CHAPTER 9

音声とビデオのプロトコルのインスペクション

ここでは、音声とビデオのプロトコルのアプリケーション インスペクションについて説明します。特定のプロトコルに関してインスペクションを使用する必要がある理由、およびインスペクションを適用する全体的な方法については、「アプリケーション レイヤ プロトコル インスペクションの準備」（P.7-1）を参照してください。

CTIQBE インスペクション

CTIQBE プロトコル インスペクションは、NAT、PAT、および双方向 NAT をサポートします。これによって、Cisco IP SoftPhone と他の Cisco TAPI/JTAPI アプリケーションが Cisco CallManager と連動し、ASA を越えてコール セットアップを行えるようになります。

TAPI と JTAPI は、多くの Cisco VoIP アプリケーションで使用されます。CTIQBE は、Cisco TSP が Cisco CallManager と通信するために使用されます。

CTIQBE インスペクションをイネーブルにする方法については、「アプリケーション レイヤ プロトコル インスペクションの設定」（P.7-11）を参照してください。

CTIQBE インスペクションの制限事項

CTIQBE アプリケーション インスペクションの使用時に適用される制限を次にまとめます。

- CTIQBE アプリケーション インスペクションは、**alias** コマンドを使用するコンフィギュレーションをサポートしません。
- CTIQBE コールのステートフル フェールオーバーはサポートされていません。
- **debug ctiqbe** コマンドを入力すると、メッセージの伝送が遅れ、リアルタイム環境のパフォーマンスに影響を与える場合があります。このデバッグまたはログをイネーブルにし、ASA を介して Cisco IP SoftPhone でコール セットアップを完了できない場合は、Cisco IP SoftPhone の動作するシステムで Cisco TSP 設定のタイムアウト値を増やしてください。

次に、CTIQBE アプリケーション インスペクションを特定の事例で使用する際に、特別に注意が必要な事項をまとめます。

- 2 つの Cisco IP SoftPhone が異なる Cisco CallManager に登録されていて、各 CallManager が ASA の異なるインターフェイスに接続されている場合、これら 2 つの電話間のコールは失敗します。
- Cisco IP SoftPhone と比較して Cisco CallManager の方がセキュリティの高いインターフェイス上に配置されている状態で、NAT または外部 NAT が Cisco CallManager IP アドレスに必要な場合、マッピングはスタティックである必要があります。Cisco IP SoftPhone では Cisco CallManager IP アドレスを PC 上の Cisco TSP コンフィギュレーションで明示的に指定することが必要なためです。
- PAT または外部 PAT の使用時に Cisco CallManager IP アドレスを変換する場合、Cisco IP SoftPhone を正常に登録させるには、TCP ポート 2748 を PAT（インターフェイス）アドレスと同じポートにスタティックにマッピングする必要があります。CTIQBE 受信ポート（TCP 2748）は固定されていて、Cisco CallManager、Cisco IP SoftPhone、Cisco TSP のいずれにおいてもユーザによる設定はできません。

CTIQBE インスペクションの確認とモニタリング

show ctiqbe コマンドは、ASA を越えて確立されている CTIQBE セッションに関する情報を表示します。CTIQBE インスペクション エンジンで割り当てられたメディア接続に関する情報が表示されます。

次の条件における **show ctiqbe** コマンドの出力例を示します。ASA を越えてセットアップされているアクティブ CTIQBE セッションは 1 つだけです。そのセッションは、ローカル アドレス 10.0.0.99 の内部 CTI デバイス（たとえば、Cisco IP SoftPhone）と 172.29.1.77 の外部 Cisco CallManager の間で確立されています。ここで、TCP ポート 2748 は、Cisco CallManager です。このセッションのハートビート間隔は 120 秒です。

```
hostname# # show ctiqbe

Total: 1
        LOCAL           FOREIGN         STATE   HEARTBEAT
-----------------------------------------------------------------
1       10.0.0.99/1117  172.29.1.77/2748        1       120
        -----------------------------------------------------------------
        RTP/RTCP: PAT xlates: mapped to 172.29.1.99(1028 - 1029)
        -----------------------------------------------------------------
        MEDIA: Device ID 27     Call ID 0
               Foreign 172.29.1.99      (1028 - 1029)
               Local   172.29.1.88      (26822 - 26823)
        -----------------------------------------------------------------
```

CTI デバイスは、すでに CallManager に登録されています。デバイスの内部アドレスおよび RTP 受信ポートは 172.29.1.99 の UDP ポート 1028 に PAT 変換されています。RTCP 受信ポートは UDP 1029 に PAT 変換されています。

`RTP/RTCP: PAT xlates:` で始まる行は、内部 CTI デバイスが外部 CallManager に登録され、CTI デバイスのアドレスとポートがその外部インターフェイスに PAT 変換されている場合に限り表示されます。この行は、CallManager が内部インターフェイス上にある場合、または内部 CTI デバイスのアドレスとポートが、CallManager が使用しているのと同じ外部インターフェイスに変換されている場合は、表示されません。

この出力は、コールがこの CTI デバイスと 172.29.1.88 にある別の電話機の間に確立されていることを示します。他の電話機の RTP および RTCP 受信ポートは、UDP 26822 および 26823 です。ASA は 2 番目の電話機と CallManager に関連する CTIQBE セッション レコードを維持できないので、他の電話機は、CallManager と同じインターフェイス上にあります。CTI デバイス側のアクティブ コール レッグは、Device ID 27 および Call ID 0 で確認できます。

これらの CTIQBE 接続の **show xlate debug** コマンドの出力例を示します。

```
hostname# show xlate debug
3 in use, 3 most used
Flags:  D - DNS, d - dump, I - identity, i - inside, n - no random,
        r - portmap, s - static
TCP PAT from inside:10.0.0.99/1117 to outside:172.29.1.99/1025 flags ri idle 0:00:22
timeout 0:00:30
UDP PAT from inside:10.0.0.99/16908 to outside:172.29.1.99/1028 flags ri idle 0:00:00
timeout 0:04:10
UDP PAT from inside:10.0.0.99/16909 to outside:172.29.1.99/1029 flags ri idle 0:00:23
timeout 0:04:10
```

show conn state ctiqbe コマンドは、CTIQBE 接続のステータスを表示します。出力には、CTIQBE インスペクション エンジンによって割り当てられたメディア接続が「C」フラグで示されます。次に、**show conn state ctiqbe** コマンドの出力例を示します。

```
hostname# show conn state ctiqbe
1 in use, 10 most used
hostname# show conn state ctiqbe detail
1 in use, 10 most used
Flags: A - awaiting inside ACK to SYN, a - awaiting outside ACK to SYN,
       B - initial SYN from outside, C - CTIQBE media, D - DNS, d - dump,
       E - outside back connection, F - outside FIN, f - inside FIN,
       G - group, g - MGCP, H - H.323, h - H.225.0, I - inbound data,
       i - incomplete, J - GTP, j - GTP data, k - Skinny media,
       M - SMTP data, m - SIP media, O - outbound data, P - inside back connection,
       q - SQL*Net data, R - outside acknowledged FIN,
       R - UDP RPC, r - inside acknowledged FIN, S - awaiting inside SYN,
       s - awaiting outside SYN, T - SIP, t - SIP transient, U - up
```

H.323 インスペクション

ここでは、H.323 アプリケーション インスペクションについて説明します。

H.323 インスペクションの概要

H.323 インスペクションは、Cisco CallManager や VocalTec Gatekeeper など、H.323 準拠のアプリケーションをサポートします。H.323 は、国際電気通信連合によって定義されている、LAN を介したマルチメディア会議用のプロトコル群です。ASA は、H.323 v3 機能の同一コール シグナリング チャネルでの複数コールを含めて、H.323 を Version 6 までサポートします。

H.323 インスペクションをイネーブルにした場合、ASA は、H.323 Version 3 で導入された機能である同一コール シグナリング チャネルでの複数コールをサポートします。この機能によってセットアップ時間が短縮され、ASA でのポート使用が減少します。

H.323 インスペクションの 2 つの主要機能は次のとおりです。

- H.225 と H.245 の両メッセージ内に埋め込まれている必要な IPv4 アドレスを NAT 処理します。H.323 メッセージは PER 符号化形式で符号化されているため、ASA では ASN.1 デコーダを使用して H.323 メッセージを復号化します。
- ネゴシエートされた H.245 と RTP/RTCP 接続をダイナミックに割り当てます。RAS を使用すると、H.225 接続もダイナミックに割り当てることができます。

H.323 の動作

H.323 のプロトコルのコレクションは、合計で最大 2 つの TCP 接続と 4 ～ 8 つの UDP 接続を使用できます。FastConnect は 1 つの TCP 接続だけを使用し、RAS は登録、アドミッション、およびステータス用に 1 つの UDP 接続を使用します。

H.323 クライアントは、最初に TCP ポート 1720 を使用して、H.323 サーバへの TCP 接続を確立し、Q.931 コール セットアップを要求します。H.323 端末は、コール セットアップ プロセスの一部として、H.245 TCP 接続に使用するため、クライアントに 1 つのポート番号を供給します。H.323 ゲートキーパーが使用されている環境では、初期パケットは UDP を使用して送信されます。

H.323 インスペクションは、Q.931 TCP 接続をモニタして、H.245 ポート番号を決定します。H.323 端末が、FastConnect を使用していない場合は、ASA が H.225 メッセージのインスペクションに基づいて、H.245 接続をダイナミックに割り当てます。

（注） RAS を使用すると、H.225 接続もダイナミックに割り当てることができます。

各 H.245 メッセージ内で、H.323 エンドポイントが、後続の UDP データ ストリームに使用するポート番号を交換します。H.323 インスペクションは、H.245 メッセージを調査して、ポート番号を識別し、メディア交換用の接続をダイナミックに作成します。RTP はネゴシエートされたポート番号を使用し、RTCP はその次に高いポート番号を使用します。

H.323 制御チャネルは、H.225、H.245、および H.323 RAS を処理します。H.323 インスペクションでは、次のポートが使用されます。

- 1718：ゲートキーパー検出 UDP ポート
- 1719：RAS UDP ポート
- 1720：TCP 制御ポート

RAS シグナリング用に予約済み H.323 ポート 1719 のトラフィックを許可する必要があります。さらに、H.225 コール シグナリング用に、予約済み H.323 ポート 1720 のトラフィックを許可する必要があります。ただし、H.245 シグナリング ポートは、H.225 シグナリングのエンドポイント間でネゴシエートされます。H.323 ゲートキーパーの使用時、ASA は、ACF メッセージと RCF メッセージのインスペクションに基づいて H.225 接続を開きます。

H.225 メッセージを検査した後、ASA は H.245 チャネルを開き、H.245 チャネルで送信されるトラフィックも検査します。ASA を通過するすべての H.245 メッセージは、H.245 アプリケーション インスペクションを受けます。このインスペクションでは、埋め込み IP アドレスが変換され、H.245 メッセージでネゴシエートされたメディア チャネルが開かれます。

H.323 ITU 規準では、メッセージ長を定義する TPKT ヘッダーが最初に送信されてから、H.225 と H.245 が信頼できる接続上を送信されることが要求されています。TPKT ヘッダーは、必ずしも H.225 メッセージや H.245 メッセージと同一の TCP パケットで送信される必要はないため、ASA は、メッセージを正しく処理して復号化するために TPKT 長を記憶しておく必要があります。ASA は、次のメッセージに備えて、TPKT 長が含まれるレコードを接続ごとに保持します。

ASA でメッセージ内の IP アドレスに NAT を行う必要がある場合、チェックサム、UUIE 長、および TPKT（H.225 メッセージが入っている TCP パケットに含まれている場合）は変更されます。TPKT が別の TCP パケットで送信される場合、ASA がその TPKT へのプロキシ ACK を実行し、新しい TPKT を新しい長さで H.245 メッセージに追加します。

（注） ASA は、TPKT に対する ACK の代理処理では TCP オプションをサポートしていません。

H.323 インスペクションを通過するパケットが通る各 UDP 接続は、H.323 接続としてマークされ、**timeout** コマンドで設定された H.323 タイムアウト値でタイムアウトします。

（注） ゲートキーパーがネットワーク内にある場合は、H.323 エンドポイント間のコール セットアップをイネーブルにできます。ASA には、RegistrationRequest/RegistrationConfirm（RRQ/RCF）メッセージに基づいてコールのピンホールを開くオプションが含まれています。これらの RRQ/RCF メッセージは ゲートキーパーとの間で送受信されるので、発信側エンドポイントの IP アドレスは不明で、ASA は発信元 IP アドレス/ポート 0/0 を通じてピンホールを開きます。デフォルトでは、このオプションは無効になっています。H.323 エンドポイント間のコール セットアップをイネーブルにするには、H.323 インスペクション ポリシー マップの作成時に、パラメータ コンフィギュレーション モードで **ras-rcf-pinholes enable** コマンドを入力します。「H.323 インスペクション ポリシー マップの設定」（P.9-7）を参照してください。

