンドを使用してインターフェイスの FTP/SCP アクセスを有効にしている場合）。ファイルは取り込まれて解析され、エラーが存在すれば報告されます。現在のフィルタ セット内に存在するすべてのフィルタは、インポートされたフィルタに置き換わります。詳細については、

FTP、SSH、およびSCPアクセスを参照してください。現在のフィルタ リストをエクスポートし（メッセージフィルタのエクスポート （124ページ）を参照）、そのファイルを編集してインポートすることを推奨します。

メッセージ フィルタをインポートする場合、使用するエンコードを選択するよう求められます。

次の条件ではエラーが発生します。

- ファイルが存在しない。
- フィルタ名が一意ではない。
- フィルタに付けた *filtname* が予約語である。
- フィルタに構文エラーが発生している。
- インターフェイスなど、存在しないシステム リソースを参照するアクションを実行するフィルタ。

メッセージ フィルタのエクスポート

`export filename[seqnum|filtname|range]`

既存のフィルタ セットを、アプライアンスのFTP/SCPルートディレクトリにあるconfigurationディレクトリ内のファイルに所定の形式で出力します。詳細については、FTP、SSH、およびSCP アクセスを参照してください。

メッセージフィルタをエクスポートする場合、使用するエンコードを選択するよう求められます。

次の条件ではエラーが発生します。

- 指定した名前のフィルタが存在しない。
- 指定したシーケンス番号のフィルタが存在しない。

非 ASCII 文字セットの表示

このシステムでは、CLI で非 ASCII 文字が UTF-8 で表示されます。お使いのターミナル/ディスプレイが UTF-8 をサポートしていない場合、フィルタが正常に表示されません。

フィルタ内の非 ASCII 文字を管理する最も良い方法は、フィルタをテキスト ファイルで編集してから、そのテキスト ファイルをアプライアンスにインポートすることです（メッセージフィルタのインポート （123 ページ）を参照）。

メッセージ フィルタ リストの表示

`list [seqnum|filtname|range]`

指定したフィルタの本文を出力せずに、概要を表形式で表示します。表示される情報は次のとおりです。

- フィルタ名
- フィルタ シーケンス番号

- フィルタの active/inactive 状態
- フィルタの valid/invalid 状態

次の条件ではエラーが発生します。

- 範囲の指定が不正である。

メッセージ フィルタの詳細の表示

`detail [seqnum|filtname|range]`

フィルタの本文や追加の状態情報など、指定したフィルタの情報をすべて表示します。

フィルタ ログ サブスクリプションの構成

`logconfig`

サブメニューを入力し、`archive()` アクションによって生成されたメールボックス ファイルのフィルタ ログ オプションを設定できます。これらのオプションは、通常の `logconfig` コマンドで使用されるオプションとよく似ていますが、ログを参照するフィルタを追加または削除することによってのみ、ログを作成または削除できます。

各フィルタ ログ サブスクリプションには次のデフォルト値が設定されています。この値は、logconfig サブコマンドを使用して変更できます。

- 取得方法：FTP Poll
- ファイル サイズ：10MB
- ファイルの最大数：10

詳細については、「ロギング」の章を参照してください。

```
mail3.example.com> filters

Choose the operation you want to perform:
- NEW - Create a new filter.
- DELETE - Remove a filter.
- IMPORT - Import a filter script from a file.
- EXPORT - Export filters to a file
- MOVE - Move a filter to a different position.
- SET - Set a filter attribute.
- LIST - List the filters.
- DETAIL - Get detailed information on the filters.
- LOGCONFIG - Configure log subscriptions used by filters.
- ROLLOVERNOW - Roll over a filter log file.
```

```
[]> logconfig

Currently configured logs:

1. "joesmith" Type: "Filter Logs" Retrieval: FTP Poll

Choose the operation you want to perform:

- EDIT - Modify a log setting.


[]> edit

Enter the number of the log you wish to edit.

[]> 1

Choose the method to retrieve the logs.

1. FTP Poll

2. FTP Push

3. SCP Push

[1]> 1

Please enter the filename for the log:

[joesmith.mbox]>

Please enter the maximum file size:

[10485760]>

Please enter the maximum number of files:

[10]>

Currently configured logs:

1. "joesmith" Type: "Filter Logs" Retrieval: FTP Poll

Enter "EDIT" to modify or press Enter to go back.

[]>
```

