

モバイル ネットワークのインスペクション

次の項では、LTE などのモバイル ネットワークで使用されるプロトコルに対するアプリケーション インスペクションについて説明します。これらのインスペクションには、 キャリア ライセンスが必要です。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を参照してください。

- モバイル ネットワーク インスペクションの概要 （1 ページ）
- モバイル ネットワーク プロトコル インスペクションのライセンス （10 ページ）
- GTP インスペクションのデフォルト （11 ページ）
- モバイル ネットワーク インスペクションの設定 （12 ページ）
- モバイル ネットワーク インスペクションのモニタリング （47 ページ）
- モバイル ネットワーク インスペクションの履歴 （52 ページ）

モバイル ネットワーク インスペクションの概要

次の項では、LTE などのモバイル ネットワークで使用されるプロトコルに対応するインスペクションについて説明します。インスペクションに加えて SCTP トラフィックで利用できるサービスは他にもあります。

GTP インスペクションの概要

GPRS トンネリング プロトコルは、General Packet Radio Service（GPRS）トラフィック用に GSM、UMTS および LTE ネットワークで使用されます。GTP は、トンネル制御および管理プロトコルを提供します。このプロトコルによるトンネルの作成、変更、および削除により、モバイル ステーションに GPRS ネットワーク アクセスが提供されます。GTP は、ユーザー データ パケットの伝送にもトンネリング メカニズムを使用します。

サービス プロバイダー ネットワークは、GTP を使用して、エンドポイント間の GPRS バックボーンを介してマルチプロトコル パケットをトンネリングします。GTPv0-1 では、GTP は gateway GPRS support node（GGSN）と serving GPRS support node（SGSN）間のシグナリングの

ために使用されます。GTPv2 では、シグナリングは Packet Data Network Gateway（PGW）と Serving Gateway（SGW）および他のエンドポイント間で行われます。GGSN/PGW は、GPRS ワイヤレス データ ネットワークと他のネットワーク間のインターフェイスです。SGSN/SGW は、モビリティ、データ セッション管理、およびデータ圧縮を実行します。

ASA を使用して、不正なローミング パートナーに対する保護を行えます。デバイスをホームの GGSN/PGW エンドポイントと訪問した SGSN/SGW エンドポイント間に配置し、トラフィック上で GTP インスペクションを使用します。GTP インスペクションは、これらのエンドポイント間のトラフィックでのみ動作します。GTPv2 では、これは S5/S8 インターフェイスとして知られています。

GTP および関連する規格は、3GPP（第 3 世代パートナーシップ プロジェクト）によって定義されます。詳細については、http://www.3gpp.org を参照してください。

モバイル端末の場所変更の追跡

GTP インスペクションを使用すると、モバイル端末の場所の変更を追跡できます。場所の変更を追跡すると、不正なローミング請求を特定するのに役立つ場合があります。たとえば、モバイル端末が、米国のセルから欧州のセルに 30 分以内に移動するなど、ある場所から別の場所にありえない時間で移動した場合などです。

場所のロギングを有効にすると、システムは International Mobile Subscriber Identity（IMSI）ごとに新しい場所または変更された場所の syslog メッセージを生成します。

- 324010 は新しい PDP コンテキストの作成を示し、携帯電話の国コード（MCC）、モバイル ネットワーク コード（MNC）、情報要素、および必要に応じてユーザーが現在登録されているセル ID が含まれます。セル ID は、セル グローバル識別（CGI）または E-UTRAN セル グローバル識別子（ECGI）から抽出されます。
- 324011 は、IMSI が PDP コンテキストの作成中に保存されたものから移動したことを示します。メッセージには、以前および現在の MCC/MNC、情報要素、および必要に応じてセル ID が表示されます。

デフォルトでは、syslog メッセージにタイムスタンプ情報は含まれません。これらのメッセージを分析してありえないローミングを識別する場合は、タイムスタンプも有効にする必要があります。タイムスタンプ ロギングは GTP インスペクション マップに含まれません。**logging timestamp** コマンドを使用します。

場所のロギングの有効化に関する詳細については、GTP インスペクション ポリシー マップの設定 （12 ページ）を参照してください。

GTP インスペクションの制限事項

次に、GTP インスペクションに関する制限事項の一部を示します。

- GTPv2 ピギーバック メッセージはサポートされていません。これらは常にドロップされます。
- GTPv2 emergency UE attach は、IMSI（International Mobile Subscriber Identity）が含まれている場合にのみサポートされます。

- GTPインスペクションは初期のデータは検査しません。つまり、セッション要求の作成直後かつセッション応答の作成前に PGW または SGW から送信されたデータのことです。
- GTPv2 の場合、インスペクションは 3GPP 29.274 V15.5.0 までサポートされています。GTPv1 の場合、3GPP 29.060 V15.2.0 までサポートされています。GTPv0 の場合、リリース 8 までサポートしています。
- GTPインスペクションは、セカンダリ PDP コンテキストへの SGSN 間ハンドオフをサポートしていません。インスペクションは、プライマリおよびセカンダリ両方の PDP コンテキストに対しハンドオフを実行する必要があります。
- GTP インスペクションを有効にすると、GTP-in-GTP カプセル化を使用する接続は常にドロップされます。

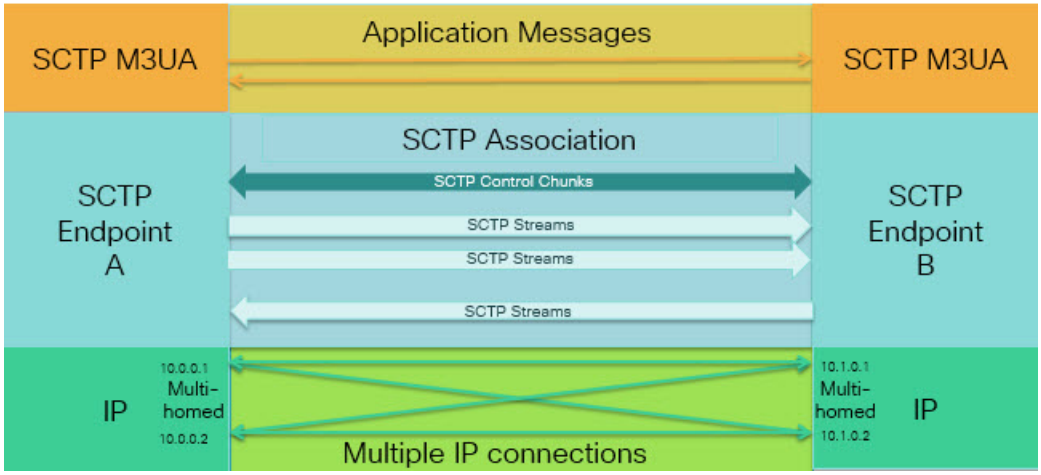
Stream Control Transmission Protocol（SCTP）インスペクションとアクセス制御

SCTP（Stream Control Transmission Protocol）は RFC 4960 で説明されています。プロトコルは IP 経由のテレフォニー シグナリング プロトコル SS7 をサポートしており、4G LTE モバイル ネットワーク アーキテクチャにおける複数のインターフェイス用の転送プロトコルでもあります。

SCTP は、TCP や UDP と同様、プロトコル スタックの IP の最上部で動作するトランスポート層プロトコルです。ただし、SCTP は、1 つ以上の送信元 IP アドレスまたは宛先 IP アドレス上の 2 つのエンド ノード間でアソシエーションと呼ばれる論理的な通信チャネルを作成します。これはマルチホーミングと呼ばれます。アソシエーションでは、各ノード（送信元と宛先）での IP アドレスのセットと、各ノードでのポートが定義されます。セット内の任意の IP アドレスは、複数の接続を形成するためにこのアソシエーションに関連付けられたデータ パケットの送信元または宛先 IP アドレスとして使用できます。各接続内では、メッセージを送信するために複数のストリームが存在する可能性があります。SCTP 内のストリームは、論理的なアプリケーション データ チャネルを表します。

次の図は、アソシエーションとそのストリームとの関係を示しています。

図 *1: SCTP* アソシエーションとストリームの関係



ASA を通過する SCTP トラフィックがある場合、SCTP ポートに基づいてアクセスを制御し、アプリケーション層のインスペクションを実行して、接続を有効にし、オプションでペイロードプロトコル ID でフィルタリングを行い、アプリケーションを選択的にドロップ、ログに記録、またはレート制限できます。

（注） 各ノードは、最大 3 つの IP アドレスを持つことができます。上限である 3 を超えたアドレスは無視され、アソシエーションに含まれません。セカンダリ IP アドレスのピンホールは、自動的に開きます。これらを許可するアクセス制御ルールを記述する必要はありません。

ハートビートと関連付けの状態

関連付けを維持し、正確なレポートを確保するために、インスペクションエンジンは SCTP ハートビートをモニタリングします。システムによって、接続時間がハートビート信号に結び付けられます。ハートビートが失われたか、関連付けが明示的に終了すると、デバイスは接続を閉じてイベント期間を終了します。

SCTP の関連付けとポートレポート

SCTP の検査は、フローではなく関連付けに基づいて行われます。デバイスは関連付け全体を単一のユニットとして追跡するため、個々のストリームポートはロギング出力での従来のポートフィールドにマッピングされません。そのため、接続の詳細を表示するときに、送信元ポートと宛先ポートが 0 と表示されることがあります。これは、関連付けが個々の送信元/宛先ポートのペアではなく、IP アドレスと関連付け ID によって一意に識別されるためです。

SCTP ステートフル インスペクション

TCP と同様、SCTP トラフィックは、正しく構造化されたトラフィックと RFC 4960 の限定的な適用についてレイヤ 4 で自動的に検査されます。次のプロトコル要素が検査され、適用されます。