H.245 メッセージでの H.239 サポート

ASA は、2 つの H.323 エンドポイントの間に存在します。2 つの H.323 エンドポイントが、スプレッドシート データなどのデータ プレゼンテーションを送受信できるようにテレプレゼンテーション セッションをセットアップするとき、ASA はエンドポイント間で H.239 ネゴシエーションが成功することを保証します。

H.239 は、H.300 シリーズ エンドポイントが 1 回のコールで追加ビデオ チャネルを開くことができる機能を提供する規格です。コールで、エンドポイント（ビデオ電話など）はビデオ用チャネルとデータ プレゼンテーション用チャネルを送信します。H.239 ネゴシエーションは H.245 チャネルで発生します。

ASA が追加メディア チャネル用とメディア制御チャネル用のピンホールを開きます。エンドポイントは、オープン論理チャネル メッセージ（OLC）を使用して新しいチャネルの作成を通知します。メッセージ拡張は H.245 バージョン 13 の一部です。

テレプレゼンテーション セッションの復号化と符号化は、デフォルトでイネーブルにされています。H.239 の符号化と復号化は ASN.1 コーダによって実行されます。

H.323 インスペクションの制限事項

H.323 インスペクションは、Cisco Unified Communications Manager (CUCM) 7.0 でテストおよびサポートされています。CUCM 8.0 以降ではサポートされません。H.323 インスペクションは、他のリリースや製品で機能する場合があります。

H.323 アプリケーション インスペクションの使用に関して、次の既知の問題および制限があります。

- 完全にサポートされているのは、スタティック NAT だけです。スタティック PAT は、H.323 メッセージのオプション フィールドに埋め込まれた IP アドレスを正しく変換できないことがあります。この問題が発生した場合は、H.323 でスタティック PAT を使用しないでください。
- ダイナミック NAT または PAT ではサポートされません。
- 拡張 PAT ではサポートされません。
- セキュリティ レベルが同一のインターフェイス間の NAT ではサポートされません。
- 外部 NAT ではサポートされません。
- NAT64 ではサポートされません。
- NetMeeting クライアントが H.323 ゲートキーパーに登録し、同じく H.323 ゲートキーパーに登録されている H.323 ゲートウェイを呼び出そうとすると、接続は確立されますが、どちらの方向でも音声が聞こえません。この問題は、ASA の問題ではありません。
- ネットワーク スタティック アドレスを設定した場合、このネットワーク スタティック アドレスが第三者のネットマスクおよびアドレスと同じであると、すべての発信 H.323 接続が失敗します。

H.323 インスペクションの設定

H.323 インスペクションは RAS、H.225、H.245 をサポートし、埋め込まれた IP アドレスとポートをすべて変換する機能を備えています。ステートのトラッキングとフィルタリングを実行し、インスペクション機能のアクティベーションをカスケードできます。H.323 インスペクションは、電話番号のフィルタリング、T.120 のダイナミック制御、H.245 のトンネル機能制御、HSI グループ、プロトコルのステート トラッキング、H.323 通話時間制限の適用、音声/ビデオ制御をサポートします。

H.323 検査はデフォルトではイネーブルです。デフォルト以外の処理が必要な場合にのみ設定する必要があります。H.323 インスペクションをカスタマイズする場合は、次のプロセスを使用します。

手順

ステップ 1　「H.323 インスペクション ポリシー マップの設定」（P.9-7）

ステップ 2　「H.323 インスペクション サービス ポリシーの設定」（P.9-10）

H.323 インスペクション ポリシー マップの設定

ネットワークに対してデフォルトのインスペクション動作が十分でない場合は、H.323 インスペクション ポリシー マップを作成して H.323 インスペクションのアクションをカスタマイズできます。

トラフィックの一致基準を定義するときに、クラス マップを作成するか、またはポリシー マップに match ステートメントを直接含めることができます。次の手順では、両方の方法について説明します。

はじめる前に

一部のトラフィック照合オプションでは、照合のために正規表現を使用します。これらのテクニックの 1 つを使用する場合は、最初に正規表現または正規表現のクラス マップを作成します。

手順

ステップ 1　（任意）次の手順に従って、H.323 インスペクションのクラス マップを作成します。

クラス マップは複数のトラフィック照合をグループ化します。代わりに、ポリシー マップで **match** コマンドを直接指定できます。クラス マップを作成することとインスペクション ポリシー マップでトラフィックとの照合を直接定義することの違いは、クラス マップでは複雑な照合基準を作成でき、クラス マップを再利用できるということです。

クラス マップと照合しないトラフィックを指定するには、**match not** コマンドを使用します。たとえば、**match not** コマンドで文字列「example.com」を指定すると、「example.com」が含まれるすべてのトラフィックはクラス マップと照合されません。

このクラス マップで指定するトラフィックに対しては、インスペクション ポリシー マップでトラフィックに対して実行するアクションを指定します。

match コマンドごとに異なるアクションを実行する場合、ポリシー マップに直接トラフィックを特定する必要があります。

a. 次のコマンドを入力して、クラス マップを作成します。

```
hostname(config)# class-map type inspect h323 [match-all | match-any] class_map_name
hostname(config-cmap)#
```

class_map_name には、クラス マップの名前を指定します。**match-all** キーワードはデフォルトです。トラフィックがクラス マップと一致するには、すべての基準と一致する必要があることを指定します。**match-any** キーワードは、トラフィックが少なくとも基準の 1 つに一致したらクラス マップと一致することを指定します。CLI がクラスマップ コンフィギュレーション モードに入り、1 つ以上の **match** コマンドを入力できます。

b. （任意）クラス マップに説明を追加するには、次のコマンドを使用します。

```
hostname(config-cmap)# description string
```

string には、クラス マップの説明を 200 文字以内で指定します。

c. 次のいずれかの **match** コマンドを使用して、アクションを実行するトラフィックを指定します。**match not** コマンドを使用すると、**match not** コマンドの基準に一致しないすべてのトラフィックにアクションが適用されます。

- **match** [**not**] **called-party regex** {*regex_name* | **class** *class_name*}：指定した正規表現または正規表現クラスに対して着信側を照合します。
- **match** [**not**] **calling-party regex** {*regex_name* | **class** *class_name*}：指定した正規表現または正規表現クラスに対して発信側を照合します。
- **match** [**not**] **media-type** {**audio** | **data** | **video**}：メディア タイプを照合します。

ステップ 2　H.323 インスペクション ポリシー マップを作成します。

```
hostname(config)# policy-map type inspect h323 policy_map_name
hostname(config-pmap)#
```

policy_map_name には、ポリシー マップの名前を指定します。CLI はポリシー マップ コンフィギュレーション モードに入ります。

ステップ 3　（任意）このポリシー マップに説明を追加するには、次のコマンドを使用します。

```
hostname(config-pmap)# description string
```

ステップ 4　一致したトラフィックにアクションを適用するには、次の手順を実行します。

ポリシー マップには、複数の **class** コマンドまたは **match** コマンドを指定できます。**class** コマンドと **match** コマンドの順序については、「インスペクション ポリシー マップのアクションの定義」（P.2-4）を参照してください。

a.　次のいずれかの方法を使用して、アクションを実行するトラフィックを指定します。

- H.323 クラス マップを作成した場合は、次のコマンドを入力してそれを指定します。

  ```
  hostname(config-pmap)# class class_map_name
  hostname(config-pmap-c)#
  ```

- H.323 クラス マップで記述された **match** コマンドの 1 つを使用して、ポリシー マップでトラフィックを直接指定します。**match not** コマンドを使用すると、**match not** コマンドの基準に一致しないすべてのトラフィックにアクションが適用されます。

b.　次のコマンドを入力して、一致したトラフィックに対して実行するアクションを指定します。

```
hostname(config-pmap-c)# {drop [log] | drop-connection | reset}
```

drop キーワードはパケットをドロップします。メディア タイプの照合の場合、**log** キーワードを含めてシステム ログ メッセージを送信できます。

drop-connection キーワードを指定すると、パケットをドロップし、接続を閉じます。このオプションは、着信側または発信側の照合に使用できます。

reset キーワードを指定すると、パケットをドロップして接続を閉じ、サーバとクライアントの両方またはいずれかに TCP リセットを送信します。このオプションは、着信側または発信側の照合に使用できます。

ステップ 5　インスペクション エンジンに影響のあるパラメータを設定するには、次の手順を実行します。

a.　パラメータ コンフィギュレーション モードに入るには、次のコマンドを入力します。

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

b.　1 つまたは複数のパラメータを設定します。次のオプションを設定できます。オプションをディセーブルにするには、コマンドの **no** 形式を使用してください。

- **ras-rcf-pinholes enable**：H.323 エンドポイント間のコール セットアップをイネーブルにします。ゲートキーパーがネットワーク内にある場合は、H.323 エンドポイント間のコール セットアップをイネーブルにできます。RegistrationRequest/RegistrationConfirm（RRQ/RCF）メッセージに基づいてコールのピンホールを開くには、このオプションを使用します。これらの RRQ/RCF メッセージは ゲートキーパーとの間で送受信されるので、発信側エンドポイントの IP アドレスは不明で、ASA は発信元 IP アドレス/ポート 0/0 を通じてピンホールを開きます。デフォルトでは、このオプションは無効になっています。

- **timeout users** *time*：H.323 コールの制限時間（hh: mm: ss 形式）を設定します。タイムアウトを付けない場合は、00:00:00 を指定してください。範囲は、0:0:0 ～ 1193:0:0 です。
- **call-party-number**：コール設定時に発信側の番号を強制的に送信します。
- **h245-tunnel-block action** {**drop-connection** | **log**}：H.245 トンネル ブロッキングを適用します。接続をドロップするか、単にログに記録するだけかを選択します。
- **rtp-conformance** [**enforce-payloadtype**]：ピンホール上を流れる RTP パケットのプロトコル準拠をチェックします。オプションの **enforce-payloadtype** キーワードを指定すると、シグナリング交換に基づいてペイロード タイプを強制的に音声やビデオにします。
- **state-checking** {**h225** | **ras**}：ステート チェック検証をイネーブルにします。個別にコマンドを入力して、H.225 および RAS のステート チェックをイネーブルにすることができます。

ステップ 6　パラメータ コンフィギュレーション モードのままで、HSI グループを設定できます。

a. HSI グループを定義し、HSI グループ コンフィギュレーション モードを開始します。

```
hostname(config-pmap-p)# hsi-group id
```

id には、HSI グループ ID を指定します。範囲は 0 ～ 2147483647 です。

b. IP アドレスを使用して HSI を HSI グループに追加します。HSI グループあたり最大 5 つのホストを追加できます。

```
hostname(config-h225-map-hsi-grp)# hsi ip_address
```

c. HSI グループにエンドポイントを追加します。

```
hostname(config-h225-map-hsi-grp)# endpoint ip_address if_name
```

ip_address には追加するエンドポイント、*if_name* にはエンドポイントを ASA に接続するときに使用するインターフェイスを指定します。HSI グループあたり最大 10 個のエンドポイントを追加できます。

例

次の例は、電話番号のフィルタリングを設定する方法を示しています。

```
hostname(config)# regex caller 1 “5551234567”
hostname(config)# regex caller 2 “5552345678”
hostname(config)# regex caller 3 “5553456789”

hostname(config)# class-map type inspect h323 match-all h323_traffic
hostname(config-pmap-c)# match called-party regex caller1
hostname(config-pmap-c)# match calling-party regex caller2

hostname(config)# policy-map type inspect h323 h323_map
hostname(config-pmap)# parameters
hostname(config-pmap-p)# class h323_traffic
hostname(config-pmap-c)# drop
```

H.323 インスペクション サービス ポリシーの設定

デフォルトの ASA 設定には、すべてのインターフェイスでグローバルに適用されるデフォルト ポートでの H.323 H.255、および RAS のインスペクションが含まれます。インスペクションの設定をカスタマイズする一般的な方法は、デフォルトのグローバル ポリシーをカスタマイズすることです。または、たとえばインターフェイス固有のポリシーなど、必要に応じて新しいサービス ポリシーを作成することもできます。

手順

ステップ 1　必要な場合は、L3/L4 クラス マップを作成して、インスペクションを適用するトラフィックを識別します。

class-map *name*
match *parameter*

例：

```
hostname(config)# class-map h323_class_map
hostname(config-cmap)# match access-list h323
```