メッセージのエンコードの変更

`localeconfig` コマンドを使用して、メッセージ処理中のメッセージのヘッダーおよびフッターのエンコードの変更に関する AsyncOS の動作を設定できます。

```
example.com> localeconfig

Behavior when modifying headers: Use encoding of message body
Behavior for untagged non-ASCII headers: Impose encoding of message body
Behavior for mismatched footer or heading encoding: Try both body and footer or heading
 encodings
Behavior when decoding errors found: Disclaimer is displayed as inline content and the
message body is added as an attachment.
```

```
Choose the operation you want to perform:
- SETUP - Configure multi-lingual settings.
[]> setup

If a header is modified, encode the new header in the same encoding as the message body?

(Some MUAs incorrectly handle headers encoded in a different encoding than the body.
However, encoding a modified header in the same encoding as the message body may cause
certain
characters in the modified header to be lost.) [Y]>

If a non-ASCII header is not properly tagged with a character set and is being used or
modified,
impose the encoding of the body on the header during processing and final representation
 of the message?
(Many MUAs create non-RFC-compliant headers that are then handled in an undefined way.
Some MUAs handle headers encoded in character sets that differ from that of the main
body in an incorrect way.
Imposing the encoding of the body on the header may encode the header more precisely.
This will be used to interpret the content of headers for processing, it will not modify
 or rewrite the
header unless that is done explicitly as part of the processing.) [Y]>

Disclaimers (as either footers or headings) are added in-line with the message body
whenever possible.
However, if the disclaimer is encoded differently than the message body, and if imposing
 a single encoding
will cause loss of characters, it will be added as an attachment. The system will always
 try to use the
message body's encoding for the disclaimer. If that fails, the system can try to edit
the message body to
use an encoding that is compatible with the message body as well as the disclaimer.
Should the system try to
re-encode the message body in such a case? [Y]>

If the disclaimer that is added to the footer or header of the message generates an error
 when decoding the message body,
it is added at the top of the message body. This prevents you to rewrite a new message
content that must merge with
the original message content and the header/footer-stamp. The disclaimer is now added
as an additional MIME part
that displays only the header disclaimer as an inline content, and the rest of the message
 content is split into
separate email attachments. Should the system try to ignore such errors when decoding
the message body? [N]>

Behavior when modifying headers: Use encoding of message body
Behavior for untagged non-ASCII headers: Impose encoding of message body
Behavior for mismatched footer or heading encoding: Try both body and footer or heading
 encodings
Behavior when decoding errors found: Disclaimer is displayed as inline content and the
message body
is added as an attachment.

Choose the operation you want to perform:
- SETUP - Configure multi-lingual settings.
[]>
```

最初のプロンプトは、ヘッダーが（たとえばフィルタによって）変更されていた場合、メッセージヘッダーのエンコードをメッセージ本文に一致するように変更するかどうかを指定します。

2番めのプロンプトは、ヘッダーの文字セットが適切にタグで指定されていない場合、ヘッダーに対してメッセージ本文のエンコードを強制する必要があるかどうかを制御します。

3 番めのプロンプトは、免責事項のスタンプ（および複数のエンコード）がメッセージ本文でどのように機能するかを制御するために使用されます。詳細については、「テキスト リソース」の章の「免責事項スタンプと複数エンコード方式」を参照してください。

4 番めのプロンプトは、メッセージ本文のデコード時にエラーが生成された場合に、免責事項スタンプの動作を設定するために使用されます。[はい（Yes）] を選択するとデコード エラーは無視され、免責事項スタンプが行われます。[いいえ（No）] を選択すると、メッセージに免責事項テキストが添付されます。

サンプル メッセージ フィルタ

次の例では、filter コマンドを使用して新しいフィルタを 3 つ作成します。

- 最初のフィルタの名前は、**big_messages** です。これは body-size ルールを使用して、10 MB より大きいメッセージをドロップします。
- 2 番めのフィルタの名前は、**no_mp3s** です。これは attachment-filename ルールを使用して、.mp3 ファイル拡張子が付いた添付ファイルを含むメッセージをドロップします。
- 3 番めのフィルタの名前は、**mailfrompm** です。これは mail-from ルールを使用して、postmaster@example.com からのメールをすべて調べ、administrator@example.com のブラインド カーボン コピーを作成します。