- チャンクのタイプ、フラグ、および長さ。
- 検証タグ。
- 送信元ポートと宛先ポート。アソシエーション リダイレクト攻撃を防ぐため。
- IP アドレス。

SCTP ステートフル インスペクションは、アソシエーションの状態に基づいてパケットの受け入れまたは拒否を行います。

- 最初のアソシエーション確立のための 4 方向開閉シーケンスの検証。
- アソシエーションおよびストリーム内の TSN の転送進捗状況の確認。
- ハートビートの障害による中断チャンクを確認した場合のアソシエーションの終了。SCTP エンドポイントは、爆弾攻撃に応答して中断チャンクを送信する場合があります。

これらの強制チェックを行わない場合は、特定のトラフィック クラスの接続の設定（すべてのサービス） で説明されているように、特定のトラフィック クラスに対し SCTP ステート バイパスを設定できます。

SCTP アクセス制御

SCTP トラフィックのアクセス ルールを作成できます。これらのルールは TCP/UDP ポート ベースのルールと似ており、プロトコルとして単に **sctp** を使用し、ポート番号は SCTP ポートです。SCTP 用のサービス オブジェクトまたはグループを作成するか、またはポートを直接指定できます。次の項を参照してください。

- サービス オブジェクトとサービス グループの設定
- ポートベースの照合に使用する拡張 ACE の追加

SCTP NAT

SCTP アソシエーション確立メッセージのアドレスにスタティック ネットワーク オブジェクト NAT を適用できます。スタティック Twice NAT を設定できますが、SCTP アソシエーションの宛先部分のトポロジが不明であるため、これは推奨されません。ダイナミック NAT/PAT を使用することはできません。

SCTP 用の NAT は、SCTP アプリケーション レイヤのインスペクションではなく、SCTP ステートフル インスペクションによって決まります。したがって、SCTP ステート バイパスを設定している場合は、NAT トラフィックはできません。

SCTP アプリケーション レイヤのインスペクション

SCTP アプリケーション SCTP インスペクションとフィルタリングを有効にすることにより、アクセス ルールをさらに絞り込むことができます。ペイロード プロトコル ID（PPID）に基づいて、SCTP トラフィック クラスを選択的にドロップ、ログに記録、またはレート制限することができます。

PPID でフィルタリングする場合は、次の点に注意してください。

- PPID はデータのかたまりの中にあり、特定のパケットは複数のデータ チャンクまたは 1 つの制御チャンクを持つことができます。パケットに 1 つの制御チャンクまたは複数のデータ チャンクが含まれている場合、割り当てられたアクションがドロップされてもパケットはドロップされません。
- PPID フィルタリングを使用してパケットをドロップまたはレート制限する場合は、送信機によりドロップされたパケットが再送されることに注意してください。レート制限が適用された PPID のパケットは再試行で通過する可能性がありますが、ドロップされた PPID のパケットは再びドロップされます。ネットワーク上のこのような反復的ドロップの最終成果を評価することができます。

SCTP に関する制限事項

SCTP サポートには次の制限事項が含まれます。

- 各ノードは、最大 3 つの IP アドレスを持つことができます。上限である 3 を超えたアドレスは無視され、アソシエーションに含まれません。セカンダリ IP アドレスのピンホールは、自動的に開きます。これらを許可するアクセス制御ルールを記述する必要はありません。
- 使用されないピンホールは、5 分後にタイムアウトします。
- マルチホーム エンドポイントのデュアル スタック IPv4 および IPv6 アドレスはサポートされません。
- ネットワーク オブジェクト スタティック NAT は、唯一サポートされているタイプの NAT です。また、NAT46 および NAT64 はサポートされません。
- SCTP パケットのフラグメンテーションとリアセンブリは、Diameter、M3UA、および SCTP の PPID ベースのインスペクションで処理されたトラフィックにのみ実行されます。
- SCTP で IP アドレスを動的に追加または削除するために使用される ASCONF チャンクは、サポートされません。
- IP アドレスに解決できるホスト名を指定するために使用される、INIT および INIT-ACK SCTP メッセージ内のホスト名パラメータは、サポートされません。
- ASA、またはネットワーク内の他の場所で設定されているかどうかにかかわらず、SCTP/M3UA は等コスト マルチパス ルーティング（ECMP）をサポートしません。ECMP を使用すると、複数のベストパスを介してパケットを宛先にルーティングできます。ただし、単一の宛先への SCTP/M3UA パケット応答は、送出されたときと同じインターフェイスに戻る必要があります。応答が M3UA サーバーから送信される可能性があるとしても、常に送出されたときと同じインターフェイスに戻る必要があります。この問題の症状として、SCTP INIT-ACK パケットがドロップされます。これは、**show asp drop flow sctp-chunk-init-timeout** カウンタで確認できます。

```
Flow drop:
```

```
SCTP INIT timed out (not receiving INIT ACK)(sctp-chunk-init-timeout)
```

この問題が発生した場合は、M3UA サーバーへのスタティック ルートを設定するか、またはポリシーベース ルーティングを設定して、INIT-ACK パケットが INIT パケットと同じインターフェイスを確実に通過するネットワーク設計を実装することで解決できます。

Diameter インスペクション

Diameter は、LTE（Long Term Evolution）および IMS（IP Multimedia Subsystem）用の EPS（Evolved Packet System）などの次世代モバイルと固定電気通信ネットワークで使用される認証、認可、およびアカウンティング（AAA）プロトコルです。RADIUS や TACACS がこれらのネットワークで Diameter に置き換えられます。

Diameter はトランスポート層として TCP および SCTP を使用し、TCP/TLS および SCTP/DTLS によって通信を保護します。また、オプションで、データ オブジェクトの暗号化も提供できます。Diameter の詳細については、RFC 6733 を参照してください。

Diameterアプリケーションは、課金のユーザーアクセス、サービス認証、QoS、およびレートの決定といったサービス管理タスクを実行します。Diameter アプリケーションは LTE アーキテクチャのさまざまなコントロール プレーン インターフェイスで使用されますが、ASA は、次のインターフェイスについてのみ、Diameter コマンド コードおよび属性値ペア（AVP）を検査します。

- S6a：モビリティ マネージメント エンティティ（MME）-ホーム サブスクリプション サービス（HSS）
- S9：PDN ゲートウェイ（PDG）- 3GPP AAA プロキシ/サーバー
- Rx：ポリシー/課金ルール機能（PCRF） - コール セッション制御機能（CSCF）

Diameterインスペクションでは、Diameterエンドポイント用にピンホールを開いて通信を可能にします。このインスペクションは、3GPP バージョン 12 をサポートし、RFC 6733 に準拠しています。TCP/TLS（インスペクションをイネーブルにするときに TLS を指定する場合）および SCTP には使用できますが、SCTP/DTLS には使用できません。SCTP Diameter セッションにセキュリティを提供するには IPsec を使用します。

パケットや接続のドロップまたはロギングなどの特別なアクションを適用するために、オプションで、Diameter インスペクション ポリシー マップを使用し、アプリケーション ID、コマンド コード、および AVP に基づいてトラフィックをフィルタリングできます。新規に登録された Diameter アプリケーション用のカスタム AVP を作成できます。フィルタリングにより、ネットワークで許可するトラフィックを微調整できます。

（注） 他のインターフェイス上で動作するアプリケーションに対するDiameterメッセージはデフォルトで許可され、渡されます。ただし、アプリケーション ID によってこれらのアプリケーションを破棄するための Diameter インスペクション ポリシー マップを設定できますが、これらのサポートされていないアプリケーションに対してコマンド コードまたは AVP に基づいてアクションを指定することはできません。

M3UA インスペクション

MTP3 User Adaptation（M3UA）は、SS7 Message Transfer Part 3（MTP3）レイヤと連動する IP ベース アプリケーション用のSS7ネットワークへのゲートウェイを提供するクライアント/サーバー プロトコルです。M3UA により、IP ネットワーク上で SS7 ユーザー パート（ISUP など）を実行することが可能になります。M3UA は RFC 4666 で定義されています。

M3UA は SCTP をトランスポート層として使用します。SCTP ポート 2905 がデフォルト ポートです。

MTP3 レイヤは、ルーティングおよびノードアドレッシングなどのネットワーク機能を提供しますが、ノードの識別にポイント コードを使用します。M3UA 層は、発信ポイント コード（OPC）および宛先ポイント コード（DPC）を交換します。これは、IP が IP アドレスを使用してノードを識別する仕組みと似ています。

M3UAインスペクションは、限定されたプロトコル準拠を提供します。オプションで、厳密なアプリケーション サーバー プロセス（ASP）のステート チェックおよび選択されたメッセージの追加のメッセージの検証を実装できます。厳密なASPのステートチェックが必要なのは、ステートフル フェールオーバーが必要な場合、またはクラスタ内での動作が必要な場合です。ただし、厳密な ASP のステート チェックは、上書きモードでのみ動作し、ロードシェアリングまたはブロードキャスト モードで実行している場合は動作しません（RFC 4666 より）。インスペクションは、エンドポイントごとに ASP が 1 つだけあると仮定します。

オプションで、ポイント コードまたはサービス インジケータ（SI）に基づいてアクセス ポリシーを適用できます。また、メッセージのクラスおよびタイプに基づいてレート制限を適用できます。

M3UA プロトコル準拠

M3UAインスペクションでは、次の限定されたプロトコルを強制できます。インスペクションは、要件を満たさないパケットをドロップしてログに記録します。

- 共通のメッセージヘッダー。インスペクションでは、共通ヘッダー内のすべてのフィールドを確認します。
  - バージョン 1 のみ。
  - メッセージの長さが正しく設定されている必要があります。
  - 予約済みの値を使用したメッセージ タイプのクラスは許可されません。

- メッセージ クラス内での無効なメッセージ ID は許可されません。

- ペイロード データ メッセージ。
  - 特定のタイプの 1 つのパラメータのみが許可されます。
  - SCTP ストリーム 0 でのデータ メッセージは許可されません。