デフォルト グローバル ポリシーの inspection_default クラス マップは、すべてのインスペクション タイプのデフォルト ポートを含む特別なクラス マップです（**match default-inspection-traffic**）。このマップをデフォルト ポリシーまたは新しいサービス ポリシーで使用する場合は、このステップを省略できます。

照合ステートメントについては、「トラフィックの特定（レイヤ 3/4 クラス マップ）」（P.1-14）を参照してください。

ステップ 2　クラス マップ トラフィックで実行するアクションを設定するポリシー マップを追加または編集します。

policy-map *name*

例：

```
hostname(config)# policy-map global_policy
```

デフォルト設定では、global_policy ポリシー マップはすべてのインターフェイスにグローバルに割り当てられます。global_policy を編集する場合は、ポリシー名として global_policy を入力します。

ステップ 3　H.323 インスペクションに使用する L3/L4 クラス マップを指定します。

class *name*

例：

```
hostname(config-pmap)# class inspection_default
```

デフォルト ポリシーを編集する場合、または新しいポリシーで特別な inspection_default クラス マップを使用する場合は、*name* として **inspection_default** を指定します。それ以外の場合は、この手順ですでに作成したクラスを指定します。

ステップ 4　H.323 インスペクションを設定します。

inspect h323 {**h255** | **ras**} [*h323_policy_map*]

h323_policy_map は、オプションの H.323 インスペクション ポリシー マップです。デフォルト以外のインスペクション処理が必要な場合にのみマップが必要です。H.323 インスペクション ポリシー マップの作成の詳細については、「H.323 インスペクション ポリシー マップの設定」（P.9-7）を参照してください。

例：

```
hostname(config-class)# no inspect h323 h225
hostname(config-class)# no inspect h323 ras
hostname(config-class)# inspect h255 h323-map
hostname(config-class)# inspect ras h323-map
```

（注）　デフォルトのグローバル ポリシー（または使用中の任意のポリシー）を編集して、異なる H.323 インスペクション ポリシー マップを使用する場合は、**no inspect h323** コマンドで H.323 インスペクションを除去した後、新しい H.323 インスペクション ポリシー マップ名を指定して再度追加します。

ステップ 5　既存のサービス ポリシー（たとえば、global_policy という名前のデフォルト グローバル ポリシー）を編集している場合は、以上で終了です。それ以外の場合は、1 つまたは複数のインターフェイスでポリシー マップをアクティブにします。

service-policy *policymap_name* {**global** | **interface** *interface_name*}

例：

```
hostname(config)# service-policy global_policy global
```

global キーワードはポリシー マップをすべてのインターフェイスに適用し、**interface** は 1 つのインターフェイスに適用します。グローバル ポリシーは 1 つしか適用できません。インターフェイスのグローバル ポリシーは、そのインターフェイスにサービス ポリシーを適用することで上書きできます。各インターフェイスには、ポリシー マップを 1 つだけ適用できます。

H.323 および H.225 タイムアウト値の設定

[Configuration] > [Firewall] > [Advanced] > [Global Timeouts] ページで H.323/H.255 グローバル タイムアウト値を設定できます。H.255 シグナリング接続を閉じるまでの非アクティブ状態の間隔（デフォルトは 1 時間）または H.323 制御接続を閉じるまでの非アクティブ状態間隔（デフォルトは 5 分）を設定できます。

H.225 シグナリング接続を閉じるまでのアイドル時間を設定するには、**timeout h225** コマンドを使用します。H.225 タイムアウトのデフォルトは 1 時間です。

H.323 制御接続を閉じるまでのアイドル時間を設定するには、**timeout h323** コマンドを使用します。デフォルトは 5 分です。

H.323 インスペクションの確認とモニタリング

ここでは、H.323 セッションに関する情報を表示する方法について説明します。

- 「H.225 セッションのモニタリング」（P.9-12）
- 「H.245 セッションのモニタリング」（P.9-12）
- 「H.323 RAS セッションのモニタリング」（P.9-13）

H.225 セッションのモニタリング

show h225 コマンドは、ASA を越えて確立されている H.225 セッションの情報を表示します。このコマンドは、**debug h323 h225 event**、**debug h323 h245 event**、および **show local-host** コマンドとともに、H.323 インスペクション エンジンの問題のトラブルシューティングに使用されます。

異常なほど多くの接続が存在する場合は、デフォルトのタイムアウト値または設定した値に基づいてセッションがタイムアウトしているかどうか確認します。タイムアウトしていなければ問題があるので、調査が必要です。

次に、**show h225** コマンドの出力例を示します。

```
hostname# show h225
Total H.323 Calls: 1
1 Concurrent Call(s) for
    Local:   10.130.56.3/1040   Foreign: 172.30.254.203/1720
    1. CRV 9861
    Local:   10.130.56.3/1040   Foreign: 172.30.254.203/1720
0 Concurrent Call(s) for
    Local:   10.130.56.4/1050   Foreign: 172.30.254.205/1720
```

この出力は、現在 ASA を通過しているアクティブ H.323 コールが 1 つ、ローカル エンドポイント 10.130.56.3 と外部のホスト 172.30.254.203 の間にあることを示しています。また、これらの特定のエンドポイントの間に、同時コールが 1 つあり、そのコールの CRV が 9861 であることを示しています。

ローカル エンドポイント 10.130.56.4 と外部ホスト 172.30.254.205 に対して、同時コールは 0 です。つまり H.225 セッションがまだ存在しているものの、このエンドポイント間にはアクティブ コールがないことを意味します。この状況は、**show h225** コマンドを実行したときに、コールはすでに終了しているが、H.225 セッションがまだ削除されていない場合に発生する可能性があります。または、2 つのエンドポイントが、「maintainConnection」を TRUE に設定しているため、TCP 接続をまだ開いたままにしていることを意味する可能性もあります。したがって、「maintainConnection」を再度 FALSE に設定するまで、またはコンフィギュレーション内の H.225 タイムアウト値に基づくセッションのタイムアウトが起こるまで、セッションは開いたままになります。

H.245 セッションのモニタリング

show h245 コマンドは、スロー スタートを使用しているエンドポイントが ASA を越えて確立した H.245 セッションの情報を表示します。スロー スタートは、コールの 2 つのエンドポイントが H.245 用の別の TCP コントロール チャネルを開いた場合です。ファスト スタートは、H.245 メッセージが H.225 コントロール チャネルで H.225 メッセージの一部として交換された場合です。

次に、**show h245** コマンドの出力例を示します。

```
hostname# show h245
Total: 1
        LOCAL            TPKT    FOREIGN          TPKT
1       10.130.56.3/1041         0       172.30.254.203/1245    0
        MEDIA: LCN 258 Foreign 172.30.254.203 RTP 49608 RTCP 49609
                       Local   10.130.56.3 RTP 49608 RTCP 49609
        MEDIA: LCN 259 Foreign 172.30.254.203 RTP 49606 RTCP 49607
                       Local   10.130.56.3 RTP 49606 RTCP 49607
```

ASAでアクティブな H.245 コントロール セッションが、現在 1 つあります。ローカル エンドポイントは、10.130.56.3 であり、TPKT 値が 0 であることから、このエンドポイントからの次のパケットには TPKT ヘッダーがあると予測します。TKTP ヘッダーは、各 H.225/H.245 メッセージの前に送られる 4 バイトのヘッダーです。このヘッダーで、この 4 バイトのヘッダーを含むメッセージの長さがわかります。外部のホストのエンドポイントは、172.30.254.203 であり、TPKT 値が 0 であることから、このエンドポイントからの次のパケットには TPKT ヘッダーがあると予測します。

これらのエンドポイント間でネゴシエートされたメディアには、258 という LCN があり、外部に 172.30.254.203/49608 という RTP IP アドレス/ポート ペアと 172.30.254.203/49609 という RTCP IP アドレス/ポート ペアを持ち、ローカルに 10.130.56.3/49608 という RTP IP アドレス/ポート ペアと 49609 という RTCP ポートを持っています。

259 という 2 番目の LCN には、外部に 172.30.254.203/49606 という RTP IP アドレス/ポート ペアと 172.30.254.203/49607 という RTCP IP アドレス/ポート ペアがあり、ローカルに 10.130.56.3/49606 という RTP IP アドレス/ポート ペアと 49607 という RTCP ポートを持っています。

H.323 RAS セッションのモニタリング

show h323-ras コマンドは、ASA を越えてゲートキーパーとその H.323 エンドポイントの間に確立されている H.323 RAS セッションの接続情報を表示します。このコマンドは、**debug h323 ras event** および **show local-host** コマンドとともに、H.323 RAS インスペクション エンジンの問題のトラブルシューティングに使用されます。

次に、**show h323-ras** コマンドの出力例を示します。

```
hostname# show h323-ras
Total: 1
        GK                      Caller
        172.30.254.214 10.130.56.14
```

この出力は、ゲートキーパー 172.30.254.214 とそのクライアント 10.130.56.14 の間にアクティブな登録が 1 つあることを示しています。

MGCP インスペクション

ここでは、MGCP アプリケーション インスペクションについて説明します。

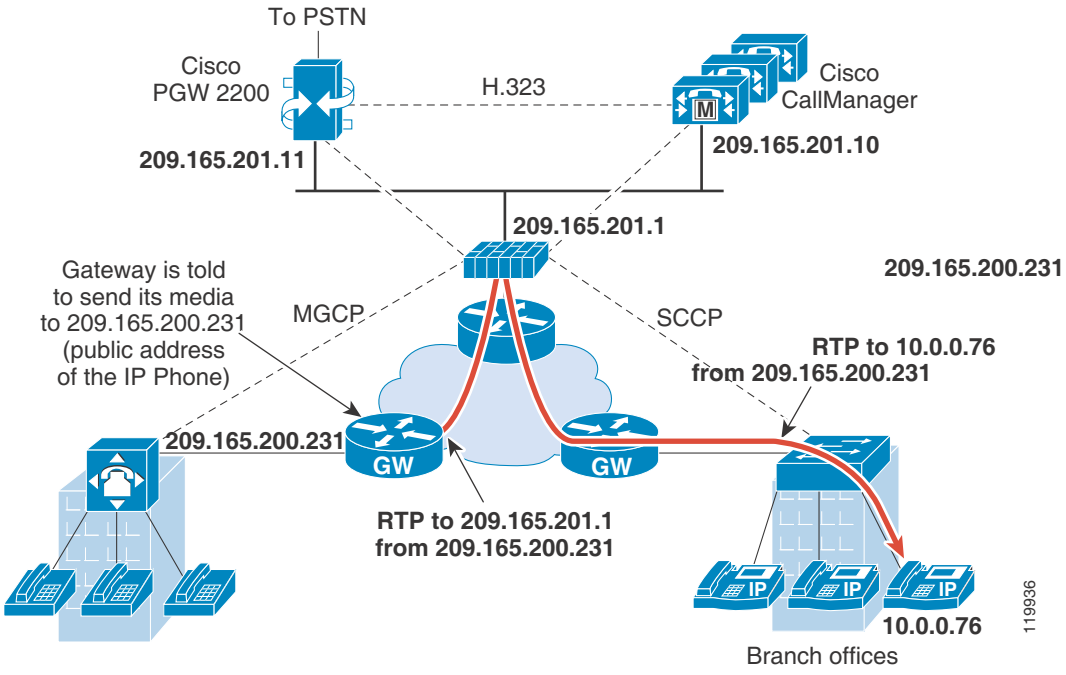
MGCP インスペクションの概要

MGCP は、メディア ゲートウェイ コントローラまたはコール エージェントと呼ばれる外部のコール制御要素からメディア ゲートウェイを制御するために使用するマスター /スレーブ プロトコルです。メディア ゲートウェイは一般に、電話回線を通じた音声信号と、インターネットまたは他のパケット ネットワークを通じたデータ パケットとの間の変換を行うネットワーク要素です。NAT および PAT を MGCP とともに使用すると、限られた外部（グローバル）アドレスのセットで、内部ネットワークの多数のデバイスをサポートできます。メディア ゲートウェイの例は次のとおりです。

- トランキング ゲートウェイ。電話ネットワークと Voice over IP ネットワークとの間のインターフェイスです。このようなゲートウェイは通常、大量のデジタル回線を管理します。
- 住宅用ゲートウェイ。従来のアナログ（RJ11）インターフェイスを Voice over IP ネットワークに提供します。住宅用ゲートウェイの例としては、ケーブル モデムやケーブル セットトップ ボックス、xDSL デバイス、ブロードバンド ワイヤレス デバイスなどがあります。
- ビジネス ゲートウェイ。従来のデジタル PBX（構内交換機）インターフェイスまたは統合 soft PBX インターフェイスを Voice over IP ネットワークに提供します。