filter -> list サブコマンドを使用し、フィルタのリストを表示して、フィルタがアクティブで有効であることを確認します。次に、move サブコマンドを使用して、最初と最後のフィルタの位置を入れ替えます。最後に、変更を確定してフィルタを有効にします。

```
mail3.example.com> filters

Choose the operation you want to perform:

- NEW - Create a new filter.

- IMPORT - Import a filter script from a file.

[]> new

Enter filter script. Enter '.' on its own line to end.

big_messages:

if (body-size >= 10M) {

drop();

}

.

1 filters added.

Choose the operation you want to perform:

- NEW - Create a new filter.
```

```
- DELETE - Remove a filter.
- IMPORT - Import a filter script from a file.
- EXPORT - Export filters to a file
- MOVE - Move a filter to a different position.
- SET - Set a filter attribute.
- LIST - List the filters.
- DETAIL - Get detailed information on the filters.
- LOGCONFIG - Configure log subscriptions used by filters.
- ROLLOVERNOW - Roll over a filter log file.
[]> new
Enter filter script. Enter '.' on its own line to end.
no_mp3s:
if (attachment-filename == '(?i)\\.mp3$') {
drop();
}
.
1 filters added.
Choose the operation you want to perform:
- NEW - Create a new filter.
- DELETE - Remove a filter.
- IMPORT - Import a filter script from a file.
- EXPORT - Export filters to a file
- MOVE - Move a filter to a different position.
- SET - Set a filter attribute.
- LIST - List the filters.
- DETAIL - Get detailed information on the filters.
- LOGCONFIG - Configure log subscriptions used by filters.
- ROLLOVERNOW - Roll over a filter log file.
[]> new
Enter filter script. Enter '.' on its own line to end.
mailfrompm:
if (mail-from == "^postmaster$")
```

```
{ bcc ("administrator@example.com");}

.

1 filters added.

Choose the operation you want to perform:

- NEW - Create a new filter.

- DELETE - Remove a filter.

- IMPORT - Import a filter script from a file.

- EXPORT - Export filters to a file

- MOVE - Move a filter to a different position.

- SET - Set a filter attribute.

- LIST - List the filters.

- DETAIL - Get detailed information on the filters.

- LOGCONFIG - Configure log subscriptions used by filters.

- ROLLOVERNOW - Roll over a filter log file.

[]> list

Num Active Valid Name

1 Y Y big_messages

2 Y Y no_mp3s

3 Y Y mailfrompm

Choose the operation you want to perform:

- NEW - Create a new filter.

- DELETE - Remove a filter.

- IMPORT - Import a filter script from a file.

- EXPORT - Export filters to a file

- MOVE - Move a filter to a different position.

- SET - Set a filter attribute.

- LIST - List the filters.

- DETAIL - Get detailed information on the filters.

- LOGCONFIG - Configure log subscriptions used by filters.

- ROLLOVERNOW - Roll over a filter log file.

[]> move

Enter the filter name, number, or range to move:
```

```
[]> 1

Enter the target filter position number or name:

[]> last

1 filters moved.

Choose the operation you want to perform:

- NEW - Create a new filter.

- DELETE - Remove a filter.

- IMPORT - Import a filter script from a file.

- EXPORT - Export filters to a file

- MOVE - Move a filter to a different position.

- SET - Set a filter attribute.

- LIST - List the filters.

- DETAIL - Get detailed information on the filters.

- LOGCONFIG - Configure log subscriptions used by filters.

- ROLLOVERNOW - Roll over a filter log file.

[]> list

Num Active Valid Name

1 Y Y no_mp3s

2 Y Y mailfrompm

3 Y Y big_messages

Choose the operation you want to perform:

- NEW - Create a new filter.

- DELETE - Remove a filter.


- IMPORT - Import a filter script from a file.

- EXPORT - Export filters to a file

- MOVE - Move a filter to a different position.

- SET - Set a filter attribute.

- LIST - List the filters.

- DETAIL - Get detailed information on the filters.

- LOGCONFIG - Configure log subscriptions used by filters.

- ROLLOVERNOW - Roll over a filter log file.

[]> move
```

```
Enter the filter name, number, or range to move:
[]> 2
Enter the target filter position number or name:
[]> 1
1 filters moved.
Choose the operation you want to perform:
- NEW - Create a new filter.
- DELETE - Remove a filter.
- IMPORT - Import a filter script from a file.
- EXPORT - Export filters to a file
- MOVE - Move a filter to a different position.
- SET - Set a filter attribute.
- LIST - List the filters.
- DETAIL - Get detailed information on the filters.
- LOGCONFIG - Configure log subscriptions used by filters.
- ROLLOVERNOW - Roll over a filter log file.

[]> list
Num Active Valid Name
1 Y Y mailfrompm
2 Y Y no_mp3s
3 Y Y big_messages
Choose the operation you want to perform:
- NEW - Create a new filter.
- DELETE - Remove a filter.
- IMPORT - Import a filter script from a file.
- EXPORT - Export filters to a file

- MOVE - Move a filter to a different position.
- SET - Set a filter attribute.
- LIST - List the filters.

- DETAIL - Get detailed information on the filters.
```

```
- LOGCONFIG - Configure log subscriptions used by filters.

- ROLLOVERNOW - Roll over a filter log file.

[]>

mail3.example.com> commit

Please enter some comments describing your changes:

[]> entered and enabled 3 filters: no_mp3s, mailfrompm, big_messages

Do you want to save the current configuration for rollback? [Y]> n

Changes committed: Fri May 23 11:42:12 2014 GMT
```