- [Affected Point Code] フィールドは次のメッセージに含まれている必要があり、含まれていない場合、メッセージはドロップされます。利用可能な宛先（DAVA）、利用できない宛先（DUNA）、宛先の状態監査（DAUD）、シグナリング輻輳（SCON）、利用できない宛先ユーザー部（DUPU）、制限された宛先（DRST）。
- 次のメッセージについてメッセージ タグの検証を有効にすると、特定のフィールドの内容が確認および検証されます。検証で合格しなかったメッセージはドロップされます。
  - 利用できない宛先ユーザー部（DUPU）：ユーザー/理由フィールドが存在し、有効な理由およびユーザー コードのみが含まれている必要があります。
  - エラー：すべての必須フィールドが存在し、許可された値のみが含まれている必要があります。各エラー メッセージには、そのエラー コードの必須フィールドが含まれている必要があります。
  - 通知：ステータス タイプおよびステータス情報フィールドには、許可された値のみが含まれている必要があります。

- アプリケーション サーバー プロセス（ASP）の厳密な状態検証を有効にすると、システムは M3UA セッションの ASP の状態を維持し、検証結果に基づいて ASP メッセージを許可またはドロップします。ASP の厳密な状態検証を無効にすると、すべての ASP メッセージが検査されずに転送されます。

M3UA インスペクションの制限事項

次に、M3UA インスペクションに関する制限事項の一部を示します。

- NAT は、M3UA データに埋め込まれている IP アドレスではサポートされません。
- M3UA の厳密なアプリケーション サーバー プロセス（ASP）状態の確認は、SCTP ステートフル インスペクションと依存性があります。SCTP ステート バイパスと M3UA の厳密な ASP 確認は、同じトラフィック上で実行しないでください。
- 厳密な ASP のステート チェックが必要なのは、ステートフル フェールオーバーが必要な場合、またはクラスタ内での動作が必要な場合です。ただし、厳密な ASP のステート チェックは、上書きモードでのみ動作し、ロードシェアリングまたはブロードキャスト モードで実行している場合は動作しません（RFC 4666 より）。インスペクションは、エンドポイントごとに ASP が 1 つだけあると仮定します。

RADIUS アカウンティング インスペクションの概要

RADIUS アカウンティング インスペクションの目的は、RADIUS サーバーを使用した GPRS ネットワークの過剰請求攻撃を防ぐことです。RADIUS アカウンティング インスペクションを実行するために キャリア ライセンスは必要ありませんが、GTP インスペクションを実行し、GPRS を設定しなければ意味がありません。

GPRS ネットワークの過剰請求攻撃は、コンシューマに対して、利用していないサービスの請求を行います。この場合、悪意のある攻撃者は、サーバーへの接続をセットアップし、SGSN から IP アドレスを取得します。攻撃者がコールを終了しても、攻撃者のサーバーはパケットの送信を続けます。このパケットはGGSNによってドロップされますが、サーバーからの接続はアクティブなままです。攻撃者に割り当てられていたIPアドレスが解放され、正規ユーザーに再割り当てされるので、正規ユーザーは、攻撃者が利用するサービスの分まで請求されることになります。

RADIUSアカウンティングインスペクションは、GGSNへのトラフィックが正規のものかどうかを確認することにより、このような攻撃を防ぎます。RADIUSアカウンティングの機能を正しく設定しておくと、ASA は、RADIUS アカウンティング要求の開始メッセージと終了メッセージに含まれる Framed IP 属性との照合結果に基づいて接続を切断します。終了メッセージの Framed IP 属性の IP アドレスが一致している場合、ASA は、一致する IP アドレスを持つ送信元との接続をすべて検索します。

ASA でメッセージを検証できるように、RADIUS サーバーとの事前共有秘密キーを設定することもできます。共有秘密が設定されていない場合、ASA は、ソース IP アドレスが RADIUS メッセージを送信できるよう設定されたIPアドレスであるということだけをチェックします。

（注） GPRS をイネーブルにして RADIUS アカウンティング インスペクションを使用すると、ASA はアカウンティング要求の STOP メッセージで 3GPP-Session-Stop-Indicator をチェックして、セカンダリ PDP コンテキストを正しく処理します。具体的には、ASA では、アカウンティング要求の終了メッセージがユーザーセッションおよび関連するすべての接続を終了する前に、メッセージに 3GPP-SGSN-Address 属性が含まれる必要があります。一部のサードパーティの GGSN は、この属性をデフォルトでは送信しない場合があります。

モバイル ネットワーク プロトコル インスペクションのライセンス

次のプロトコルのインスペクションには、次の表に記載されているライセンスが必要です。

- GTP
- SCTP。
- Diameter
- M3UA

モデル	ライセンス要件
ASA 仮想 (全モデル)	キャリア ライセンス（デフォルトではイネーブル）
Cisco Secure Firewall 3100	キャリア ライセンス
Firepower 4100	キャリア ライセンス
Firepower 9300	キャリア ライセンス
他のすべてのモデル	キャリア ライセンスは他のモデルでは使用できません。これらのプロトコルは検査できません。

GTP インスペクションのデフォルト

GTP インスペクションはデフォルトではイネーブルになっていません。ただし、ユーザー自身のインスペクションマップを指定せずにイネーブルにすると、次の処理を行うデフォルトマップが使用されます。マップを設定する必要があるのは、異なる値が必要な場合のみです。

- エラーは許可されません。
- 要求の最大数は 200 です。
- トンネルの最大数は 500 です。これは、PDP コンテキスト（エンドポイント）の数に相当します。
- GTP エンドポイントのタイムアウトは 30 分です。エンドポイントには、GSN（GTPv0,1）および SGW/PGW（GTPv2）が含まれています。
- PDP コンテキストのタイムアウトは 30 分です。GTPv2 では、これはベアラ- コンテキスト タイムアウトです。
- 要求のタイムアウトは 1 分です。
- シグナリング タイムアウトは 30 分です。
- トンネリングのタイムアウトは 1 時間です。
- T3 応答タイムアウトは 20 秒です。
- 不明なメッセージ ID が許可されます。**match message v1/v2 id range** コマンドを設定して、サポートされていないコマンドや許可されていないコマンドをドロップしたり、ログに記録したりできます。未定義のメッセージやシステムでサポートされていない GTP リリースで定義されたメッセージは不明と見なされます。

モバイル ネットワーク インスペクションの設定

モバイル ネットワークで使用されるプロトコルのインスペクションはデフォルトで有効になっていません。モバイル ネットワークをサポートするには、それらを設定する必要があります。

手順

ステップ 1 （任意）GTP インスペクション ポリシー マップの設定 （12 ページ）。

ステップ 2 （オプション）SCTP インスペクション ポリシー マップの設定 （17 ページ）。

ステップ 3 （オプション）Diameter インスペクション ポリシー マップの設定 （19 ページ）。

ソフトウェアではまだサポートされていない属性値ペア（AVP）でフィルタリングする場合は、Diameter インスペクション ポリシー マップで使用するカスタム AVP を作成できます。カスタム Diameter 属性値ペア（AVP）の作成 （24 ページ）を参照してください。

ステップ 4 （任意）暗号化された Diameter TCP/TLS トラフィックを検査する場合は、次の説明に従って、必要な TLS プロキシを作成します。 暗号化された Diameter セッションの検査 （25 ページ）

ステップ 5 （任意） M3UA インスペクション ポリシー マップの設定 （38 ページ）

ステップ 6 モバイル ネットワーク インスペクションのサービス ポリシーの設定 （42 ページ）。

ステップ 7 （オプション）RADIUS アカウンティング インスペクションの設定 （44 ページ）。

RADIUS アカウンティング インスペクションは、過剰請求攻撃から保護します。

GTP インスペクション ポリシー マップの設定

GTP トラフィックで追加のパラメーターを実行する際にデフォルト マップがニーズを満たさない場合は、GTP マップを作成し、設定します。

始める前に

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 GTP インスペクション ポリシー マップを作成します。**policy-map type inspect gtp** *policy_map_name*

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。**description** *string*

ステップ 3 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 **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**] **apn 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する Access Point Name (APN) に一致します。
- **match** [**not**] **message** {**v1** | **v2**} **id** {*message_id* | **range** *message_id_1 message_id_2*}：メッセージ ID（1 ～ 255）を照合します。1 つの ID または ID の範囲を指定できます。メッセージが GTPv0/1 用（**v1**）か GTPv2 用（**v2**）かを指定する必要があります。
- **match** [**not**] **message length min** *bytes* **max** *bytes*：UDP ペイロード（GTP ヘッダーと残りのメッセージ）の長さが最小値と最大値の間（1 ～ 65536）であるメッセージを照合します。
- **match** [**not**] **msisdn regex** {*regex_name* | **class** *class_name*}：PDP コンテキスト作成要求、セッション作成要求、およびベアラー変更応答のメッセージ内のモバイル ステーション国際サブスクライバ電話番号（MSISDN）情報要素を指定した正規表現または正規表現クラスと照合します。正規表現では、特定の MSISDN または MSISDN の範囲を最初の x 桁に基づいて識別できます。MSISDN フィルタリングは GTPv1 および GTPv2 のみでサポートされています。
- **match** [**not**] **selection-mode** *mode_value*：PDP コンテキスト作成要求内の選択モードの情報要素を照合します。選択モードでは、メッセージにアクセス ポイント名（APN）の発信元を指定しますが、次のいずれかになります。選択モードフィルタリングは、GTPv1 および GTPv2 のみでサポートされています。
  - 0：確認済み。APN はモバイル ステーションまたはネットワークによって指定されており、サブスクリプションが確認されています。
  - 1：モバイル ステーション。APN はモバイル ステーションによって指定されており、サブスクリプションは確認されていません。
  - 2：ネットワーク。APN はネットワークによって指定されており、サブスクリプションは確認されていません。
  - 3：予約済み（未使用）
- **match** [**not**] **version** {*version_id* | **range** *version_id_1 version_id_2*}：0 ～ 255 のいずれかの GTP バージョンに一致します。1 つのバージョンまたはバージョンの範囲を指定できます。

b) 次のコマンドのいずれかを入力して、一致する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。

- **drop** [**log**]：一致するすべてのパケットをドロップします。システム ログ メッセージも送信するには、**log** キーワードを追加します。
- **rate-limit** *message_rate*：メッセージのレートを制限します。このオプションでは、**message id** のみ使用できます。