MGCP メッセージは UDP を介して送信されます。応答はコマンドの送信元アドレス（IP アドレスと UDP ポート番号）に返送されますが、コマンド送信先と同じアドレスからの応答は到達しない場合があります。これは、複数のコール エージェントがフェールオーバー コンフィギュレーションで使用されているときに、コマンドを受信したコール エージェントが制御をバックアップ コール エージェントに引き渡し、バックアップ コール エージェントが応答を送信する場合に起こる可能性があります。次の図は、NAT と MGCP を使用する方法を示しています。

図 9-1　　NAT と MGCP の使用



MGCP エンドポイントは、物理または仮想のデータ送信元および宛先です。メディア ゲートウェイには、他のマルチメディア エンドポイントとのメディア セッションを確立して制御するために、コール エージェントが接続を作成、変更、および削除できるエンドポイントが含まれています。また、コール エージェントは、特定のイベントを検出してシグナルを生成するようにエンドポイントに指示できます。エンドポイントは、サービス状態の変化を自動的にコール エージェントに伝達します。

- 通常、ゲートウェイは UDP ポート 2427 をリッスンしてコール エージェントからのコマンドを受信します。
- コール エージェントがゲートウェイからのコマンドを受信するポート。通常、コール エージェントは UDP ポート 2727 をリッスンしてゲートウェイからコマンドを受信します。

（注） MGCP インスペクションでは、MGCP シグナリングと RTP データで異なる IP アドレスを使用することはサポートされていません。一般的かつ推奨される方法は、ループバック IP アドレスや仮想 IP アドレスなどの復元力のある IP アドレスから RTP データを送信することです。ただし、ASA は、MGCP シグナリングと同じアドレスから RTP データを受信する必要があります。

MGCP インスペクションの設定

MGCP インスペクションをイネーブルにするには、次のプロセスを使用します。

手順

ステップ 1　「インスペクション制御を追加するための MGCP インスペクション ポリシー マップの設定」（P.9-15）。

ステップ 2　「MGCP インスペクション サービス ポリシーの設定」（P.9-16）。

インスペクション制御を追加するための MGCP インスペクション ポリシー マップの設定

ASA がピンホールを開く必要のあるコール エージェントとゲートウェイがネットワークに複数ある場合、MGCP マップを作成します。作成した MGCP マップは、MGCP インスペクションをイネーブルにすると適用できます。

手順

ステップ 1　MGCP インスペクション ポリシー マップを作成するには、次のコマンドを入力します。

```
hostname(config)# policy-map type inspect mgcp map_name
hostname(config-pmap)#
```

policy_map_name には、ポリシー マップの名前を指定します。CLI はポリシー マップ コンフィギュレーション モードに入ります。

ステップ 2　（任意）このポリシー マップに説明を追加するには、次のコマンドを使用します。

```
hostname(config-pmap)# description string
```

ステップ 3　パラメータ コンフィギュレーション モードを開始します。

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

ステップ 4　1 つまたは複数のパラメータを設定します。次のオプションを設定できます。オプションをディセーブルにするには、コマンドの **no** 形式を使用してください。

- **call-agent** *ip_address group_id*：1 つ以上のゲートウェイを管理できるコール エージェント グループを設定します。コール エージェントのグループ情報は、どのコール エージェントも応答を送信できるように、グループ内の（ゲートウェイがコマンドを送信する先以外の）コール エージェントに接続を開くために使用されます。同じ *group_id* を持つコール エージェントは、同じグループに属します。1 つのコール エージェントは複数のグループに所属できます。*group_id* オプションには、0 ～ 4294967295 の数字を指定します。*ip_address* オプションには、コール エージェントの IP アドレスを指定します。

  （注）　MGCP コール エージェントは、AUEP メッセージを送信して、MGCP エンドポイントが存在するかどうかを判定します。これによって、ASA を通過するフローが確立され、MGCP エンドポイントをコール エージェントに登録できます。

- **gateway** *ip_address group_id*：特定のゲートウェイを管理しているコール エージェントのグループを指定します。*ip_address* オプションを使用して、ゲートウェイの IP アドレスを指定します。*group_id* オプションには 0 ～ 4294967295 の数字を指定します。この数字は、ゲートウェイを管理しているコール エージェントの *group_id* に対応している必要があります。1 つのゲートウェイは 1 つのグループだけに所属できます。
- **command-queue** *command_limit*：MGCP コマンド キューで許容されるコマンドの最大数（1 ～ 2147483647）を設定します。デフォルトは 200 です。

例

次の例は、MGCP マップを定義する方法を示しています。

```
hostname(config)# policy-map type inspect mgcp sample_map
hostname(config-pmap)# parameters
hostname(config-pmap-p)# call-agent 10.10.11.5 101
hostname(config-pmap-p)# call-agent 10.10.11.6 101
hostname(config-pmap-p)# call-agent 10.10.11.7 102
hostname(config-pmap-p)# call-agent 10.10.11.8 102
hostname(config-pmap-p)# gateway 10.10.10.115 101
hostname(config-pmap-p)# gateway 10.10.10.116 102
hostname(config-pmap-p)# gateway 10.10.10.117 102
hostname(config-pmap-p)# command-queue 150
```

MGCP インスペクション サービス ポリシーの設定

MGCP インスペクションは、デフォルトのインスペクション ポリシーでイネーブルになっていないため、このインスペクションが必要な場合はイネーブルにする必要があります。ただし、デフォルトの inspect クラスにはデフォルトの MGCP ポートが含まれているので、デフォルトのグローバル インスペクション ポリシーを編集するだけで MGCP インスペクションを追加できます。または、たとえばインターフェイス固有のポリシーなど、必要に応じて新しいサービス ポリシーを作成することもできます。

手順

ステップ 1　必要な場合は、L3/L4 クラス マップを作成して、インスペクションを適用するトラフィックを識別します。

```
class-map name
match parameter
```

例：

```
hostname(config)# class-map mgcp_class_map
hostname(config-cmap)# match access-list mgcp
```

デフォルト グローバル ポリシーの inspection_default クラス マップは、すべてのインスペクション タイプのデフォルト ポートを含む特別なクラス マップです（**match default-inspection-traffic**）。このマップをデフォルト ポリシーまたは新しいサービス ポリシーで使用する場合は、このステップを省略できます。

照合ステートメントについては、「トラフィックの特定（レイヤ 3/4 クラス マップ）」（P.1-14）を参照してください。

ステップ 2　クラス マップ トラフィックで実行するアクションを設定するポリシー マップを追加または編集します。

```
policy-map name
```

例：

```
hostname(config)# policy-map global_policy
```

デフォルト設定では、global_policy ポリシー マップはすべてのインターフェイスにグローバルに割り当てられます。global_policy を編集する場合は、ポリシー名として global_policy を入力します。

ステップ 3　MGCP インスペクションに使用する L3/L4 クラス マップを指定します。

```
class name
```

例：

```
hostname(config-pmap)# class inspection_default
```

デフォルト ポリシーを編集する場合、または新しいポリシーで特別な inspection_default クラス マップを使用する場合は、*name* として **inspection_default** を指定します。それ以外の場合は、この手順ですでに作成したクラスを指定します。

ステップ 4　MGCP インスペクションを設定します。

```
inspect mgcp [mgcp_policy_map]
```

mgcp_policy_map は、オプションの MGCP インスペクション ポリシー マップです。MGCP インスペクション ポリシー マップの作成の詳細については、「インスペクション制御を追加するための MGCP インスペクション ポリシー マップの設定」（P.9-15）を参照してください。

例：

```
hostname(config-class)# no inspect mgcp
hostname(config-class)# inspect mgcp mgcp-map
```

（注）　デフォルトのグローバル ポリシー（または使用中の任意のポリシー）を編集して、異なる MGCP インスペクション ポリシー マップを使用する場合は、**no inspect mgcp** コマンドで MGCP インスペクションを除去した後、新しい MGCP インスペクション ポリシー マップ名を指定して再度追加します。

ステップ 5　既存のサービス ポリシー（たとえば、global_policy という名前のデフォルト グローバル ポリシー）を編集している場合は、以上で終了です。それ以外の場合は、1 つまたは複数のインターフェイスでポリシー マップをアクティブにします。

service-policy *policymap_name* {**global** | **interface** *interface_name*}

例：

```
hostname(config)# service-policy global_policy global
```

global キーワードはポリシー マップをすべてのインターフェイスに適用し、**interface** は 1 つのインターフェイスに適用します。グローバル ポリシーは 1 つしか適用できません。インターフェイスのグローバル ポリシーは、そのインターフェイスにサービス ポリシーを適用することで上書きできます。各インターフェイスには、ポリシー マップを 1 つだけ適用できます。

MGCP タイムアウト値の設定

[Configuration] > [Firewall] > [Advanced] > [Global Timeouts] ページで複数の MGCP グローバル タイムアウト値を設定できます。MGCP メディア接続を閉じるまでの非アクティブ状態の間隔を設定できます（デフォルトは 5 分）。PAT xlate のタイムアウトを設定することもできます（30 秒）。

timeout mgcp コマンドを使用して、MGCP メディア接続を閉じるまでの非アクティブ状態の間隔を設定できます。デフォルトは 5 分です。

timeout mgcp-pat コマンドを使用して、PAT xlate のタイムアウトを設定できます。MGCP にはキープアライブ メカニズムがないため、Cisco 以外の MGCP ゲートウェイ（コール エージェント）を使用すると、デフォルトのタイムアウト間隔（30 秒）の後で PAT xlate は切断されます。

MGCP インスペクションの確認とモニタリング

show mgcp commands コマンドは、コマンド キュー内の MGCP コマンド数を表示します。**show mgcp sessions** コマンドは、既存の MGCP セッション数を表示します。**detail** オプションは、各コマンド（またはセッション）に関する追加情報を出力に含めます。次に、**show mgcp commands** コマンドの出力例を示します。

```
hostname# show mgcp commands
1 in use, 1 most used, 200 maximum allowed
CRCX, gateway IP: host-pc-2, transaction ID: 2052, idle: 0:00:07
```

次に、**show mgcp detail** コマンドの出力例を示します。

```
hostname# show mgcp commands detail
1 in use, 1 most used, 200 maximum allowed
CRCX, idle: 0:00:10
        Gateway IP      host-pc-2
        Transaction ID  2052
        Endpoint name   aaln/1
```

```
        Call ID         9876543210abcdef
        Connection ID
        Media IP        192.168.5.7
        Media port      6058
```

次に、**show mgcp sessions** コマンドの出力例を示します。

```
hostname# show mgcp sessions
1 in use, 1 most used
Gateway IP host-pc-2, connection ID 6789af54c9, active 0:00:11
```

次に、**show mgcp sessions detail** コマンドの出力例を示します。

```
hostname# show mgcp sessions detail
1 in use, 1 most used
Session active 0:00:14
        Gateway IP      host-pc-2
        Call ID         9876543210abcdef
        Connection ID   6789af54c9
        Endpoint name   aaln/1
        Media lcl port  6166
        Media rmt IP    192.168.5.7
        Media rmt port  6058
```

RTSP インスペクション

ここでは、RTSP アプリケーション インスペクションについて説明します。

- 「RTSP インスペクションの概要」（P.9-19）
- 「RealPlayer 設定要件」（P.9-20）
- 「RSTP インスペクションの制限事項」（P.9-20）
- 「RTSP インスペクションの設定」（P.9-21）

RTSP インスペクションの概要

RTSP インスペクション エンジンを使用することにより、ASA は RTSP パケットを通過させることができます。RTSP は、RealAudio、RealNetworks、Apple QuickTime 4、RealPlayer、および Cisco IP/TV の各接続で使用されます。

（注）　Cisco IP/TV では、RTSP TCP ポート 554 および 8554 を使用します。

RTSP アプリケーションは、制御チャネルとしての TCP（例外的に UDP）とともに予約済みポート 554 を使用します。ASA は、RFC 2326 に準拠して、TCP だけをサポートします。この TCP 制御チャネルは、クライアント上で設定されているトランスポート モードに応じて、音声/ビデオ トラフィックの送信に使用されるデータ チャネルのネゴシエーションに使用されます。

サポートされている RDT トランスポートは、rtp/avp、rtp/avp/udp、x-real-rdt、x-real-rdt/udp、x-pn-tng/udp です。

ASA は、ステータス コード 200 の SETUP 応答メッセージを解析します。SETUP 応答メッセージが、着信方向に移動している場合、サーバは ASA との相対位置関係で外部に存在することになるため、サーバから着信する接続に対してダイナミック チャネルを開くことが必要になります。この応答メッセージが発信方向である場合、ASA は、ダイナミック チャネルを開く必要はありません。