メッセージ フィルタの例

この項では、実際のフィルタの例を示し、各フィルタについて簡単に説明します。

関連項目

- オープンリレー防止フィルタ （133 ページ）
- ポリシー適用フィルタ （134 ページ）
- ルーティングおよびドメイン スプーフィング （137 ページ）

オープンリレー防止フィルタ

このフィルタは、次のように電子メール アドレスに %、余分な @、および ! 文字が含まれているメッセージをバウンスします。

- `user%otherdomain@validdomain`
- `user@otherdomain@validdomain:`
- `domain!user@validdomain`

```
sourceRouted:

if (rcpt-to == "(%|@|!)(.*)@") {

bounce();

}
```

Cisco アプライアンスは、従来の Sendmail/Qmail システムを活用するためによく使用される、このようなサード パーティ製のリレー ハックの影響を受けません。これらの記号の多く（% など）は正当な電子メール アドレスの一部である可能性があるため、Cisco アプライアンスはこれらを有効なアドレスとして受け入れ、設定済みの受信者リストと照合し、次の内部サーバに渡します。Cisco アプライアンスは、これらのメッセージを外部にリレーしません。

このようなフィルタは、このタイプのメッセージをリレーできるように誤って設定されたオープンソース MTA を使用しているユーザを保護するために所定の場所に設定されます。

（注） このようなタイプのアドレスを処理するように、リスナーを設定することもできます。詳細については、Webインターフェイスを使用してリスナーを作成することによる接続要求のリスニングを参照してください。

ポリシー適用フィルタ

- 件名に基づき通知するフィルタ （134 ページ）
- 競合他社に送信されたメールの BCC およびスキャン （134 ページ）
- 特定のユーザをブロックするフィルタ （135 ページ）
- メッセージのアーカイブおよびドロップ フィルタ （135 ページ）
- 大きい「To:」ヘッダーのフィルタ （135 ページ）
- 空白の「From:」フィルタ （136 ページ）
- SRBS フィルタ （136 ページ）
- SRBS 変更フィルタ （136 ページ）
- ファイル名の正規表現フィルタ （136 ページ）
- ヘッダー内の SenderBase レピュテーション スコアの表示フィルタ （137 ページ）
- ポリシーのヘッダーへの挿入フィルタ （137 ページ）
- 多数の受信者のバウンス フィルタ （137 ページ）

件名に基づき通知するフィルタ

このフィルタは、件名に特定の用語が含まれているかどうかに基づいて通知を送信します。

```
search_for_sensitive_content:

if (Subject == "(?i)plaintiff|lawsuit|judge" ) {

notify ("admin@company.com");

}
```

競合他社に送信されたメールの BCC およびスキャン

このフィルタは、競合他社に送信されたメッセージをスキャンし、ブラインドコピーを作成します。辞書と `header-dictionary-match()` ルールを使用して、柔軟性の高い競合他社のリストを指定できます（ディクショナリ ルール （47 ページ）を参照）。

```
competitorFilter:

if (rcpt-to == '@competitor1.com|@competitor2.com') {

bcc-scan('legal@example.com');
```

```
}
```

特定のユーザをブロックするフィルタ

このフィルタを使用すると、特定のアドレスからの電子メールをブロックします。

```
block_harrasing_user:
if (mail-from == "ex-employee@hotmail\\.com") {
notify ("admin@company.com");
drop ();
}
```

メッセージのアーカイブおよびドロップ フィルタ

ファイルタイプが一致するメッセージのみをログ記録およびドロップします。

```
drop_attachments:
if (mail-from != "user@example.com") AND (attachment-filename ==
'(?i)\\.(asp|bas|bat|cmd|cpl|exe|hta|ins|isp|js)$')
{
archive("Drop_Attachments");
insert-header("X-Filter", "Dropped by: $FilterName MID: $MID");
drop-attachments-by-name("\\.(asp|bas|bat|cmd|cpl|exe|hta|ins|isp|js)$");
}
```

大きい「To:」ヘッダーのフィルタ

「To」ヘッダーが非常に大きいメッセージを検索します。

`archive()` 行を使用して適切なアクションを検証し、drop() をイネーブルまたはディセーブルにして安全性を高めます。

```
toTooBig:
if(header('To') == "^.{500,}") {
archive('tooTooBigdropped');
drop();
}
```