ポリシーマップでは、複数の **match** コマンドを指定できます。**match** コマンドの順序については、複数のトラフィック クラスの処理方法を参照してください。

ステップ 4 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。

- **anti-replay**[*window_size*]：GTP-U メッセージのスライディング ウィンドウを指定することによって、アンチリプレイをイネーブルにします。スライディング ウィンドウのサイズはメッセージの数であり、128、256、512、または 1024 になります。サイズを指定しないと、デフォルトで 512 になります。有効なメッセージが表示されると、ウィンドウは新しいシーケンス番号に移行します。シーケンス番号は 0 ～ 65535 の範囲であり、最大値に達するとラッピングされます。また、これらは PDP コンテキストごとに一意です。メッセージは、シーケンス番号がウィンドウ内であれば有効と見なされます。アンチリプレイは、ハッカーが GTP データ パケットをキャプチャし、それらをリプレイするときに発生する可能性があるセッション ハイジャックや DoS 攻撃を防ぐのに役立ちます。
- **[permit errors]**：無効な GTP パケットや別の方法で解析されるとドロップされるすべてのパケットを許可します。パケットは、ポリシーマップで定義したアクションに基づいてドロップできます。
- **request-queue** *max_requests*：キューで応答待ちができる GTP 要求数の最大値を設定します。デフォルトは 200 です。この上限に達した後に新しい要求が到着すると、最も長い時間キューに入っていた要求が削除されます。「Error Indication」、「Version Not Supported」および「SGSN Context Acknowledge」というメッセージは、要求と見なされないため、応答待ち要求のキューに入れられません。
- **tunnel-limit** *max_tunnels*：許可されるアクティブな GTP トンネルの最大数を設定します。これは、PDP コンテキストまたはエンドポイントの数に相当します。デフォルトは 500 です。このコマンドで指定したトンネル数に達すると、新しい要求はドロップされます。
- **timeout** {**endpoint** | **pdp-context** | **request** | **signaling** | **t3-response** | **tunnel**} *time*：指定したサービスのアイドル 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。このコマンドは、タイムアウトごとに別々に入力します。

- **endpoint**：GTP エンドポイントが削除されるまでの非アクティブ時間の最大値。
- **pdp-context**：GTP セッションの PDP コンテキストを削除するまでの非アクティブ時間の最大値。GTPv2 では、これはベアラー コンテキストです。
- **request**：要求キューから要求が削除されるまでの非アクティブ時間の最大値。ドロップされた要求への後続の応答もドロップされます。
- **signaling**：GTP シグナリングが削除されるまでの非アクティブ時間の最大値。
- **t3-response**：接続を除去する前に応答を待機する最大時間。
- **tunnel**：GTP トンネルが切断されるまでの非アクティブ時間の最大値。

ステップ 5 パラメータ コンフィギュレーション モードになっている間に、IP パケットとアンチスプーフィングに対して GTP-U チェックを設定します。

gtp-u-header-check[**anti-spoofing** [**gtpv2-dhcp-bypass** | **gtpv2-dhcp-drop**]]

キーワードを指定しないと、このコマンドは GTP データ パケットの内部ペイロードが有効な IP パケットであるかどうかを確認し、非 IP ヘッダーがある場合はそのパケットをドロップします。

また、**anti-spoofing** キーワードを含めると、内部ペイロードの IP ヘッダー内のモバイル ユーザー IP アドレスが GTP 制御メッセージ（セッション作成応答など）に割り当てた IP アドレスと一致しているかどうかを確認し、IP アドレスが一致しない場合は GTP-U メッセージをドロップします。このチェックでは、IPv4、IPv6、および IPv4v6 PDN タイプがサポートされています。モバイル端末が DHCP を使用してそのアドレスを取得する場合、GTPv2 でのエンドユーザーの IP アドレスは 0.0.0.0（IPv4）または *prefix*::0（IPv6）になります。その場合、システムは内部パケットで検出した最初の IP アドレスを使用してエンドユーザー IP アドレスを更新します。次のキーワードを使用して、DHCP で取得したアドレスのデフォルトの動作を変更できます。

- **gtpv2-dhcp-bypass**：0.0.0.0 または *prefix*::0 アドレスを更新しないでください。その代わりに、エンドユーザーの IP アドレスが 0.0.0.0 または *prefix*::0 の場合はパケットを許可します。IP アドレスの取得に DHCP を使用すると、このオプションはアンチスプーフィング チェックをバイパスします。
- **gtpv2-dhcp-drop**：0.0.0.0 または *prefix*::0 アドレスを更新しません。その代わりに、エンドユーザーの IP アドレスが 0.0.0.0 または *prefix*::0 の場合はすべてのパケットをドロップします。このオプションは、IP アドレスの取得に DHCP を使用するユーザーへのアクセスを防ぎます。

ステップ 6 必要に応じて、パラメータ コンフィギュレーション モードに入っている間に、IMSI プレフィックス フィルタリングを設定します。

mcc *country_code* **mnc** *network_code*

drop mcc *country_code* **mnc** *network_code*

コマンドは必要な回数入力して、ターゲットとなるすべての MCC/MNC ペアを指定できますが、ポリシーマップ内のすべてのコマンドは **mcc** または **drop mcc** である必要があります。これらのコマンドを組み合わせることはできません。

デフォルトでは、GTP インスペクションは、有効なモバイル カントリ コード（MCC）とモバイル ネットワーク コード（MNC）の組み合わせをチェックしません。IMSI プレフィックス フィルタリングを設定すると、受信パケットの IMSI の MCC と MNC が、設定された MCC と MNC の組み合わせと比較されます。次に、コマンドに基づいて次のいずれかのアクションが実行されます。

- **mcc** コマンド：一致しない場合、パケットはドロップされます。
- **drop mcc** コマンド：一致する場合、パケットはドロップされます。

モバイル カントリ コードは 0 以外の 3 桁の数字で、1 桁または 2 桁の値のプレフィックスとして 0 が追加されます。モバイル ネットワーク コードは 2 桁または 3 桁の数字です。

許可またはドロップするすべての MCC と MNC の組み合わせを追加します。デフォルトでは、ASA は MNC と MCC の組み合わせが有効であるかどうかをチェックしないため、設定した組み合わせが有効であるかどうかを確認する必要があります。MCC および MNC コードの詳細については、ITU E.212 勧告『*Identification Plan for Land Mobile Stations*』を参照してください。

ステップ 7 必要に応じて、パラメータ コンフィギュレーション モードの間に場所のロギングを有効にします。

location-logging [**cell-id**]

サブスクライバの場所をログに記録し、モバイル端末の場所の変更を追跡します。場所の変更を追跡すると、不正なローミング請求を識別するのに役立ちます。場所のログを有効にすると、システムは International Mobile Subscriber Identity（IMSI）ごとに新しい（メッセージ 324010）場所または変更された（メッセージ 324011）場所の syslog メッセージを生成します。

ユーザーが現在登録されているセル ID をログ メッセージに含める場合は、**cell-id** パラメータを指定します。セル ID は、セル グローバル識別（CGI）または E-UTRAN セル グローバル識別子（ECGI）から抽出されます。

ステップ 8 必要に応じて、パラメータ コンフィギュレーション モードに入っている間に、GSN または PGW プーリングを設定します。

permit-response to-object-group *SGSN-SGW_name* **from-object-group** *GSN-PGW_pool*

ASA が GTP インスペクションを実行する場合、デフォルトで ASA は、GTP 要求で指定されていない GSN または PGW からの GTP 応答をドロップします。これは、GSN または PGW のプール間でロードバランシングを使用して、GPRS の効率とスケーラビリティを高めているときに発生します。

GSN/PGW プーリングを設定し、ロード バランシングをサポートするために、GSN/PGW エンドポイントを指定するネットワーク オブジェクト グループを作成し、これを **from-object-group** パラメータで指定します。同様に、SGSN/SGW のためにネットワーク オブジェクト グループを作成し、**to-object-group** パラメータとして選択します。応答を行う GSN/PGW が GTP 要求の送信先 GSN/PGW と同じオブジェクト グループに属しており、応答している GSN/PGW によ

る GTP 応答の送信が許可されている先のオブジェクト グループに SGSN/SGW がある場合に、ASA で応答が許可されます。

ネットワーク オブジェクト グループは、エンドポイントをホスト アドレスまたはエンドポイントを含むサブネットから識別できます。

例：

次に、GSN/PGW プーリングの例を示します。クラス C ネットワーク全体が GSN/PGW プールとして定義されていますが、ネットワーク全体を指定する代わりに、複数の個別の IP アドレスを **network-object** コマンドで 1 つずつ指定できます。この例では、次に、プールから SGSN/SgW への応答を許可するように、GTP インスペクション マップを変更します。

```
hostname(config)# object-group network gsnpool32
hostname(config-network)# network-object 192.168.100.0 255.255.255.0
hostname(config)# object-group network sgsn32
hostname(config-network)# network-object host 192.168.50.100

hostname(config)# policy-map type inspect gtp gtp-policy
hostname(config-pmap)# parameters
hostname(config-pmap-p)# permit-response to-object-group sgsn32
from-object-group gsnpool32
```