RFC 2326 では、クライアント ポートとサーバ ポートが、SETUP 応答メッセージ内に含まれていることは必要でないため、ASA では、状態を維持し、SETUP メッセージ内のクライアント ポートを記憶します。QuickTime が、SETUP メッセージ内にクライアント ポートを設定すると、サーバは、サーバ ポートだけで応答します。

RTSP インスペクションは、PAT またはデュアル NAT をサポートしていません。また、ASA は、RTSP メッセージが HTTP メッセージ内に隠される HTTP クローキングを認識できません。

RealPlayer 設定要件

RealPlayer を使用するときは、転送モードを正しく設定することが重要です。ASA では、サーバからクライアントに、またはその逆に **access-list** コマンドを追加します。RealPlayer の場合、[Options] > [Preferences] > [Transport] > [RTSP] [Settings] をクリックして転送モードを変更します。

RealPlayer で TCP モードを使用する場合は、[Use TCP to Connect to Server] チェックボックスおよび [Attempt to use TCP for all content] チェックボックスをオンにします。ASA で、インスペクション エンジンを設定する必要はありません。

RealPlayer で UDP モードを使用する場合、[Use TCP to Connect to Server] および [Attempt to use UDP for static content] チェックボックスをオンにします。マルチキャストでの使用ができないライブ コンテンツについては、ASA で、**inspect rtsp** *port* コマンドを追加します。

RSTP インスペクションの制限事項

RSTP インスペクションには次の制限が適用されます。

- ASA は、マルチキャスト RTSP または UDP による RTSP メッセージをサポートしません。
- ASA には、RTSP メッセージが HTTP メッセージ内に隠されている HTTP クローキングを認識する機能はありません。
- 埋め込み IP アドレスが HTTP メッセージまたは RTSP メッセージの一部として SDP ファイル内に含まれているため、ASA は、RTSP メッセージに NAT を実行できません。パケットはフラグメント化できますが、ASA ではフラグメント化されたパケットに対して NAT を実行することはできません。
- Cisco IP/TV では、メッセージの SDP 部分に対して ASA が実行する変換の数は、Content Manager にあるプログラム リストの数に比例します（各プログラム リストには、少なくとも 6 個の埋め込み IP アドレスを含めることができます）。
- Apple QuickTime 4 または RealPlayer 用の NAT を設定できます。Cisco IP/TV は、ビューアと Content Manager が外部ネットワークにあり、サーバが内部ネットワークにあるときにだけ NAT を使用できます。

RTSP インスペクションの設定

RTSP インスペクションはデフォルトでイネーブルになっています。デフォルト以外の処理が必要な場合にのみ設定する必要があります。RTSP インスペクションをカスタマイズする場合は、次のプロセスを使用します。

手順

ステップ 1　「RTSP インスペクション ポリシー マップの設定」（P.9-21）

ステップ 2　「RTSP インスペクション サービス ポリシーの設定」（P.9-23）

RTSP インスペクション ポリシー マップの設定

ネットワークに対してデフォルトのインスペクション動作が十分でない場合は、RTSP インスペクション ポリシー マップを作成して RTSP インスペクションのアクションをカスタマイズできます。

トラフィックの一致基準を定義するときに、クラス マップを作成するか、またはポリシー マップに match ステートメントを直接含めることができます。次の手順では、両方の方法について説明します。

はじめる前に

一部のトラフィック照合オプションでは、照合のために正規表現を使用します。これらのテクニックの 1 つを使用する場合は、最初に正規表現または正規表現のクラス マップを作成します。

手順

ステップ 1　（任意）次の手順に従って、RTSP インスペクションのクラス マップを作成します。

クラス マップは複数のトラフィック照合をグループ化します。代わりに、ポリシー マップで **match** コマンドを直接指定できます。クラス マップを作成することとインスペクション ポリシー マップでトラフィックとの照合を直接定義することの違いは、クラス マップでは複雑な照合基準を作成でき、クラス マップを再利用できるということです。

クラス マップと照合しないトラフィックを指定するには、**match not** コマンドを使用します。たとえば、**match not** コマンドで文字列「example.com」を指定すると、「example.com」が含まれるすべてのトラフィックはクラス マップと照合されません。

このクラス マップで指定するトラフィックに対しては、インスペクション ポリシー マップでトラフィックに対して実行するアクションを指定します。

match コマンドごとに異なるアクションを実行する場合、ポリシー マップに直接トラフィックを特定する必要があります。

a. 次のコマンドを入力して、クラス マップを作成します。

```
hostname(config)# class-map type inspect rtsp [match-all | match-any] class_map_name
hostname(config-cmap)#
```

class_map_name には、クラス マップの名前を指定します。**match-all** キーワードはデフォルトです。トラフィックがクラス マップと一致するには、すべての基準と一致する必要があることを指定します。**match-any** キーワードは、トラフィックが少なくとも基準の 1 つに一致したらクラス マップと一致することを指定します。CLI がクラスマップ コンフィギュレーション モードに入り、1 つ以上の **match** コマンドを入力できます。

b. （任意）クラス マップに説明を追加するには、次のコマンドを使用します。

```
hostname(config-cmap)# description string
```

c. 次のいずれかの **match** コマンドを使用して、アクションを実行するトラフィックを指定します。**match not** コマンドを使用すると、**match not** コマンドの基準に一致しないすべてのトラフィックにアクションが適用されます。

- **match** [**not**] **request-method** *method*：RTSP 要求方式を照合します。要求方式は、announce、describe、get_parameter、options、pause、play、record、redirect、setup、set_parameter、teardown です。
- **match** [**not**] **url-filter regex** {*regex_name* | **class** *class_name*}：指定した正規表現または正規表現クラスに対して URL を照合します。

ステップ 2　RTSP インスペクション ポリシー マップを作成するには、次のコマンドを入力します。

```
hostname(config)# policy-map type inspect rtsp policy_map_name
hostname(config-pmap)#
```

policy_map_name には、ポリシー マップの名前を指定します。CLI はポリシー マップ コンフィギュレーション モードに入ります。

ステップ 3　（任意）このポリシー マップに説明を追加するには、次のコマンドを使用します。

```
hostname(config-pmap)# description string
```

ステップ 4　一致したトラフィックにアクションを適用するには、次の手順を実行します。

a. 次のいずれかの方法を使用して、アクションを実行するトラフィックを指定します。

- RTSP クラス マップを作成した場合は、次のコマンドを入力してそれを指定します。

```
hostname(config-pmap)# class class_map_name
hostname(config-pmap-c)#
```

- RTSP クラス マップで記述された **match** コマンドの 1 つかを使用して、ポリシー マップでトラフィックを直接指定します。**match not** コマンドを使用すると、**match not** コマンドの基準に一致しないすべてのトラフィックにアクションが適用されます。

b. 次のコマンドを入力して、一致したトラフィックに対して実行するアクションを指定します。

```
hostname(config-pmap-c)# {drop-connection [log] | log | rate-limit message_rate}
```

drop-connection キーワードを指定すると、パケットをドロップし、接続を閉じます。このオプションは、URL のマッチングに使用できます。

単独または **drop-connection** と一緒に使用できる **log** キーワードからシステム ログ メッセージが送信されます。

rate-limit *message_rate* 引数では、1 秒あたりのメッセージのレートを制限します。このオプションは、要求方式の照合に使用できます。

ポリシー マップには、複数の **class** コマンドまたは **match** コマンドを指定できます。**class** コマンドと **match** コマンドの順序については、「インスペクション ポリシー マップのアクションの定義」（P.2-4）を参照してください。

ステップ 5　インスペクション エンジンに影響のあるパラメータを設定するには、次の手順を実行します。

a. パラメータ コンフィギュレーション モードに入るには、次のコマンドを入力します。

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

b. 1 つまたは複数のパラメータを設定します。次のオプションを設定できます。オプションをディセーブルにするには、コマンドの **no** 形式を使用してください。

- **reserve-port-protect**：メディア ネゴシエーション中の予約ポートの使用を制限します。
- **url-length-limit** *bytes*：メッセージで使用できる URL の長さを 0 ～ 6000 バイトで設定します。

例

次の例は、RTSP インスペクション ポリシー マップを定義する方法を示しています。

```
hostname(config)# regex badurl1 www.url1.com/rtsp.avi
hostname(config)# regex badurl2 www.url2.com/rtsp.rm
hostname(config)# regex badurl3 www.url3.com/rtsp.asp

hostname(config)# class-map type regex match-any badurl-list
hostname(config-cmap)# match regex badurl1
hostname(config-cmap)# match regex badurl2
hostname(config-cmap)# match regex badurl3

hostname(config)# policy-map type inspect rtsp rtsp-filter-map
hostname(config-pmap)# match url-filter regex class badurl-list
hostname(config-pmap-p)# drop-connection

hostname(config)# class-map rtsp-traffic-class
hostname(config-cmap)# match default-inspection-traffic

hostname(config)# policy-map rtsp-traffic-policy
hostname(config-pmap)# class rtsp-traffic-class
hostname(config-pmap-c)# inspect rtsp rtsp-filter-map

hostname(config)# service-policy rtsp-traffic-policy global
```

RTSP インスペクション サービス ポリシーの設定

ASA のデフォルトの設定には、すべてのインターフェイスにグローバルに適用されるデフォルト ポートの RTSP インスペクションが含まれます。インスペクションの設定をカスタマイズする一般的な方法は、デフォルトのグローバル ポリシーをカスタマイズすることです。または、たとえばインターフェイス固有のポリシーなど、必要に応じて新しいサービス ポリシーを作成することもできます。

手順

ステップ 1　必要な場合は、L3/L4 クラス マップを作成して、インスペクションを適用するトラフィックを識別します。

```
class-map name
match parameter
```

例：

```
hostname(config)# class-map rtsp_class_map
hostname(config-cmap)# match access-list rtsp
```

デフォルト グローバル ポリシーの inspection_default クラス マップは、すべてのインスペクション タイプのデフォルト ポートを含む特別なクラス マップです（**match default-inspection-traffic**）。このマップをデフォルト ポリシーまたは新しいサービス ポリシーで使用する場合は、このステップを省略できます。

照合ステートメントについては、「トラフィックの特定（レイヤ 3/4 クラス マップ）」（P.1-14）を参照してください。

ステップ 2　クラス マップ トラフィックで実行するアクションを設定するポリシー マップを追加または編集します。

```
policy-map name
```

例：

```
hostname(config)# policy-map global_policy
```

デフォルト設定では、global_policy ポリシー マップはすべてのインターフェイスにグローバルに割り当てられます。global_policy を編集する場合は、ポリシー名として global_policy を入力します。

ステップ 3　RTSP インスペクションに使用する L3/L4 クラス マップを指定します。

```
class name
```

例：

```
hostname(config-pmap)# class inspection_default
```

デフォルト ポリシーを編集する場合、または新しいポリシーで特別な inspection_default クラス マップを使用する場合は、*name* として **inspection_default** を指定します。それ以外の場合は、この手順ですでに作成したクラスを指定します。

ステップ 4　RTSP インスペクションを設定します。

```
inspect rtsp [rtsp_policy_map]
```

rtsp_policy_map は、オプションの RTSP インスペクション ポリシー マップです。デフォルト以外のインスペクション処理が必要な場合にのみマップが必要です。RTSP インスペクション ポリシー マップの作成の詳細については、「RTSP インスペクション ポリシー マップの設定」（P.9-21）を参照してください。

例：

```
hostname(config-class)# no inspect rtsp
hostname(config-class)# inspect rtsp rtsp-map
```

（注）　デフォルトのグローバル ポリシー（または使用中の任意のポリシー）を編集して、異なる RTSP インスペクション ポリシー マップを使用する場合は、**no inspect rtsp** コマンドで RTSP インスペクションを除去した後、新しい RTSP インスペクション ポリシー マップ名を指定して再度追加します。

ステップ 5　既存のサービス ポリシー（たとえば、global_policy という名前のデフォルト グローバル ポリシー）を編集している場合は、以上で終了です。それ以外の場合は、1 つまたは複数のインターフェイスでポリシー マップをアクティブにします。

service-policy *policymap_name* {**global** | **interface** *interface_name*}

例：

```
hostname(config)# service-policy global_policy global
```

global キーワードはポリシー マップをすべてのインターフェイスに適用し、**interface** は 1 つのインターフェイスに適用します。グローバル ポリシーは 1 つしか適用できません。インターフェイスのグローバル ポリシーは、そのインターフェイスにサービス ポリシーを適用することで上書きできます。各インターフェイスには、ポリシー マップを 1 つだけ適用できます。

SIP インスペクション

SIP は、インターネット会議、テレフォニー、プレゼンス、イベント通知、およびインスタント メッセージングに広く使用されているプロトコルです。テキストベースの性質とその柔軟性により、SIP ネットワークは数多くのセキュリティ脅威にさらされます。