空白の「From:」フィルタ

空白の「From」ヘッダーを特定します。

このフィルタは、「from」アドレスが空白であるさまざまな形式に対応できます。

```
blank_mail_from_stop:

if (recv-listener == "InboundMail" AND header("From") == "^$|<\\s*>") {

drop ();

}
```

また、Envelope Fromが空欄のメッセージをドロップする場合は、次のフィルタを使用します。

```
blank_mail_from_stop:

if (recv-listener == "InboundMail" AND (mail-from == "^$|<\\s*>" OR header ("From") ==
"^$|<\\s*>"))

{

drop ();
}
```

SRBS フィルタ

SenderBase レピュテーション フィルタ：

```
note_bad_reps:
if (reputation < -2) {
strip-header ('Subject');

insert-header ('Subject', '***BadRep $Reputation *** $Subject');
}
```

SRBS 変更フィルタ

特定のドメインの SenderBase Reputation Score（SBRS; SenderBase レピュテーション スコア）しきい値を変更します。

```
mod_sbrs:
if ( (rcpt-count == 1) AND (rcpt-to == "@domain\\.com$") AND (reputation < -2) ) {
drop ();
}
```

ファイル名の正規表現フィルタ

このフィルタは、メッセージ本文のサイズの範囲を指定し、正規表現に一致する添付ファイルを検索します（このパターンに一致するファイル名は、「readme.zip」、「readme.exe」、「attach.exe」、など）。

```
filename_filter:
if ((body-size >= 9k) AND (body-size <= 20k)) {
if (body-contains ("(?i)(readme|attach|information)\\.(zip|exe)$")) {
drop ();

}

}
```

ヘッダー内の SenderBase レピュテーション スコアの表示フィルタ

ヘッダーのログが記録されるので、メール ログで表示できます（「ロギング」の章を参照）。

```
Check_SBRS:

if (true) {

insert-header('X-SBRS', '$Reputation');

}
```

ポリシーのヘッダーへの挿入フィルタ

どのメール フロー ポリシーが接続を受け入れたかを示します。

```
Policy_Tracker:

if (true) {

insert-header ('X-HAT', 'Sender Group $Group, Policy $Policy applied.');

}
```

多数の受信者のバウンス フィルタ

3 つ以上の固有ドメインから 50 人を超える受信者が指定されている発信電子メール メッセージをすべてバウンスします。

```
bounce_high_rcpt_count:

if ( (rcpt-count > 49) AND (rcpt-to != "@example\\.com$") ) {

bounce-profile ("too_many_rcpt_bounce"); bounce ();

}
```

ルーティングおよびドメイン スプーフィング

- Virtual Gateway フィルタの使用 （138 ページ）
- 配信とリスナーのフィルタに対する同じリスナーの使用 （138 ページ）

• 単一のリスナーのフィルタ （138 ページ）
• スプーフィング ドメインのドロップ フィルタ（単一のリスナー） （139 ページ）
• スプーフィング ドメインのドロップ フィルタ（複数のリスナー） （139 ページ）
• 別のスプーフィング ドメインのドロップ フィルタ （139 ページ）
• ルーピングの検出フィルタ （140 ページ）

Virtual Gateway フィルタの使用

仮想ゲートウェイを使用してトラフィックを区分します。システムに2つのインターフェイス「public1」と「public2」が存在するとします。デフォルトの配信インターフェイスは「public1」です。これにより、発信トラフィックはすべて2番めのインターフェイスを介すように強制されます。バウンスおよびその他同様のタイプのメールはフィルタを通過しないため、そのようなメールは public1 から配信されます。

```
virtual_gateways:

if (recv-listener == "OutboundMail") {

alt-src-host ("public2");

}
```

配信とリスナーのフィルタに対する同じリスナーの使用

配信と受信に同じリスナーを使用します。このフィルタでは、パブリック リスナー「listener1」で受信したメッセージを、インターフェイス「listener1」から送信できます（設定したパブリック インジェクタごとに、固有のフィルタをセットアップする必要があります）。

```
same_listener:

if (recv-inj == 'listener1') {

alt-src-host('listener1');

}
```

単一のリスナーのフィルタ

単一のリスナーでフィルタを機能させます。たとえば、システム全体で実行するのではなく、メッセージ フィルタを処理する専用のリスナーを指定します。

```
textfilter-new:

if (recv-inj == 'inbound' and body-contains("some spammy message")) {

alt-rcpt-to ("spam.quarantine@spam.example.com");

}
```