例

次の例は、ネットワークのトンネル数を制限する方法を示しています。

```
hostname(config)# policy-map type inspect gtp gmap
hostname(config-pmap)# parameters
hostname(config-pmap-p)# tunnel-limit 3000

hostname(config)# policy-map global_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect gtp gmap

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

次のタスク

マップを使用するためのインスペクション ポリシーを設定できるようになりました。モバイル ネットワーク インスペクションのサービス ポリシーの設定 （42 ページ） を参照してください。

SCTP インスペクション ポリシー マップの設定

レート制限などのアプリケーション固有のペイロードプロトコル ID（PPID）に基づいて SCTP トラフィックに代替アクションを適用するには、サービス ポリシーで使用される SCTP インスペクション ポリシー マップを作成します。

（注） PPID はデータのかたまりの中にあり、特定のパケットは複数のデータ チャンクまたは 1 つの制御チャンクを持つことができます。パケットに 1 つの制御チャンクまたは複数のデータ チャンクが含まれている場合、割り当てられたアクションがドロップされてもパケットはドロップされません。たとえば、PPID 26 をドロップする SCTP インスペクション ポリシー マップを設定すると、PPID 26 データ チャンクは、Diameter PPID データ チャンクを持つパケットに結合され、そのパケットはドロップされません。

手順

ステップ 1 SCTP インスペクション ポリシー マップを作成します。**policy-map type inspect sctp** *policy_map_name*

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。**description** *string*

ステップ 3 SCTP データ チャンクの PPID に基づいて、トラフィックをドロップ、レート制限、またはログに記録します。

a) PPID に基づいてトラフィックを識別します。

match[**not**] **ppid** *ppid_1* [*ppid_2*]

ppid_1 は PPID 番号（0 ～ 4294967295）または名前です（使用可能な名前については CLI ヘルプを参照してください）。PPID の範囲を指定するには、2 番目（より大きい）の PPID、*ppid_2* を含めることができます。**match not ppid** を使用して PPID または範囲に一致しないトラフィックを特定します。

SCTP PPID の現在のリストは http://www.iana.org/assignments/sctp-parameters/sctp-parameters.xhtml#sctp-parameters-25 で確認できます。

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。

- **drop**：一致するすべてのパケットをドロップまたはログに記録します。
- **log**：システム ログ メッセージを送信します。
- **rate-limit** *rate*：メッセージのレートを制限します。レートは、キロビット/秒（kbps）単位です。

c) 選択的に処理するすべての PPID を識別するまで、プロセスを繰り返します。

例

次の例では、未割り当ての PPID（この例の作成時点で未割り当て）をドロップし、PPID 32 ～ 40 をレート制限し、Diameter PPID をログに記録するインスペクション ポリシー マップを作成します。このサービス ポリシーは、すべての SCTP トラフィックを照合する inspection_default クラスにインスペクションを適用します。

```
policy-map type inspect sctp sctp-pmap
 match ppid 58 4294967295
  drop
 match ppid 26
  drop
 match ppid 49
  drop
 match ppid 32 40
  rate-limit 1000
 match ppid diameter
  log

policy-map global_policy
 class inspection_default
  inspect sctp sctp-pmap
!
service-policy global_policy global
```

次のタスク

マップを使用するためのインスペクション ポリシーを設定できるようになりました。モバイル ネットワーク インスペクションのサービス ポリシーの設定 （42 ページ） を参照してください。

Diameter インスペクション ポリシー マップの設定

さまざまな Diameter プロトコル要素でフィルタリングするための Diameter インスペクション ポリシー マップを作成できます。その後、接続を選択的にドロップまたはログに記録できます。

Diameter メッセージ フィルタリングを設定するには、これらのプロトコル要素は RFC および技術仕様で定義されているので、これらの要素について詳しい知識を持っている必要があります。たとえば、IETF には、http://www.iana.org/assignments/aaa-parameters/aaa-parameters.xhtml に示す登録済みアプリケーション、コマンドコード、および属性値ペアのリストがありますが、Diameterインスペクションではリストされているすべての項目をサポートしていません。技術仕様については、3GPP Web サイトを参照してください。

始める前に

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、Diameter 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との照合をグループ化します。または、**match** コマンドを直接ポリシー マップに指定できます。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) クラス マップを作成します。**class-map type inspect diameter** [**match-all** | **match-any**] *class_map_name*

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します。**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**] **application-id** *app_id* [*app_id_2*]：アプリケーション識別子を照合します。*app_id* は Diameter アプリケーションの名前または番号（0 ～ 4294967295）です。照合する連続番号が付されたアプリケーションの範囲がある場合は、2 番目の ID を含めることができます。アプリケーションの名前または番号別に範囲を定義でき、第 1 ID および第 2 ID の間のすべての番号に適用されます。

  これらのアプリケーションは IANA に登録されます。次のコア アプリケーションがサポートされますが、他のアプリケーションもフィルタ処理できます。アプリケーション名のリストについては、CLI ヘルプを参照してください。

  - **3gpp-rx-ts29214** （16777236）
  - **3gpp-s6a** （16777251）

- **3gpp-s9** （16777267）
- **common-message** （0）。（基本 Diameter プロトコル）

- **match** [**not**] **command-code** *code* [*code_2*]：コマンド コードを照合します。*code* は Diameter コマンド コードの名前または番号（0 ～ 4294967295）です。照合する連続番号が付されたコマンド コードの範囲がある場合は、2 番目のコードを含めることができます。コマンド コードの名前または番号別に範囲を定義でき、第 1 コードおよび第 2 コードの間のすべての番号に適用されます。

  たとえば、次のコマンドは、Capability Exchange Request/Answer コマンド コードを照合します。

  ```
  match command-code cer-cea
  ```

- 属性値ペア（AVP）を照合します。

  属性によってのみ AVP を照合するには、次の手順を実行します。

  match[**not**] **avp** コード[*code_2*] [**vendor-id** *id_number*]

  属性の値に基づいて AVP を照合する場合：

  match[**not**] **avp** コード[**vendor-id** *id_number*]値

  それぞれの説明は次のとおりです。

  - *code*：属性値ペアの名前または番号（1 ～ 4294967295）。最初のコードについては、カスタム AVP、RFC または 3GPP 技術仕様に登録されている AVP、およびソフトウェアで直接サポートされている AVP の名前を指定できます。特定の範囲の AVP を照合する場合は、2 つ目のコードを番号のみで指定します。値によって AVP を照合する場合は、2 つ目のコードを指定できません。AVP 名のリストについては、CLI ヘルプを参照してください。
  - **vendor-id** *id_number*：（任意）ベンダーの ID 番号（0 ～ 4294967295）も照合します。たとえば、3GPP ベンダー ID は 10415、IETF は 0。
  - *value*：AVP の値の部分。これは、AVP の データ タイプがサポートされている場合にのみ設定できます。たとえば、アドレス データ タイプがある AVP の IP アドレスを指定できます。次に、サポートされているデータ タイプの値オプションの特定の構文を示します。
    - [Diameter Identity]、[Diameter URI]、[Octet String]：これらのデータ タイプの照合には正規表現または正規表現クラス オブジェクトを使用します。

      {**regex** *regex_name* | **class** *regex_class*}
    - [Address]：照合する IPv4 または IPv6 アドレスを指定します。たとえば、10.100.10.10 または 2001:DB8::0DB8:800:200C:417A。
    - [Time]：開始日時と終了日時を指定します。両方を指定する必要があります。時間は 24 時間形式で指定します。

date *year month day* **time** *hh:mm:ss* **date** *year month day* **time** *hh:mm:ss*

次に例を示します。

```
date 2015 feb 5 time 12:00:00 date 2015 mar 9 time 12:00:00
```

- [Numeric]：番号の範囲を指定します。

  range *number_1 number_2*

  有効な番号の範囲は、データ タイプによって異なります。

  - Integer32：-2147483647 ～ 2147483647
  - Integer64：-9223372036854775807 ～ 9223372036854775807
  - Unsigned32：0 ～ 4294967295
  - Unsigned64：0 ～ 18446744073709551615
  - Float32：8 桁の小数点表現
  - Float64：16 桁精度の小数点表記

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 Diameter インスペクション ポリシー マップを作成します。**policy-map type inspect diameter** *policy_map_name*

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）説明をポリシー マップに追加します。**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。

- Diameter 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。**class** *class_map_name*
- Diameter クラス マップで説明されている **match** コマンドのいずれかを使用して、ポリシー マップに直接トラフィックを指定します。

b) 次のコマンドのいずれかを入力して、一致する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。

- **drop**：一致するすべてのパケットをドロップします。
- **drop-connection**：パケットをドロップし、接続を閉じます。
- **log**：システム ログ メッセージを送信します。

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、複数のトラフィック クラスの処理方法を参照してください。

例：

```
hostname(config)# policy-map type inspect diameter diameter-map
hostname(config-pmap)# class diameter-class-map
hostname(config-pmap-c)# drop
hostname(config-pmap-c)# match command-code cer-cea
hostname(config-pmap-c)# log
```

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。

- **unsupported** {**application-id** |**command-code** |**avp**} **action log**：ロギングをサポートされていない直径要素に対してイネーブルにします。これらのオプションでは、ソフトウェアで直接サポートされていないアプリケーション ID、コマンド コード、および AVP が指定されます。デフォルトでは、ロギングなしで要素が許可されています。コマンドを 3 回入力して、すべての要素のロギングを有効にできます。
- **strict-diameter** {**state** | **session**}：Diameter プロトコルの RFC 6733 への厳密な準拠をイネーブルにします。デフォルトでは、インスペクションによって、Diameter のフレームが RFC に準拠していることが確認されます。コマンドを 2 回入力することで、**state** マシン検証または **session** 関連メッセージの検証、あるいはその両方を追加できます。

例：

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)# unsupported application-id action log
hostname(config-pmap-p)# unsupported command-code action log
hostname(config-pmap-p)# unsupported avp action log
hostname(config-pmap-p)# strict-diameter state
hostname(config-pmap-p)# strict-diameter session
```