SIP アプリケーション インスペクションでは、メッセージ ヘッダーおよび本文のアドレス変換、ポートの動的なオープン、および基本的な健全性チェックが行われます。SIP メッセージの健全性を実現するアプリケーション セキュリティおよびプロトコルへの準拠と、SIP ベースの攻撃の検出もサポートされます。

SIP インスペクションはデフォルトでイネーブルになっています。SCCP インスペクションは、デフォルト以外の処理が必要な場合、または暗号化されたトラフィックのインスペクションをイネーブルにするために TLS プロキシを設定する場合にのみ設定する必要があります。ここでは、SIP インスペクションについてより詳細に説明します。

SIP インスペクションの概要

IETF で定義されている SIP により、特に 2 者間の音声会議などのコール処理セッションまたは「コール」が使用可能になります。SIP は、コール シグナリング用の SDP で動作します。SDP は、メディア ストリーム用のポートを指定します。SIP を使用することにより、ASA は SIP VoIP ゲートウェイおよび VoIP プロキシ サーバをサポートできます。SIP と SDP の定義は、次の RFC に記載されています。

- SIP：Session Initiation Protocol、RFC 3261
- SDP：Session Description Protocol、RFC 2327

ASA 経由の SIP コールをサポートする場合は、シグナリング メッセージは予約済みの宛先ポート（UDP/TCP 5060）経由で送信され、メディア ストリームはダイナミックに割り当てられるため、メディア接続アドレスのシグナリング メッセージ、メディア ポート、およびメディアの初期接続を検査する必要があります。また、SIP は、IP パケットのユーザデータ部分に IP アドレスを埋め込みます。ASA がサポートする SIP 要求 URI の最大長は 255 であることに注意してください。

SIP インスペクションの制限事項

SIP インスペクションは、埋め込まれた IP アドレスに NAT を適用します。ただし、送信元と宛先両方のアドレスを変換するように NAT を設定している場合、外部アドレス（「trying」応答メッセージの SIP ヘッダー内の「from」）は書き換えられません。そのため、宛先アドレスの変換を回避するように SIP トラフィックを使用している場合は、オブジェクト NAT を使用する必要があります。

PAT を SIP で使用する場合、次の制限事項が適用されます。

- ASA で保護されているネットワークの SIP プロキシにリモート エンドポイントを登録しようとすると、次のような一定の条件下で登録が失敗します。
  - PAT がリモート エンドポイント用に設定されている。
  - SIP レジストラ サーバが外部ネットワークにある。
  - エンドポイントからプロキシ サーバに送信された REGISTER メッセージの接続先フィールドにポートが設定されていない。
- SDP 部分の所有者/作成者フィールド（o=）の IP アドレスが接続フィールド（c=）の IP アドレスと異なるパケットを SIP デバイスが送信すると、o= フィールドの IP アドレスが正しく変換されない場合があります。これは、o= フィールドでポート値を提供しない SIP プロトコルの制限によるものです。
- PAT を使用する場合は、ポートを持たない内部 IP アドレスを含む SIP ヘッダー フィールドは変換されない可能性があるため、内部 IP アドレスが外部に漏れます。この漏出を避けるには、PAT の代わりに NAT を設定します。

SIP インスタント メッセージ

インスタント メッセージとは、ほぼリアルタイムにユーザ間でメッセージを転送することです。SIP は、Windows Messenger RTC Client バージョン 4.7.0105 を使用する Windows XP のチャット機能のみをサポートします。次の RFC で定義されているように、MESSAGE/INFO 方式および 202 Accept 応答を使用して IM をサポートします。

- Session Initiation Protocol（SIP）Specific Event Notification、RFC 3265
- Session Initiation Protocol（SIP）Extension for Instant Messaging、RFC 3428

MESSAGE/INFO 要求は、登録または加入の後、任意の時点で着信する可能性があります。たとえば、2 人のユーザはいつでもオンラインになる可能性がありますが、何時間もチャットをすることはありません。そのため、SIP インスペクション エンジンは、設定されている SIP タイムアウト値に従ってタイムアウトするピンホールを開きます。この値は、登録継続時間よりも 5 分以上長く設定する必要があります。登録継続時間は Contact Expires 値で定義し、通常 30 分です。

MESSAGE/INFO 要求は、通常、ポート 5060 以外の動的に割り当てられたポートを使用して送信されるため、SIP インスペクション エンジンを通過する必要があります。

（注）　チャット機能のみがサポートされています。ホワイトボード、ファイル転送、アプリケーション共有はサポートされていません。RTC Client 5.0 はサポートされていません。

SIP インスペクションは、テキストベースの SIP メッセージを変換し、メッセージの SDP 部分の内容長を再計算した後、パケット長とチェックサムを再計算します。また、エンドポイントが受信すべきアドレスまたはポートとして、SIP メッセージの SDP 部分に指定されたポートに対するメディア接続をダイナミックに開きます。

SIP インスペクションでは、SIP ペイロードから取得したインデックス CALL_ID/FROM/TO を持つデータベースが使用されます。これらのインデックスにより、コール、送信元、宛先が識別されます。このデータベースには、SDP のメディア情報フィールド内で見つかったメディア アドレスとメディア ポート、およびメディア タイプが格納されます。1 つのセッションに対して、複数のメディア アドレスとポートが存在することが可能です。ASA は、これらのメディア アドレス/ポートを使用して、2 つのエンドポイント間に RTP/RTCP 接続を開きます。

初期コール セットアップ（INVITE）メッセージでは、予約済みポート 5060 を使用する必要があります。ただし、後続のメッセージにはこのポート番号がない場合もあります。SIP インスペクション エンジンはシグナリング接続のピンホールを開き、それらの接続を SIP 接続としてマークします。これは、SIP アプリケーションに到達した変換対象のメッセージに対して行われます。

コールのセットアップ時に、SIP セッションは、着信側エンドポイントから応答メッセージでメディア アドレスとメディア ポートを受信し、着信側エンドポイントがどの RTP ポートで受信するかを知らされるまで「一時的な」状態にあります。1 分以内に、応答メッセージの受信に障害があった場合は、シグナリング接続は切断されます。

最終的なハンドシェイクが行われると、コール状態はアクティブに移行し、シグナリング接続は、BYE メッセージの受信まで継続されます。

内部エンドポイントが、外部エンドポイントに発呼した場合、メディア ホールが、外部インターフェイスに対して開き、内部エンドポイントから送信された INVITE メッセージで指定された内部エンドポイントのメディア アドレスとメディア ポートに、RTP/RTCP UDP パケットが流れることが許可されます。内部インターフェイスに対する要求外の RTP/RTCP UDP パケットは、ASA のコンフィギュレーションで特別に許可されない限り、ASA を通過できません。

デフォルトの SIP インスペクション

SIP インスペクションはデフォルトでイネーブルになっており、次を含むデフォルトのインスペクション ポリシー マップを使用します。

- SIP インスタント メッセージ（IM）の拡張機能：イネーブル
- SIP トラフィック以外の SIP ポート使用：許可
- サーバとエンドポイントの IP アドレスの非表示：ディセーブル
- ソフトウェアのバージョンと SIP 以外の URI をマスク：ディセーブル
- 1 以上の宛先ホップ カウントを保証：イネーブル
- RTP 準拠：適用強制しない
- SIP 準拠：ステート チェックとヘッダー検証を実行しない

暗号化されたトラフィックのインスペクションがイネーブルになっていないことにも注意してください。暗号化されたトラフィックを検査するには、TLS プロキシを設定する必要があります。

SIP インスペクションの設定

SIP アプリケーション インスペクションでは、メッセージ ヘッダーおよび本文のアドレス変換、ポートの動的なオープン、および基本的な健全性チェックが行われます。SIP メッセージの健全性を実現するアプリケーション セキュリティおよびプロトコルへの準拠と、SIP ベースの攻撃の検出もサポートされます。

SIP インスペクションはデフォルトでイネーブルになっています。SCCP インスペクションは、デフォルト以外の処理が必要な場合、または暗号化されたトラフィックのインスペクションをイネーブルにするために TLS プロキシを設定する場合にのみ設定する必要があります。SIP インスペクションをカスタマイズする場合は、次のプロセスを使用します。

手順

ステップ 1　「SIP インスペクション ポリシー マップの設定」（P.9-28）

ステップ 2　「SIP インスペクション サービス ポリシーの設定」（P.9-32）

SIP インスペクション ポリシー マップの設定

ネットワークに対してデフォルトのインスペクション動作が十分でない場合は、SIP インスペクション ポリシー マップを作成して SIP インスペクションのアクションをカスタマイズできます。

トラフィックの一致基準を定義するときに、クラス マップを作成するか、またはポリシー マップに match ステートメントを直接含めることができます。次の手順では、両方の方法について説明します。

はじめる前に

一部のトラフィック照合オプションでは、照合のために正規表現を使用します。これらのテクニックの 1 つを使用する場合は、最初に正規表現または正規表現のクラス マップを作成します。

手順

ステップ 1　（任意）次の手順に従って、SIP インスペクションのクラス マップを作成します。

クラス マップは複数のトラフィック照合をグループ化します。代わりに、ポリシー マップで **match** コマンドを直接指定できます。クラス マップを作成することとインスペクション ポリシー マップでトラフィックとの照合を直接定義することの違いは、クラス マップでは複雑な照合基準を作成でき、クラス マップを再利用できるということです。

クラス マップと照合しないトラフィックを指定するには、**match not** コマンドを使用します。たとえば、**match not** コマンドで文字列「example.com」を指定すると、「example.com」が含まれるすべてのトラフィックはクラス マップと照合されません。

このクラス マップで指定するトラフィックに対しては、インスペクション ポリシー マップでトラフィックに対して実行するアクションを指定します。

match コマンドごとに異なるアクションを実行する場合、ポリシー マップに直接トラフィックを特定する必要があります。

a. 次のコマンドを入力して、クラス マップを作成します。

```
hostname(config)# class-map type inspect sip [match-all | match-any] class_map_name
hostname(config-cmap)#
```

class_map_name には、クラス マップの名前を指定します。**match-all** キーワードはデフォルトです。トラフィックがクラス マップと一致するには、すべての基準と一致する必要があることを指定します。match-any キーワードは、トラフィックが少なくとも 1 つの **match** ステートメントと一致したらクラス マップと一致することを指定します。CLI がクラス マップ コンフィギュレーション モードに入り、1 つ以上の **match** コマンドを入力できます。

b. （任意）クラス マップに説明を追加するには、次のコマンドを使用します。

```
hostname(config-cmap)# description string
```

string には、クラス マップの説明を 200 文字以内で指定します。

c. 次のいずれかの **match** コマンドを使用して、アクションを実行するトラフィックを指定します。**match not** コマンドを使用すると、**match not** コマンドの基準に一致しないすべてのトラフィックにアクションが適用されます。

- **match** [**not**] **called-party regex** {*regex_name* | **class** *class_name*}：指定された正規表現または正規表現クラスに対して、To ヘッダーで指定された着信側を照合します。
- **match** [**not**] **calling-party regex** {*regex_name* | **class** *class_name*}：指定された正規表現または正規表現クラスに対して、From ヘッダーで指定された発信側を照合します。
- **match** [**not**] **content length gt** *bytes*：SIP ヘッダーのコンテンツの長さが指定されたバイト数（0 ～ 65536）を超えているメッセージを照合します。
- **match** [**not**] **content type** {**sdp** | **regex** {*regex_name* | **class** *class_name*}：コンテンツ タイプを SDP として、または指定された正規表現または正規表現クラスに対して照合します。
- **match** [**not**] **im-subscriber regex** {*regex_name* | **class** *class_name*}：指定された正規表現または正規表現クラスに対して SIP IM サブスクライバを照合します。
- **match** [**not**] **message-path regex** {*regex_name* | **class** *class_name*}：指定された正規表現または正規表現クラスに対して SIP via ヘッダーを照合します。
- **match** [**not**] **request-method** *method*：ack、bye、cancel、info、invite、message、notify、options、prack、refer、register、subscribe、unknown、update の SIP 要求方式を照合します。

- **match [not] third-party-registration regex** {*regex_name* | **class** *class_name*}：指定された正規表現または正規表現クラスに対してサードパーティ登録の要求者を照合します。
- **match [not] uri {sip | tel} length gt** *bytes*：指定された長さ（0 ～ 65536 バイト）を超えている、選択したタイプ（SIP または TEL）の SIP ヘッダーの URI を照合します。

d. クラス マップ コンフィギュレーション モードを終了するには、「**exit**」と入力します。

ステップ 2　SIP インスペクション ポリシー マップを作成するには、次のコマンドを入力します。

```
hostname(config)# policy-map type inspect sip policy_map_name
hostname(config-pmap)#
```

policy_map_name には、ポリシー マップの名前を指定します。CLI はポリシー マップ コンフィギュレーション モードに入ります。

ステップ 3　（任意）このポリシー マップに説明を追加するには、次のコマンドを使用します。

```
hostname(config-pmap)# description string
```