スプーフィング ドメインのドロップ フィルタ（単一のリスナー）

スプーフィング ドメイン（内部のアドレスからであると偽り、単一のリスナーで機能する）が使用されている電子メールをドロップします。以下の IP アドレスは、架空のドメイン mycompany.com を表しています。

```
DomainSpoofed:
if (mail-from == "mycompany\\.com$") {
if ((remote-ip != "1.2.") AND (remote-ip != "3.4.")) {
drop();
}
}
```

スプーフィング ドメインのドロップ フィルタ（複数のリスナー）

前述と同じですが、複数のリスナーを使用して動作します。

```
domain_spoof:
if ((recv-listener == "Inbound") and (mail-from == "@mycompany\\.com")) {
archive('domain_spoof');
drop ();
}
```

別のスプーフィング ドメインのドロップ フィルタ

概要：ドメイン スプーフィング対策フィルタ：

```
reject_domain_spoof:
if (recv-listener == "MailListener") {
insert-header("X-Group", "$Group");
if ((mail-from == "@test\\.mycompany\\.com") AND (header("X-Group") != "RELAYLIST")) {
notify("me@here.com");
drop();
strip-header("X-Group");
}
```

ループピングの検出フィルタ

このフィルタを使用して、メールループを発生させている要因を検出、停止、および判断します。このフィルタは、Exchangeサーバまたはそれ以外の場所で発生している構成の問題を判断するために役立ちます。

```
External_Loop_Count:

if (header("X-ExtLoop1")) {


if (header("X-ExtLoopCount2")) {

if (header("X-ExtLoopCount3")) {

if (header("X-ExtLoopCount4")) {

if (header("X-ExtLoopCount5")) {

if (header("X-ExtLoopCount6")) {

if (header("X-ExtLoopCount7")) {

if (header("X-ExtLoopCount8")) {

if (header("X-ExtLoopCount9")) {

notify ('joe@example.com');

drop();

}

else {insert-header("X-ExtLoopCount9", "from
$RemoteIP");}}

else {insert-header("X-ExtLoopCount8", "from $RemoteIP");}}

else {insert-header("X-ExtLoopCount7", "from $RemoteIP");}}

else {insert-header("X-ExtLoopCount6", "from $RemoteIP");}}

else {insert-header("X-ExtLoopCount5", "from $RemoteIP");}}

else {insert-header("X-ExtLoopCount4", "from $RemoteIP");}}

else {insert-header("X-ExtLoopCount3", "from $RemoteIP");}}

else {insert-header("X-ExtLoopCount2", "from $RemoteIP");}}

else {insert-header("X-ExtLoop1", "1");

}
```

（注） デフォルトでは、AsyncOSは自動的にメールのループを検出し、100回ループしたメッセージをドロップします。

スキャン動作の設定

スキャン パラメータを設定することで、本文と添付ファイルのスキャン動作（スキャン中にスキップする添付ファイルのタイプなど）を制御できます。これらのパラメータを設定するには、[スキャン動作（Scan Behavior）] ページまたは `scanconfig` コマンドを使用します。スキャン動作の設定はグローバルな設定であるため、すべてのスキャンの動作に影響します。

（注） zip などの圧縮ファイルに含まれる MIME タイプをスキャンする場合、スキャン リストに「compressed」または「zip」または「application/zip」リストを含める必要があります。

ステップ 1 [セキュリティサービス（Security Services）] > [スキャン動作（Scan Behavior）] をクリックします。

ステップ 2 添付ファイル タイプのマッピングを定義します。次のいずれかを実行します。

- 新しい添付ファイル タイプのマッピングを追加する。[マッピングの追加（Add Mappin）] をクリックします。
- 設定ファイルを使用して添付ファイル タイプ マッピングのリストをインポートする。[インポートリスト（Import List）] をクリックし、configuration ディレクトリから該当する設定ファイルをインポートします。

  （注） この手順を実行するためには、設定ファイルが、アプライアンスの configuration ディレクトリに存在する必要があります。設定ファイルの管理を参照してください。

- 既存の添付ファイル タイプ マッピングを変更するには [編集（Edit）] をクリックします。

ステップ 3 グローバル設定を行います。次の手順を実行します。

a) [グローバル設定（Global Settings）] で、[グローバル設定を編集（Edit Global Settings）] をクリックします。

b) 目的のフィールドを編集します。

フィールド	説明
上記の表にある MIME タイプ/フィンガープリントの添付ファイルの場合のアクション（Action for attachments with MIME types / fingerprints in table above）	添付ファイル タイプ マッピングで定義されている添付ファイル タイプをスキャンするか、またはスキップするかを選択します。
スキャンする添付ファイル繰り返しの最大深度（Maximum depth of attachment recursion to scan）	スキャンする添付ファイルの繰り返しの最大レベルを指定します。
スキャンする最大添付ファイルサイズ（Maximum attachment size to scan）	スキャンする添付ファイルの最大サイズを指定します。