例

次の例は、一部のアプリケーションをログに記録し、特定の IP アドレスをブロックする方法を示しています。

```
class-map type inspect diameter match-any log_app
```

```
  match application-id 3gpp-s6a
  match application-id 3gpp-s13

class-map type inspect diameter match-all block_ip
  match command-code cer-cea
  match avp host-ip-address 1.1.1.1

policy-map type inspect diameter diameter_map
  parameters
    unsupported application-id log
  class log_app
    log
  class block_ip
    drop-connection

policy-map global_policy
  class inspection_default
    inspect diameter diameter_map

service-policy global_policy global
```

次のタスク

マップを使用するためのインスペクションポリシーを設定できるようになりました。モバイル ネットワーク インスペクションのサービス ポリシーの設定 （42 ページ） を参照してください。

カスタム Diameter 属性値ペア（AVP）の作成

新しい属性値ペア（AVP）が定義され、登録されると、カスタム Diameter AVP を作成して、Diameterインスペクション ポリシーマップにそれらを定義し、使用することができます。RFC または AVP を定義するその他のソースから AVP の作成に必要な情報を取得します。

カスタム AVP は、AVP 照合用の Diameterインスペクション ポリシーマップまたはクラス マップで使用する場合にのみ、作成します。

手順

カスタム Diameter AVP を作成します。

diameter avp*name* **code** *value* **data-type** *type* [**vendor-id** *id_number*] [**description** *text*]

それぞれの説明は次のとおりです。

- *name*：作成しているカスタム AVP の名前（最大 32 文字）。Diameter インスペクション ポリシー マップまたはクラス マップでの match avp コマンドでこの名前を参照します。
- **code** *value*：カスタム AVP コード値（256 ～ 4294967295）。システムで定義済みのコードとベンダー ID の組み合わせを入力することはできません。
- **data-type** *type*：AVP のデータ タイプ。次のいずれかの型で AVP を定義できます。新しい AVP が別の型の場合は、その型のカスタム AVP は作成できません。

- **address**：IP アドレスの場合。
- **diameter-identity**：Diameter のアイデンティティ データ。
- **diameter-uri**：Diameter の Uniform Resource Identifier（URI）。
- **float32**：32 ビット浮動小数点。
- **float64**：64 ビット浮動小数点。
- **int32**：32 ビット整数。
- **int64**：64 ビット整数。
- **octetstring**：オクテット文字列。
- **time**：時間の値。
- **uint32**：32 ビットの符号なし整数。
- **uint64**：64 ビットの符号なし整数。

- **vendor-id** *id_number*：（任意）AVP を定義したベンダーの 0 ～ 4294967295 の ID 番号。たとえば、3GPP ベンダー ID は 10415、IETF は 0。
- **description** *text*：（任意）AVP の説明（最大 80 文字）。スペースを含める場合は、説明を引用符で囲みます。

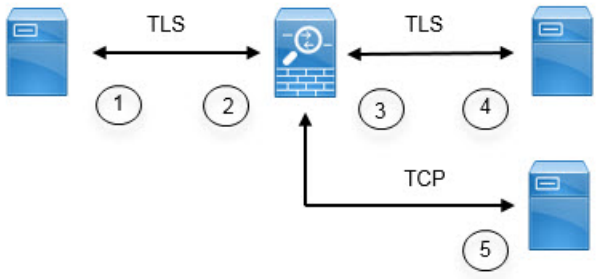
暗号化された Diameter セッションの検査

Diameter アプリケーションが TCP 上で暗号化されたデータを使用する場合、インスペクションはメッセージのフィルタリング ルールを実装するためにパケット内を確認することはできません。したがって、フィルタリング ルールを作成し、それらを暗号化された TCP トラフィックにも適用する場合は、TLS プロキシを設定する必要があります。暗号化されたトラフィックで厳密なプロトコルを適用するには、プロキシも必要です。この設定は SCTP/DTLS トラフィックには適用されません。

TLS プロキシは中間者として機能します。このプロキシは、トラフィックを復号化し、検査してから再度暗号化し、目的の宛先に送信します。したがって、接続の両側（Diameter サーバーと Diameter クライアント）は ASA を信頼する必要があり、すべての当事者が必要な証明書を保有している必要があります。TLS プロキシを実装するには、デジタル証明書を十分に理解しておく必要があります。ASA 全般設定ガイドのデジタル証明書に関する章を参照してください。

次の図は、Diameter のクライアントおよびサーバーと ASA の間の関係と、信頼を確立するための認定要件を示します。このモデルでは、Diameter クライアントは MME（モビリティ マネージメント エンティティ）であり、エンドユーザーではありません。リンクの各側の CA 証明書は、リンクの反対側の証明書の署名に使用されるものです。たとえば、ASA プロキシ TLS サーバー CA 証明書は、Diameter/TLS クライアント証明書の署名に使用されるものです。

図 *2 : Diameter TLS* インスペクション



1	Diameter TLS クライアント（MME） • クライアント ID 証明書 • ASA TLS プロキシ サーバーの ID 証明書の署名に使用される CA 証明書	2	ASA プロキシ TLS サーバー • サーバー ID 証明書 • Diameter TLS クライアントの ID 証明書の署名に使用される CA 証明書
3	ASA プロキシ TLS クライアント • クライアント ID（スタティックまたは LDC）証明書 • Diameter TLS サーバーの ID 証明書の署名に使用される CA 証明書	4	Diameter TLS サーバー（フル プロキシ） • サーバー ID 証明書 • ASA プロキシ TLS クライアントの ID 証明書の署名に使用される CA 証明書
5	Diameter TCP サーバー（TLS オフロード）	—	—

Diameter インスペクション用の TLS プロキシを設定するには、次のオプションがあります。

- フル TLS プロキシ：ASA および Diameter クライアントと ASA および Diameter サーバー間のトラフィックを暗号化します。TLS サーバーとの信頼関係を確立するには、次のオプションがあります。
  - スタティック プロキシ クライアント トラストポイントを使用します。ASA は、Diameter サーバーとの通信時に、すべての Diameter クライアントに同じ証明書を示します。Diameter サーバーにとって全クライアントが同じように見えるので、クライアントごとに差別化サービスを提供することはできません。一方、このオプションは LDC 方式よりも高速です。
  - ローカル ダイナミック証明書（LDC）を使用します。このオプションを使用すると、ASA は Diameter サーバーとの通信時に、Diameter クライアントごとに一意の証明書を示します。LDC は、公開キーと ASA からの新しい署名を除き、受信したクライアント ID 証明書からのすべてのフィールドを保持します。この方法では、Diameter サーバーでクライアント トラフィックの可視性が向上し、クライアント証明書の特性に基づいて差別化サービスを提供できるようになります。

- TLS オフロード：ASA と Diameter クライアント間のトラフィックを暗号化しますが、ASA と Diameter サーバー間でクリアテキスト接続を使用します。このオプションは、デバイス間のトラフィックが保護された場所から離れることがないと確信している場合に、Diameter サーバーが ASA と同じデータセンターにあれば実行可能です。TLS オフロードを使用すると、必要な暗号化処理量が減るので、パフォーマンスを向上させることができます。これは、オプションの中で最速です。Diameter サーバーは、クライアントの IP アドレスのみに基づいて差別化サービスを適用できます。

3 つすべてのオプションは、ASA と Diameter クライアント間の信頼関係に対して同じ設定を使用します。

（注） TLS プロキシは TLSv1.0 ～ 1.2 を使用します。TLS のバージョンと暗号スイートを設定できます。

次の項では、Diameter インスペクション用の TLS プロキシを設定する方法について説明します。

Diameter クライアントとのサーバー信頼関係の設定

ASA は、Diameter クライアントに対して TLS プロキシ サーバーとして機能します。相互信頼関係を確立するには：

- ASA のサーバー証明書への署名に使用された認証局（CA）証明書を Diameter クライアントにインポートする必要があります。これは、クライアントの CA 証明書ストアまたはクライアントが使用する他の場所に保存されている場合があります。証明書の使用の詳細については、クライアントのドキュメントを参照してください。
- ASA がクライアントを信頼できるように、Diameter TLS クライアントの証明書への署名に使用された CA 証明書をインポートする必要があります。

次の手順では、Diameter クライアントの証明書への署名に使用された CA 証明書をインポートし、ASA TLS プロキシ サーバーで使用する ID 証明書をインポートする方法について説明します。ID 証明書をインポートする代わりに、ASA で自己署名証明書を作成できます。

手順

ステップ 1 Diameter クライアントの証明書への署名に使用されている CA 証明書を ASA トラストポイントにインポートします。

この手順によって、ASA が Diameter クライアントを信頼できます。

a) Diameter クライアント用のトラストポイントを作成します。

この例では、**enrollment terminal** は、証明書を CLI に張り付けることを示しています。トラストポイントは **diameter-clients** と呼ばれます。

```
ciscoasa(config)# crypto ca trustpoint diameter-clients
ciscoasa(ca-trustpoint)# revocation-check none
ciscoasa(ca-trustpoint)# enrollment terminal
```

b) 証明書を追加します。

```
ciscoasa(config)# crypto ca authenticate diameter-clients
Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself
MIIDRTCCAu+gAwIBAgIQKVcqP/KW74VP0NZzL+JbRTANBgkqhkiG9w0BAQUFADCB
[certificate data omitted]
/7QEM8izy0EOTSErKu7Nd76jwf5e4qttkQ==
quit

INFO: Certificate has the following attributes:
Fingerprint: 24b81433 409b3fd5 e5431699 8d490d34
Do you accept this certificate? [yes/no]: y
Trustpoint CA certificate accepted.

% Certificate successfully imported
```