ステップ 4　一致したトラフィックにアクションを適用するには、次の手順を実行します。

a. 次のいずれかの方法を使用して、アクションを実行するトラフィックを指定します。

- SIP クラス マップを作成した場合は、次のコマンドを入力してそれを指定します。

```
hostname(config-pmap)# class class_map_name
hostname(config-pmap-c)#
```

- SIP クラス マップで記述された **match** コマンドの 1 つを使用して、ポリシー マップでトラフィックを直接指定します。**match not** コマンドを使用すると、**match not** コマンドの基準に一致しないすべてのトラフィックにアクションが適用されます。

b. 次のコマンドを入力して、一致したトラフィックに対して実行するアクションを指定します。

```
hostname(config-pmap-c)# {[drop | drop-connection | reset] [log] |
rate-limit message_rate}
```

各 **match** コマンドまたは **class** コマンドですべてのオプションを使用できるわけではありません。使用できる正確なオプションについては、CLI ヘルプまたはコマンド リファレンスを参照してください。

drop キーワードを指定すると、一致するすべてのパケットをドロップします。

drop-connection キーワードを指定すると、パケットをドロップし、接続を閉じます。

reset キーワードを指定すると、パケットをドロップして接続を閉じ、サーバとクライアントの両方またはいずれかに TCP リセットを送信します。

log キーワードを指定すると、システム ログ メッセージを送信します。このキーワードは単独で、または他のキーワードのいずれかと一緒に使用できます。

rate-limit *message_rate* 引数では、メッセージのレートを制限します。レート制限は、「invite」および「register」に一致する要求方式の場合にのみ使用できます。

ポリシー マップには、複数の **class** コマンドまたは **match** コマンドを指定できます。**class** コマンドと **match** コマンドの順序については、「インスペクション ポリシー マップのアクションの定義」（P.2-4）を参照してください。

ステップ 5　インスペクション エンジンに影響のあるパラメータを設定するには、次の手順を実行します。

a. パラメータ コンフィギュレーション モードに入るには、次のコマンドを入力します。

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

b. 1 つまたは複数のパラメータを設定します。次のオプションを設定できます。オプションをディセーブルにするには、コマンドの **no** 形式を使用してください。

- **im**：インスタント メッセージングをイネーブルにします。
- **ip-address-privacy**：IP アドレスのプライバシーをイネーブルにし、サーバとエンドポイントの IP アドレスを非表示にします。
- **max-forwards-validation action** {**drop** | **drop-connection** | **reset** | **log**} [**log**]：これにより、宛先に到達するまで 0 にすることができない Max-Forwards ヘッダーの値がチェックされます。また、不適合なトラフィックに対して実行するアクション（パケットのドロップ、接続のドロップ、リセット、またはログ）と、ロギングをイネーブルまたはディセーブルのどちらにするかを選択する必要もあります。
- **rtp-conformance** [**enforce-payloadtype**]：ピンホール上を流れる RTP パケットのプロトコル準拠をチェックします。オプションの **enforce-payloadtype** キーワードを指定すると、シグナリング交換に基づいてペイロード タイプを強制的に音声やビデオにします。
- **software-version action** {**mask** [**log**] | **log**}：Server および User-Agent（エンドポイント）ヘッダー フィールドを使用するソフトウェア バージョンを識別します。SIP メッセージのソフトウェア バージョンをマスクしてオプションでロギングするか、単にロギングのみ実行することができます。
- **state-checking action** {**drop** | **drop-connection** | **reset** | **log**} [**log**]：状態遷移チェックをイネーブルにします。また、不適合なトラフィックに対して実行するアクション（パケットのドロップ、接続のドロップ、リセット、またはログ）と、ロギングをイネーブルまたはディセーブルのどちらにするかを選択する必要もあります。
- **strict-header-validation action** {**drop** | **drop-connection** | **reset** | **log**} [**log**]：RFC 3261 に従って SIP メッセージのヘッダー フィールドの厳密な検証をイネーブルにします。また、不適合なトラフィックに対して実行するアクション（パケットのドロップ、接続のドロップ、リセット、またはログ）と、ロギングをイネーブルまたはディセーブルのどちらにするかを選択する必要もあります。
- **traffic-non-sip**：既知の SIP シグナリング ポートで SIP 以外のトラフィックを許可します。
- **uri-non-sip action** {**mask** [**log**] | **log**}：Alert-Info および Call-Info ヘッダー フィールドにある SIP 以外の URI を識別します。SIP メッセージの情報をマスクしてオプションでロギングするか、単にロギングのみ実行することができます。

例

次の例は、SIP を使用したインスタント メッセージをディセーブルにする方法を示しています。

```
hostname(config)# policy-map type inspect sip mymap
hostname(config-pmap)# parameters
hostname(config-pmap-p)# no im

hostname(config)# policy-map global_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect sip mymap

hostname(config)# service-policy global_policy global
```

SIP インスペクション サービス ポリシーの設定

ASA のデフォルトの設定には、すべてのインターフェイスにグローバルに適用されるデフォルト ポートの SIP インスペクションが含まれます。インスペクションの設定をカスタマイズする一般的な方法は、デフォルトのグローバル ポリシーをカスタマイズすることです。または、たとえばインターフェイス固有のポリシーなど、必要に応じて新しいサービス ポリシーを作成することもできます。

手順

ステップ 1　必要な場合は、L3/L4 クラス マップを作成して、インスペクションを適用するトラフィックを識別します。

class-map *name*
match *parameter*

例：

```
hostname(config)# class-map sip_class_map
hostname(config-cmap)# match access-list sip
```

デフォルト グローバル ポリシーの inspection_default クラス マップは、すべてのインスペクション タイプのデフォルト ポートを含む特別なクラス マップです（**match default-inspection-traffic**）。このマップをデフォルト ポリシーまたは新しいサービス ポリシーで使用する場合は、このステップを省略できます。

照合ステートメントについては、「トラフィックの特定（レイヤ 3/4 クラス マップ）」（P.1-14）を参照してください。

ステップ 2　クラス マップ トラフィックで実行するアクションを設定するポリシー マップを追加または編集します。

policy-map *name*

例：

```
hostname(config)# policy-map global_policy
```

デフォルト設定では、global_policy ポリシー マップはすべてのインターフェイスにグローバルに割り当てられます。global_policy を編集する場合は、ポリシー名として global_policy を入力します。

ステップ 3　SIP インスペクションに使用する L3/L4 クラス マップを指定します。

class *name*

例：

```
hostname(config-pmap)# class inspection_default
```

デフォルト ポリシーを編集する場合、または新しいポリシーで特別な inspection_default クラス マップを使用する場合は、*name* として **inspection_default** を指定します。それ以外の場合は、この手順ですでに作成したクラスを指定します。

ステップ 4　SIP インスペクションを設定します。

inspect sip [*sip_policy_map*] [**tls-proxy** *proxy_name*]

それぞれの説明は次のとおりです。

- *sip_policy_map* は、オプションの SIP インスペクション ポリシー マップです。デフォルト以外のインスペクション処理が必要な場合にのみマップが必要です。SIP インスペクション ポリシー マップの作成の詳細については、「SIP インスペクション ポリシー マップの設定」（P.9-28）を参照してください。
- **tls-proxy** *proxy_name* には、このインスペクションに使用する TLS プロキシを指定します。TLS プロキシは、暗号化されたトラフィックのインスペクションをイネーブルにする場合にのみ必要です。

例：

```
hostname(config-class)# no inspect sip
hostname(config-class)# inspect sip sip-map
```

（注）　デフォルトのグローバル ポリシー（または使用中の任意のポリシー）を編集して、異なる SIP インスペクション ポリシー マップを使用する場合は、**no inspect sip** コマンドで SIP インスペクションを除去した後、新しい SIP インスペクション ポリシー マップ名を指定して再度追加します。

ステップ 5　既存のサービス ポリシー（たとえば、global_policy という名前のデフォルト グローバル ポリシー）を編集している場合は、以上で終了です。それ以外の場合は、1 つまたは複数のインターフェイスでポリシー マップをアクティブにします。

service-policy *policymap_name* {**global** | **interface** *interface_name*}

例：

```
hostname(config)# service-policy global_policy global
```

global キーワードはポリシー マップをすべてのインターフェイスに適用し、**interface** は 1 つのインターフェイスに適用します。グローバル ポリシーは 1 つしか適用できません。インターフェイスのグローバル ポリシーは、そのインターフェイスにサービス ポリシーを適用することで上書きできます。各インターフェイスには、ポリシー マップを 1 つだけ適用できます。

SIP タイムアウト値の設定

メディア接続は、接続がアイドル状態になってから 2 分以内に切断されます。ただし、これは設定可能なタイムアウトであり、時間間隔は変更することが可能です。

[Configuration] > [Firewall] > [Advanced] > [Global Timeouts] ページで複数の SIP グローバル タイムアウト値を設定できます。

SIP 制御接続のタイムアウトを設定するには、次のコマンドを入力します。

```
hostname(config)# timeout sip hh:mm:ss
```

このコマンドは、SIP 制御接続を閉じるまでのアイドル タイムアウトを設定します。

SIP メディア接続のタイムアウトを設定するには、次のコマンドを入力します。

```
hostname(config)# timeout sip_media hh:mm:ss
```

このコマンドは、SIP メディア接続を閉じるまでのアイドル タイムアウトを設定します。

SIP インスペクションの確認とモニタリング

show sip コマンドは、ASA を越えて確立されている SIP セッションの情報を表示します。このコマンドは、**debug sip** および **show local-host** コマンドとともに、SIP インスペクション エンジンの問題のトラブルシューティングに使用されます。

次に、**show sip** コマンドの出力例を示します。

```
hostname# show sip
Total: 2
call-id c3943000-960ca-2e43-228f@10.130.56.44
    state Call init, idle 0:00:01
call-id c3943000-860ca-7e1f-11f7@10.130.56.45
    state Active, idle 0:00:06
```

この例は、ASA 上の 2 つのアクティブな SIP セッションを示しています（Total フィールドで示されているように）。各 call-id は、コールを表しています。

最初のセッションは call-id c3943000-960ca-2e43-228f@10.130.56.44 で、Call Init 状態にあります。これは、このセッションがまだコール設定中であることを示しています。コール セットアップは、コールへの最後の応答が受信されるまでは完了しません。たとえば、発信者はすでに INVITE を送信して、100 Response を受信した可能性がありますが、200 OK はまだ受信していません。したがって、コール セットアップはまだ完了していません。1xx で始まっていない応答メッセージは最後の応答と考えられます。このセッションは、1 秒間アイドル状態でした。

2 番めのセッションは Active 状態です。この状態ではコール設定が完了し、エンドポイントがメディアを交換しています。このセッションは、6 秒間アイドル状態でした。

Skinny（SCCP）検査

ここでは、SCCP アプリケーション インスペクションについて説明します。

SCCP インスペクションの概要

Skinny（SCCP）は、VoIP ネットワークで使用される簡易プロトコルです。SCCP を使用する Cisco IP Phone は、H.323 環境でも使用できます。Cisco CallManager と併用すると、SCCP クライアントは、H.323 準拠端末と同時使用できます。

ASA は、SCCP に対して PAT と NAT をサポートします。IP 電話で使用できるグローバル IP アドレスよりも IP 電話が多い場合は、PAT が必要です。Skinny アプリケーション インスペクションは、SCCP シグナリング パケットの NAT と PAT をサポートすることで、すべての SCCP シグナリング パケットとメディア パケットが ASA を通過できるようにします。

Cisco CallManager と Cisco IP Phones 間の通常のトラフィックは SCCP を使用しており、特別な設定をしなくても SCCP インスペクションによって処理されます。ASA は、TFTP サーバの場所を Cisco IP Phone とその他の DHCP クライアントに送信することで、DHCP オプション 150 および 66 もサポートします。Cisco IP Phone では、デフォルト ルートを設定する DHCP オプション 3 を要求に含めることもできます。

（注）　ASA は、SCCP プロトコル バージョン 22 以前が稼働している Cisco IP Phone からのトラフィックのインスペクションをサポートします。

Cisco IP Phone のサポート

Cisco CallManager が Cisco IP Phone と比べてセキュリティの高いインターフェイスにあるトポロジでは、NAT が Cisco CallManager の IP アドレスに必要な場合、マッピングは**スタティック**である必要があります。これは、Cisco IP Phone では Cisco CallManager の IP アドレスをコンフィギュレーションで明示的に指定する必要があるためです。スタティック アイデンティティ エントリを使用すると、セキュリティが高いインターフェイス上にある Cisco CallManager が Cisco IP Phone からの登録を受け付けるようにできます。