フィールド	説明
添付ファイルメタデータスキャン（**Attachment Metadata scan**）	添付ファイルのメタデータをスキャンするか、またはスキップするかを指定します。
添付ファイルスキャンタイムアウト（**Attachment scanning timeout**）	スキャンのタイムアウト期間を指定します。
何らかの理由でスキャンされない場合、添付ファイルがパターンに一致するものと仮定します（**Assume attachment matches pattern if not scanned for any reason**）	スキャンされない添付ファイルを検索パターンに一致するものとみなすかどうかを指定します。
メッセージを分解し、指定の添付ファイルを削除できないときのアクション（**Action when message cannot be deconstructed to remove specified attachments**）	指定の添付ファイルを削除するためにメッセージを分解できないときに実行するアクションを指定します。
コンテンツまたはメッセージフィルタエラーの場合、すべてのフィルタをバイパスします（**Bypass all filters in case of a content or message filter error**）	コンテンツまたはメッセージフィルタエラーの場合にすべてのフィルタをバイパスするかどうかを指定します。
何も指定されていないときに使用する符号化（**Encoding to use when none is specified**）	エンコーディングが指定されていない場合に使用するエンコーディングを指定します。
不透明な署名の付いたメッセージを明瞭な署名のものに変換する(S/MIME アンパック)（**Convert opaque-signed messages to clear-signed (S/MIME unpacking)**）	不透明な署名の付いたメッセージを明瞭な署名のものに変換する(S/MIME アンパック)かどうかを指定します。
URL フィルタリング アクション中に検出された復号エラーによるスキャン不可メッセージに対するアクション	URL フィルタリング アクション中に検出された復号エラーが原因で、コンテンツ スキャナによってスキャンできないメッセージに対して実行するアクションを指定します。
抽出エラーによるスキャン不可メッセージに対するアクション	添付ファイル抽出エラーが原因で、メッセージをコンテンツ スキャナでスキャンできない場合に実行するアクションを指定します。
RFC違反によるスキャン不可メッセージに対するアクション	RFC 違反が原因で、メッセージをコンテンツ スキャナでスキャンできない場合に実行するアクションを指定します。

c) [送信（Submit）] をクリックします。

ステップ 4 （任意）コンテンツ スキャナ ファイルを手動で更新します。[現在のコンテンツ スキャナ ファイル（Current Content Scanner files）] で [今すぐ更新（Update Now）] をクリックします。

通常、これらのファイルは、アップデート サーバを使用して自動的に更新されます。

（注） CLI で contentscannerupdate を使用して、これらのファイルを手動で更新することもできます。

ステップ 5 変更を確定します。

スキャンできないメッセージのメッセージ処理アクションの設定

アプライアンスのコンテンツ スキャナを使用して、以下の理由によりスキャンされないメッセージを処理できるようになりました。

- ファイル抽出失敗
- RFC 違反
- URL フィルタリング アクション中に検出された復号エラー

コンテンツ スキャナによってスキャンされないメッセージに対する、次のいずれかのメッセージ処理アクションを構成できます。

- メッセージのドロップ
- メッセージをそのまま配信
- ポリシー隔離へのメッセージの送信

Web インターフェイスの [セキュリティ サービス（Security Services）] > [スキャン動作（Scan Behavior）] ページで [グローバル設定の編集（Edit Global Settings）] ボタンをクリックして、コンテンツ スキャナによってスキャンされないメッセージに対するメッセージ処理アクションを有効にして構成することができます。

メッセージの配信

メッセージを配信する場合に、次の操作を実行できます。

- メッセージの件名を変更します。
- メッセージへのカスタム ヘッダーの追加
- メッセージの受信者の変更
- 代替宛先ホストへのメッセージの送信

（注） これらのアクションは、相互に排他的ではありません。ユーザのグループのさまざまな処理ニーズに合わせて、さまざまな着信または発信ポリシーで、これらのアクションを数個またはすべてを、さまざまに組み合わせることができます。

メッセージの件名を修正

コンテンツ スキャナによってスキャンされないメッセージのテキストを、特定のテキスト文字列を前に付加または後に付加することにより変更して、ユーザが容易に識別でき、識別されたメッセージを並べ替えることができるようになります。