ステップ 2 証明書をインポートし、ASA プロキシ サーバーの ID 証明書およびキーペア用のトラストポイントを作成します。

この手順によって、Diameter クライアントが ASA を信頼できます。

a) pkcs12 形式で証明書をインポートします。

次の例では、**tls-proxy-server-tp** がトラストポイント名で、**“123”** が復号パス フレーズです。独自のトラストポイント名およびパス フレーズを使用します。

```
ciscoasa (config)# crypto ca import tls-proxy-server-tp pkcs12 "123"

Enter the base 64 encoded pkcs12.
End with a blank line or the word "quit" on a line by itself:
[PKCS12 data omitted]

quit

INFO: Import PKCS12 operation completed successfully

ciscoasa (config)#
```

b) トラストポイントを設定します。

```
ciscoasa(config)# crypto ca trustpoint tls-proxy-server-tp
ciscoasa(ca-trustpoint)# revocation-check none
```

Diameter インスペクション用のスタティック クライアント証明書によるフル TLS プロキシの設定

Diameter サーバーがすべてのクライアントに対して同じ証明書を受け入れることができる場合は、Diameter サーバーと通信するときに使用する ASA 用のスタティック クライアント証明書を設定できます。

この設定では、ASA とクライアント間（Diameter クライアントとのサーバー信頼関係の設定（27 ページ） で説明されているように）、および ASA と Diameter サーバー間に相互の信頼関係を確立する必要があります。ASA と Diameter サーバーの信頼要件は次のとおりです。

- Diameter サーバーの ID 証明書への署名に使用された CA 証明書をインポートする必要があるので、ASA は、TLS ハンドシェイク中にサーバーの ID 証明書を検証できます。
- Diameter サーバーも信頼しているクライアント証明書をインポートする必要があります。Diameter サーバーがまだ証明書を信頼していない場合は、その署名に使用される CA 証明書をサーバーにインポートします。詳細については、Diameter サーバーのドキュメントを参照してください。

手順

ステップ 1 Diameter サーバーの証明書への署名に使用されている CA 証明書を ASA トラストポイントにインポートします。

この手順によって、ASA が Diameter サーバーを信頼できます。

a) Diameter サーバー用のトラストポイントを作成します。

この例では、**enrollment terminal** は、証明書を CLI に張り付けることを示しています。登録用 URL を使用して、CA との自動登録（SCEP）を指定することもできます。トラストポイントは **diameter-server** と呼ばれます。

```
ciscoasa(config)# crypto ca trustpoint diameter-server
ciscoasa(ca-trustpoint)# revocation-check none
ciscoasa(ca-trustpoint)# enrollment terminal
```

b) 証明書を追加します。

```
ciscoasa(config)# crypto ca authenticate diameter-server
Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself
MIIDRTCCAu+gAwIBAgIQKVcqP/KW74VP0NZzL+JbRTANBgkqhkiG9w0BAQUFADCB
[certificate data omitted]
/7QEM8izy0EOTSErKu7Nd76jwf5e4qttkQ==
quit

INFO: Certificate has the following attributes:
Fingerprint: 24b81433 409b3fd5 e5431699 8d490d34
Do you accept this certificate? [yes/no]: y
Trustpoint CA certificate accepted.
```

```
% Certificate successfully imported
```

ステップ 2 証明書をインポートし、ASA プロキシ クライアントの ID 証明書およびキーペア用のトラストポイントを作成します。

この手順によって、Diameter サーバーが ASA を信頼できます。

a) pkcs12 形式で証明書をインポートします。

次の例では、**tls-proxy-client-tp** がトラストポイント名で、**“123”** が復号パス フレーズです。独自のトラストポイント名およびパス フレーズを使用します。

```
ciscoasa (config)# crypto ca import tls-proxy-client-tp pkcs12 "123"

Enter the base 64 encoded pkcs12.
End with a blank line or the word "quit" on a line by itself:
[PKCS12 data omitted]

quit

INFO: Import PKCS12 operation completed successfully

ciscoasa (config)#
```

b) トラストポイントを設定します。

```
ciscoasa(config)# crypto ca trustpoint tls-proxy-client-tp
ciscoasa(ca-trustpoint)# revocation-check none
```

ステップ 3 TLS プロキシを設定します。

a) TLS プロキシに名前を付け、TLS プロキシ コンフィギュレーション モードを開始します。

tls-proxy *name*

b) ASA が Diameter クライアントとの関係においてプロキシ サーバーとして機能するときに使用されるトラストポイントを識別します。

server trust-point *trustpoint_name*

（注） テスト目的の場合、または Diameter クライアントを信頼できると確信している場合は、この手順をスキップして、TLS プロキシ コンフィギュレーションに **no server authenticate-client** コマンドを含めることができます。

c) ASA が Diameter サーバーとの関係においてプロキシ クライアントとして機能するときに使用されるトラストポイントを識別します。

client trust-point *name*

d) （任意）クライアントが使用できる暗号方式を定義します。

client cipher-suite *cipher-list*

ここで、*cipher-list* には、次の任意の組み合わせを含めることができます。

- **3des-sha1**
- **aes128-sha1**
- **aes256-sha1**
- **des-sha1**
- **null-sha1**
- **rc4-sha1**

複数のオプションはスペースで区切ります。

TLS プロキシで使用できる暗号方式を定義しないと、プロキシ サーバーは **ssl cipher** コマンドによって定義されたグローバル暗号スイートを使用します。デフォルトでは、グローバル暗号方式レベルはmediumです。つまり、NULL-SHA、DES-CBC-SHA、およびRC4-MD5を除くすべての暗号方式が使用できます。ASAで一般に使用可能なものとは異なるスイートを使用する場合にのみ、**client cipher-suite** コマンドを指定します。

ASA 上のすべての SSL クライアント接続に最小 TLS バージョンを設定する場合は、**ssl client-version** コマンドを参照してください。デフォルトは TLS v1.0 です。

e) （任意）サーバーが使用できる暗号方式を定義します。

server cipher-suite *cipher-list*

ここで、*cipher-list* には、次の任意の組み合わせを含めることができます。

- **3des-sha1**
- **aes128-sha1**
- **aes256-sha1**
- **des-sha1**
- **null-sha1**
- **rc4-sha1**

複数のオプションはスペースで区切ります。

TLS プロキシで使用できる暗号方式を定義しないと、プロキシ サーバーは **ssl cipher** コマンドによって定義されたグローバル暗号スイートを使用します。デフォルトでは、グローバル暗号方式レベルはmediumです。つまり、NULL-SHA、DES-CBC-SHA、およびRC4-MD5を除くすべての暗号方式が使用できます。ASAで一般に使用可能なものとは異なるスイートを使用する場合にのみ、**server cipher-suite** コマンドを指定します。

ASA 上のすべての SSL サーバー接続に最小 TLS バージョンを設定する場合は、**ssl server-version** コマンドを参照してください。デフォルトは TLS v1.0 です。

例：

```
ciscoasa(config)# tls-proxy diameter-tls-static-proxy
ciscoasa(config-tlsp)# server trust-point tls-proxy-server-tp
ciscoasa(config-tlsp)# client trust-point tls-proxy-client-tp
```

次のタスク

Diameter インスペクションで TLS プロキシを使用できるようになりました。モバイル ネットワーク インスペクションのサービス ポリシーの設定 （42 ページ） を参照してください。

Diameter インスペクション用のローカル ダイナミック証明書によるフル TLS プロキシの設定

Diameter サーバーでクライアントごとに一意の証明書が必要な場合は、ローカル ダイナミック証明書（LDC）を生成するように ASA を設定することができます。これらの証明書は、クライアントが接続している間存在し、その後は破棄されます。

この設定では、ASA とクライアント間（Diameter クライアントとのサーバー信頼関係の設定（27 ページ） で説明されているように）、および ASA と Diameter サーバー間に相互の信頼関係を確立する必要があります。設定は Diameter インスペクション用のスタティック クライアント証明書によるフル TLS プロキシの設定 （29 ページ） で説明するものと同様ですが、Diameter クライアント証明書をインポートする代わりに ASA 上で LDC をセットアップする点が異なります。ASA と Diameter サーバーの信頼要件は次のとおりです。

- Diameter サーバーの ID 証明書への署名に使用された CA 証明書をインポートする必要があるので、ASA は、TLS ハンドシェイク中にサーバーの ID 証明書を検証できます。
- LDC トラストポイントを作成する必要があります。LDC サーバーの CA 証明書をエクスポートし、Diameter サーバーにインポートする必要があります。エクスポート設定は次のとおりです。証明書のインポートの詳細については、Diameter サーバーのドキュメントを参照してください。

手順

ステップ 1 Diameter サーバーの証明書への署名に使用されている CA 証明書を ASA トラストポイントにインポートします。

この手順によって、ASA が Diameter サーバーを信頼できます。

a) Diameter サーバー用のトラストポイントを作成します。

この例では、**enrollment terminal** は、証明書を CLI に張り付けることを示しています。登録用 URL を使用して、CA との自動登録（SCEP）を指定することもできます。トラストポイントは **diameter-server** と呼ばれます。

```
ciscoasa(config)# crypto ca trustpoint diameter-server
```

```
ciscoasa(ca-trustpoint)# revocation-check none
ciscoasa(ca-trustpoint)# enrollment terminal
```

b) 証明書を追加します。

```
ciscoasa(config)# crypto ca authenticate diameter-server
Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself
MIIDRTCCAu+gAwIBAgIQKVcqP/KW74VP0NZzL+JbRTANBgkqhkiG9w0BAQUFADCB
[certificate data omitted]
/7QEM8izy0EOTSErKu7Nd76jwf5e4qttkQ==
quit

INFO: Certificate has the following attributes:
Fingerprint: 24b81433 409b3fd5 e5431699 8d490d34
Do you accept this certificate? [yes/no]: y
Trustpoint CA certificate accepted.

% Certificate successfully imported
```