Cisco IP Phone では、TFTP サーバにアクセスして、Cisco CallManager サーバに接続するために必要な設定情報をダウンロードする必要があります。

TFTP サーバと比較して Cisco IP Phone の方がセキュリティの低いインターフェイス上にある場合は、ACL を使用して UDP ポート 69 の保護された TFTP サーバに接続する必要があります。TFTP サーバに対してはスタティック エントリが必要ですが、識別スタティック エントリにする必要はありません。NAT を使用する場合、識別スタティック エントリは同じ IP アドレスにマッピングされます。PAT を使用する場合は、同じ IP アドレスとポートにマッピングされます。

Cisco IP Phone が TFTP サーバおよび Cisco CallManager と比べてセキュリティの*高い*インターフェイス上にある場合、Cisco IP Phone が接続を開始できるようにするために、ACL やスタティック エントリは必要ありません。

SCCP インスペクションの制限事項

内部の Cisco CallManager のアドレスが NAT または PAT 用に別の IP アドレスかポートを設定している場合、ASA は現在のところ TFTP を経由して転送するファイルの内容に対して NAT または PAT をサポートしていないため、外部の Cisco IP Phone 用の登録は失敗します。ASA は TFTP メッセージの NAT をサポートし、TFTP ファイル用にピンホールを開きますが、ASA は電話の登録中に TFTP によって転送された Cisco IP Phone のコンフィギュレーション ファイルに埋め込まれた Cisco CallManager の IP アドレスとポートを変換することはできません。

（注）　ASA では、コール セットアップ中であるコール以外の SCCP コールのステートフル フェールオーバーはサポートされていません。

デフォルトのSCCP インスペクション

SCCP インスペクションは、次のデフォルト値を使用してデフォルトでイネーブルになっています。

- 登録：適用強制しない
- メッセージの最大 ID：0x181
- プレフィックスの長さの最小値：4
- メディア タイムアウト：00:05:00
- シグナリング タイムアウト：01:00:00
- RTP 準拠：適用強制しない

暗号化されたトラフィックのインスペクションがイネーブルになっていないことにも注意してください。暗号化されたトラフィックを検査するには、TLS プロキシを設定する必要があります。

SCCP（Skinny）インスペクションの設定

SCCP（Skinny）アプリケーション インスペクションでは、パケット データ、ピンホールの動的開放に埋め込まれている IP アドレスとポート番号を変換します。また、追加のプロトコル準拠チェックと基本的なステート トラッキングも行います。

SCCP インスペクションはデフォルトではイネーブルです。SCCP インスペクションは、デフォルト以外の処理が必要な場合、または暗号化されたトラフィックのインスペクションをイネーブルにするために TLS プロキシを設定する場合にのみ設定する必要があります。SCCP インスペクションをカスタマイズする場合は、次のプロセスを使用します。

手順

ステップ 1　「インスペクション制御を追加するための Skinny（SCCP）インスペクション ポリシー マップの設定」（P.9-36）。

ステップ 2　「SCCP インスペクション サービス ポリシーの設定」（P.9-38）。

インスペクション制御を追加するための Skinny（SCCP）インスペクション ポリシー マップの設定

メッセージがパラメータに違反したときのアクションを指定するには、SCCP インスペクション ポリシー マップを作成します。作成したインスペクション ポリシー マップは、SCCP インスペクションをイネーブルにすると適用できます。

手順

ステップ 1　SCCP インスペクション ポリシー マップを作成します。

```
hostname(config)# policy-map type inspect skinny policy_map_name
hostname(config-pmap)#
```

policy_map_name には、ポリシー マップの名前を指定します。CLI はポリシー マップ コンフィギュレーション モードに入ります。

ステップ 2　（任意）説明をポリシー マップに追加します。

```
hostname(config-pmap)# description string
```

ステップ 3　（任意）SCCP メッセージのステーション メッセージ ID フィールドに基づいてトラフィックをドロップします。

a. 0x0 ～ 0xffff の 16 進数のステーション メッセージ ID の値に基づいてトラフィックを識別します。**match [not] message-id** コマンドを使用して、単一の ID または ID の範囲を指定できます。**match not** コマンドを使用すると、**match not** コマンドの基準に一致しないすべてのトラフィックにアクションが適用されます。

```
hostname(config-pmap)# match message-id value
hostname(config-pmap)# match message-id range start_value end_value
```

例：

```
hostname(config-pmap)# match message-id 0x181

hostname(config-pmap)# match message-id range 0x200 0xffff
```

b. 一致したパケットに対して実行するアクションを指定します。パケットをドロップし、必要に応じてロギングできます。

```
hostname(config-pmap)# drop [log]
```

c. ドロップするすべてのメッセージ ID を指定するまで、このプロセスを繰り返します。

ステップ 4　インスペクション エンジンに影響するパラメータを設定します。

a. パラメータ コンフィギュレーション モードを開始します。

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

b. 1 つまたは複数のパラメータを設定します。次のオプションを設定できます。オプションをディセーブルにするには、コマンドの **no** 形式を使用してください。

- **enforce-registration**：コールを発信する前に強制的に登録を実行します。
- **message-ID max** *hex_value*：許可される最大 SCCP ステーション メッセージ ID を設定します。メッセージ ID は 16 進数で指定します。デフォルトの最大値は 0x181 です。
- **rtp-conformance [enforce-payloadtype]**：ピンホール上を流れる RTP パケットのプロトコル準拠をチェックします。オプションの **enforce-payloadtype** キーワードを指定すると、シグナリング交換に基づいてペイロード タイプを強制的に音声やビデオにします。
- **sccp-prefix-len {max | min}** *length*：許可される最大または最小の SCCP プレフィックスの長さを設定します。最小値と最大値の両方を設定するには、このコマンドを 2 回入力します。デフォルトの最小値は 4 で、デフォルトの最大値はありません。
- **timeout {media | signaling}** *time*：メディアおよびシグナリング接続のタイムアウトを設定します（hh: mm: ss 形式）。タイムアウトを設定しない場合は、番号に 0 を指定します。デフォルトのメディア タイムアウトは 5 分、デフォルトのシグナリング タイムアウトは 1 時間です。

例

次の例は、SCCP インスペクション ポリシー マップを定義する方法を示しています。

```
hostname(config)# policy-map type inspect skinny skinny-map
hostname(config-pmap)# parameters
hostname(config-pmap-p)# enforce-registration
hostname(config-pmap-p)# match message-id range 200 300
hostname(config-pmap-p)# drop log
hostname(config)# class-map inspection_default
hostname(config-cmap)# match default-inspection-traffic
hostname(config)# policy-map global_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect skinny skinny-map
hostname(config)# service-policy global_policy global
```

SCCP インスペクション サービス ポリシーの設定

ASA のデフォルトの設定には、すべてのインターフェイスにグローバルに適用されるデフォルト ポートの SCCP インスペクションが含まれます。インスペクションの設定をカスタマイズする一般的な方法は、デフォルトのグローバル ポリシーをカスタマイズすることです。または、たとえばインターフェイス固有のポリシーなど、必要に応じて新しいサービス ポリシーを作成することもできます。

手順

ステップ 1　必要な場合は、L3/L4 クラス マップを作成して、インスペクションを適用するトラフィックを識別します。

class-map *name*
match *parameter*

例：

```
hostname(config)# class-map sccp_class_map
hostname(config-cmap)# match access-list sccp
```

デフォルト グローバル ポリシーの inspection_default クラス マップは、すべてのインスペクション タイプのデフォルト ポートを含む特別なクラス マップです（**match default-inspection-traffic**）。このマップをデフォルト ポリシーまたは新しいサービス ポリシーで使用する場合は、このステップを省略できます。

照合ステートメントについては、「トラフィックの特定（レイヤ 3/4 クラス マップ）」（P.1-14）を参照してください。

ステップ 2　クラス マップ トラフィックで実行するアクションを設定するポリシー マップを追加または編集します。

policy-map *name*

例：

```
hostname(config)# policy-map global_policy
```

デフォルト設定では、global_policy ポリシー マップはすべてのインターフェイスにグローバルに割り当てられます。global_policy を編集する場合は、ポリシー名として global_policy を入力します。

ステップ 3　SCCP インスペクションに使用する L3/L4 クラス マップを指定します。

class *name*

例：

```
hostname(config-pmap)# class inspection_default
```

デフォルト ポリシーを編集する場合、または新しいポリシーで特別な inspection_default クラス マップを使用する場合は、*name* として **inspection_default** を指定します。それ以外の場合は、この手順ですでに作成したクラスを指定します。

ステップ 4　SCCP インスペクションを設定します。

inspect skinny [*sccp_policy_map*] [**tls-proxy** *proxy_name*]

それぞれの説明は次のとおりです。

- *sccp_policy_map* は、オプションの SCCP インスペクション ポリシー マップです。デフォルト以外のインスペクション処理が必要な場合にのみマップが必要です。SCCP インスペクション ポリシー マップの作成の詳細については、「インスペクション制御を追加するための Skinny（SCCP）インスペクション ポリシー マップの設定」（P.9-36）を参照してください。
- **tls-proxy** *proxy_name* には、このインスペクションに使用する TLS プロキシを指定します。TLS プロキシは、暗号化されたトラフィックのインスペクションをイネーブルにする場合にのみ必要です。

例：

```
hostname(config-class)# no inspect skinny
hostname(config-class)# inspect skinny sccp-map
```

（注）　デフォルトのグローバル ポリシー（または使用中の任意のポリシー）を編集して、異なる SCCP インスペクション ポリシー マップを使用する場合は、**no inspect skinny** コマンドで SCCP インスペクションを除去した後、新しい SCCP インスペクション ポリシー マップ名を指定して再度追加します。

ステップ 5　既存のサービス ポリシー（たとえば、global_policy という名前のデフォルト グローバル ポリシー）を編集している場合は、以上で終了です。それ以外の場合は、1 つまたは複数のインターフェイスでポリシー マップをアクティブにします。

service-policy *policymap_name* {**global** | **interface** *interface_name*}

例：

```
hostname(config)# service-policy global_policy global
```

global キーワードはポリシー マップをすべてのインターフェイスに適用し、**interface** は 1 つのインターフェイスに適用します。グローバル ポリシーは 1 つしか適用できません。インターフェイスのグローバル ポリシーは、そのインターフェイスにサービス ポリシーを適用することで上書きできます。各インターフェイスには、ポリシー マップを 1 つだけ適用できます。

SCCP インスペクションの確認およびモニタ

show skinny コマンドは、SCCP（Skinny）インスペクション エンジンの問題のトラブルシューティングに役立ちます。次の条件での **show skinny** コマンドの出力例を示します。ASA を越えて 2 つのアクティブな Skinny セッションがセットアップされています。最初の Skinny セッションは、ローカル アドレス 10.0.0.11 にある内部 Cisco IP Phone と 172.18.1.33 にある外部 Cisco CallManager の間に確立されています。TCP ポート 2000 は、CallManager です。2 番目の Skinny セッションは、ローカル アドレス 10.0.0.22 にある別の内部 Cisco IP Phone と同じ Cisco CallManager の間に確立されています。

```
hostname# show skinny
        LOCAL                   FOREIGN                 STATE
-----------------------------------------------------------------
1       10.0.0.11/52238         172.18.1.33/2000                1
  MEDIA 10.0.0.11/22948         172.18.1.22/20798
2       10.0.0.22/52232         172.18.1.33/2000                1
  MEDIA 10.0.0.22/20798         172.18.1.11/22948
```

この出力は、2 つの内部 Cisco IP Phone 間でコールが確立されていることを示します。最初と 2 番目の電話機の RTP リスン ポートは、それぞれ UDP 22948 と 20798 です。

次に、これらの Skinny 接続の **show xlate debug** コマンドの出力例を示します。

```
hostname# show xlate debug
2 in use, 2 most used
Flags:  D - DNS, d - dump, I - identity, i - inside, n - no random,
        r - portmap, s - static
NAT from inside:10.0.0.11 to outside:172.18.1.11 flags si idle 0:00:16 timeout 0:05:00
NAT from inside:10.0.0.22 to outside:172.18.1.22 flags si idle 0:00:14 timeout 0:05:00
```

音声とビデオのプロトコル インスペクションの履歴

機能名	リリース	機能情報
SIP、SCCP、および TLS プロキシでの IPv6 のサポート	9.3(1)	SIP、SCCP、および TLS プロキシ（SIP または SCCP を使用）を使用している場合、IPv6 トラフィックを検査できるようになりました。 変更されたコマンドはありません。