（注） [メッセージの件名を修正（Modify message subject）] フィールドでは、空白は無視されません。このフィールドに入力したテキストの後ろまたは前にスペース追加することで、オリジナルのメッセージ件名と、追加テキストを分けることができます（追加テキストをオリジナルの件名の前に追加する場合は追加テキストの前、オリジナルの件名の後ろに追加する場合は追加テキストの後ろにスペースを追加します）。たとえば、`[WARNING: UNSCANNABLE EXTRACTION FAILURE]` というテキストをオリジナルの件名の前に追加する場合は、この後ろに数個のスペースを追加します。

コンテンツ スキャナによってスキャンされないメッセージの件名に追加されるデフォルトのテキストは次のとおりです。

理由（**Reason**）	件名へのデフォルトのテキストの追加
抽出エラー	[WARNING: UNSCANNABLE EXTRACTION FAILED]
RFC 違反	[WARNING: UNSCANNABLE RFC NON-COMPLIANT]
URL フィルタリング アクション中に検出された復号エラー	[WARNING: DECODING ERRORS WHEN APPLYING URL FILTERING ACTIONS]

メッセージへのカスタム ヘッダーの追加

コンテンツ スキャナによってスキャンされないすべてのメッセージに追加する、追加のカスタム ヘッダーを定義できます。[はい（Yes）] をクリックし、ヘッダー名およびテキストを定義します。

メッセージの受信者の変更

メッセージの受信者を変更して、コンテンツ スキャナによってスキャンされないメッセージが別のアドレスに送信されるようにできます。[はい（Yes）] をクリックして、新しい受信者のアドレスを入力します。

代替宛先ホストへのメッセージの送信

コンテンツ スキャナによってスキャンされないメッセージに対し、別の受信者または宛先ホストに通知を送信することができます。[はい（Yes）] をクリックして代替アドレスまたはホストを入力します。

たとえば、コンテンツ スキャナによってスキャンされないメッセージを、管理者のメールボックスまたは特別なメール サーバに、その後の調査のためにルーティングできます。受信者が複数のメッセージの場合は、代替受信者に送信されるコピーは 1 つのみです。

ポリシー隔離へメッセージの送信

隔離に関するフラグが設定されると、コンテンツスキャナによってスキャンされないメッセージは、電子メールパイプラインのそれ以降の部分を引き続き通過していきます。メッセージがパイプラインの末尾に到達すると、メッセージに1つ以上の隔離に関するフラグが設定されていれば、該当するキューに入ります。メッセージがパイプラインの末尾に到達しなければ、隔離エリアには配置されません。

たとえば、コンテンツフィルタはメッセージをドロップまたは返送する場合がありますが、その場合、メッセージは隔離されません。

（注） アプライアンスにポリシー隔離が定義されていない場合、メッセージを隔離エリアに送ることはできません。

メッセージをポリシー隔離に送ることを選択すると、次の追加アクションを実行できます。

- メッセージの件名を変更します。
- メッセージへのカスタム ヘッダーの追加

メッセージの件名ヘッダーの変更

特定のテキスト文字列を前後に追加することで、ポリシー隔離に送られたメッセージを変更すると、ユーザは、識別されたメッセージを判別したり、ソートしたりすることが簡単にできるようになります。

（注） [メッセージの件名を修正（Modify message subject）]フィールドでは、空白は無視されません。このフィールドに入力したテキストの後ろまたは前にスペース追加することで、オリジナルのメッセージ件名と、追加テキストを分けることができます（追加テキストをオリジナルの件名の前に追加する場合は追加テキストの前、オリジナルの件名の後ろに追加する場合は追加テキストの後ろにスペースを追加します）。たとえば、`[WARNING: UNSCANNABLE EXTRACTION FAILURE]` というテキストをオリジナルの件名の前に追加する場合は、この後ろに数個のスペースを追加します。

ポリシー隔離に送られたメッセージの件名に追加される既定のテキスト。

理由（Reason）	件名へのデフォルトのテキストの追加
抽出エラー	[WARNING: UNSCANNABLE EXTRACTION FAILED]
RFC 違反	[WARNING: UNSCANNABLE RFC NON-COMPLIANT]
URL フィルタリング アクション中に検出された復号エラー	[WARNING: DECODING ERRORS WHEN APPLYING URL FILTERING ACTIONS]

メッセージへのカスタム ヘッダーの追加

ポリシー隔離に送られたすべてのメッセージに追加する、追加のカスタム ヘッダーを定義できます。[はい（Yes）] をクリックし、ヘッダー名およびテキストを定義します。