ステップ 2 ローカル ダイナミック証明書（LDC）に署名するローカル CA を作成します。

a) トラストポイント用の RSA キーペアを作成します。

この例では、キーペア名は ldc-signer-key です。

```
ciscoasa(config)# crypto key generate rsa label ldc-signer-key
INFO: The name for the keys will be: ldc-signer-key
Keypair generation process
ciscoasa(config)#
```

b) LDC 発行元のトラストポイントを作成します。

この例では、トラストポイント名は**ldc-server** で、上記で作成されたキーペアが使用され、自己署名済みの登録が指定されます（**enrollment self**、これは必須です）。ASA の共通名はサブジェクト名として含まれています。Diameter アプリケーションにサブジェクト名に関する固有の要件があるかどうかを確認します。

proxy-ldc-issuer コマンドは、TLS プロキシのダイナミック証明書を発行するトラストポイントに、ローカル CA の役割を定義します。

```
ciscoasa(config)# crypto ca trustpoint ldc-server
ciscoasa(ca-trustpoint)# keypair ldc-signer-key
ciscoasa(ca-trustpoint)# subject-name CN=asa3
ciscoasa(ca-trustpoint)# enrollment self
ciscoasa(ca-trustpoint)# proxy-ldc-issuer
ciscoasa(ca-trustpoint)# exit
```

c) トラストポイントを登録します。

```
ciscoasa(config)# crypto ca enroll ldc-server
```

ステップ 3 TLS プロキシを設定します。

a) TLSプロキシに名前を付け、TLSプロキシコンフィギュレーションモードを開始します。

tls-proxy *name*

b) ASA が Diameter クライアントとの関係においてサーバーとして機能するときに使用されるトラストポイントを識別します。

server trust-point *trustpoint_name*

（注） テスト目的の場合、またはDiameterクライアントを信頼できると確信している場合は、この手順をスキップして、TLS プロキシ コンフィギュレーションに **no server authenticate-client** コマンドを含めることができます。

c) ASAがダイナミック証明書を発行し、Diameterサーバーとの関係においてクライアントとして機能するときに使用される LDC トラストポイントを識別します。

client ldc issuer *name*

d) LDC キーペアを識別します。LDC トラストポイントで定義されている同じキーを指定します。

client ldc key-pair *name*

e) （任意）クライアントが使用できる暗号方式を定義します。

client cipher-suite *cipher-list*

ここで、*cipher-list* には、次の任意の組み合わせを含めることができます。

- **3des-sha1**
- **aes128-sha1**
- **aes256-sha1**
- **des-sha1**
- **null-sha1**
- **rc4-sha1**

複数のオプションはスペースで区切ります。

TLS プロキシで使用できる暗号方式を定義しないと、プロキシ サーバーは **ssl cipher** コマンドによって定義されたグローバル暗号スイートを使用します。デフォルトでは、グローバル暗号方式レベルはmediumです。つまり、NULL-SHA、DES-CBC-SHA、およびRC4-MD5を除くすべての暗号方式が使用できます。ASA で一般に使用可能なものとは異なるスイートを使用する場合にのみ、**client cipher-suite** コマンドを指定します。

ASA 上のすべての SSL クライアント接続に最小 TLS バージョンを設定する場合は、**ssl client-version** コマンドを参照してください。デフォルトは TLS v1.0 です。

f) （任意）サーバーが使用できる暗号方式を定義します。

server cipher-suite *cipher-list*

ここで、*cipher-list* には、次の任意の組み合わせを含めることができます。

- **3des-sha1**
- **aes128-sha1**
- **aes256-sha1**
- **des-sha1**
- **null-sha1**
- **rc4-sha1**

複数のオプションはスペースで区切ります。

TLS プロキシで使用できる暗号方式を定義しないと、プロキシ サーバーは **ssl cipher** コマンドによって定義されたグローバル暗号スイートを使用します。デフォルトでは、グローバル暗号方式レベルは medium です。つまり、NULL-SHA、DES-CBC-SHA、および RC4-MD5 を除くすべての暗号方式が使用できます。ASA で一般に使用可能なものとは異なるスイートを使用する場合にのみ、**server cipher-suite** コマンドを指定します。

ASA 上のすべての SSL サーバー接続に最小 TLS バージョンを設定する場合は、**ssl server-version** コマンドを参照してください。デフォルトは TLS v1.0 です。

例：

```
ciscoasa(config)# tls-proxy diameter-tls-ldc-proxy
ciscoasa(config-tlsp)# server trust-point tls-proxy-server-tp
ciscoasa(config-tlsp)# client ldc issuer ldc-server
ciscoasa(config-tlsp)# client ldc key-pair ldc-signer-key
```

ステップ 4 LDC CA 証明書をエクスポートし、Diameter サーバーにインポートします。

a) 証明書をエクスポートします。

次の例では、LDC トラストポイントは ldc-server です。独自の LDC トラストポイント名を指定します。

```
ciscoasa(config)# crypto ca export ldc-server identity-certificate
-----BEGIN CERTIFICATE-----
MIIDbDCCAlSgAwIBAgIQfWOQvGFpj7hCCB49+kS4CjANBgkqhkiG9w0BAQUFADAT
MREwDwYDVQQDEwhIdW5ueUJlZTAeFw0xMzA2MjUwMTE5MzJaFw00ODA2MjUwMTI5
...[data omitted]...
lJZ48NoI64RqfGC/KHUsOQ==
-----END CERTIFICATE-----
```

b) 証明書データをコピーし、ファイルに保存します。

これで、Diameter サーバーにインポートできます。手順については、Diameter サーバーのドキュメントを参照してください。データは Base64 形式であることに注意してください。

サーバーにバイナリ形式または DER 形式が必要な場合は、OpenSSL ツールを使用して形式を変換する必要があります。

次のタスク

Diameter インスペクションで TLS プロキシを使用できるようになりました。モバイル ネットワーク インスペクションのサービス ポリシーの設定 （42 ページ） を参照してください。

Diameter インスペクション用の TLS オフロードによる TLS プロキシの設定

ASA と Diameter サーバー間のネットワーク パスが安全であると確信している場合は、ASA とサーバー間のデータを暗号化するパフォーマンス コストを回避できます。TLS オフロードを使用すると、TLS プロキシは Diameter クライアントと ASA の間のセッションを暗号化/復号化しますが、Diameter サーバーではクリア テキストを使用します。

この設定では、ASA とクライアント間のみに相互の信頼関係を確立する必要があり、これにより設定が簡略化されます。次の手順を実行する前に、Diameter クライアントとのサーバー信頼関係の設定 （27 ページ） の手順を完了します。

手順

ステップ 1 TLS オフロードに TLS プロキシを設定します。

a) TLS プロキシに名前を付け、TLS プロキシ コンフィギュレーション モードを開始します。

tls-proxy *name*

b) ASA が Diameter クライアントとの関係においてサーバーとして機能するときに使用されるトラストポイントを識別します。

server trust-point *trustpoint_name*

（注） テスト目的の場合、または Diameter クライアントを信頼できると確信している場合は、この手順をスキップして、TLS プロキシ コンフィギュレーションに **no server authenticate-client** コマンドを含めることができます。

c) （任意）サーバーが使用できる暗号方式を定義します。

server cipher-suite *cipher-list*

ここで、*cipher-list* には、次の任意の組み合わせを含めることができます。

- **3des-sha1**
- **aes128-sha1**
- **aes256-sha1**

- **des-sha1**
- **null-sha1**
- **rc4-sha1**

複数のオプションはスペースで区切ります。

TLS プロキシで使用できる暗号方式を定義しないと、プロキシ サーバーは **ssl cipher** コマンドによって定義されたグローバル暗号スイートを使用します。デフォルトでは、グローバル暗号方式レベルはmediumです。つまり、NULL-SHA、DES-CBC-SHA、およびRC4-MD5を除くすべての暗号方式が使用できます。ASAで一般に使用可能なものとは異なるスイートを使用する場合にのみ、**server cipher-suite** コマンドを指定します。

ASA 上のすべての SSL サーバー接続に最小 TLS バージョンを設定する場合は、**ssl server-version** コマンドを参照してください。デフォルトは TLS v1.0 です。

d) ASA と Diameter サーバー間の通信がクリア テキストで行われることを指定します。この中では、ASA は Diameter サーバーのクライアントとして機能します。

client clear-text

例：

```
ciscoasa(config)# tls-proxy diameter-tls-offload-proxy
ciscoasa(config-tlsp)# server trust-point tls-proxy-server-tp
ciscoasa(config-tlsp)# client clear-text
```

ステップ 2 Diameter ポートは TCP と TLS では異なるため、Diameter サーバーからクライアントへのトラフィックに対しては、TCP ポートを TLS ポートに変換する NAT ルールを設定します。

各 Diameter サーバー用のオブジェクト NAT ルールを作成します。各ルールは以下を実行する必要があります。

- Diameter サーバー アドレスにスタティック アイデンティティ NAT を実行します。つまり、オブジェクト内の IP アドレスは、NAT ルール内の変換されたアドレスと同じである必要があります。
- 実際のポート 3868（これはデフォルトの Diameter TCP ポート番号です）を 5868（デフォルトの Diameter TLS ポート番号）に変換します。
- 送信元インターフェイスは、Diameterサーバーに接続しているものでなければならず、宛先インターフェイスは、Diameter クライアントに接続しているものでなければなりません。

次の例では、10.29.29.29 Diameter サーバーから外部インターフェイスに着信するポート 3868 上の TCP トラフィックを内部インターフェイスのポート 5868 に変換します。

```
ciscoasa(config)# object network diameter-client
ciscoasa(config-network-object)# host 10.29.29.29
ciscoasa(config-network-object)# nat (outside,inside) static 10.29.29.29
```

```
service tcp 3868 5868
```

次のタスク

Diameter インスペクションで TLS プロキシを使用できるようになりました。モバイル ネットワーク インスペクションのサービス ポリシーの設定 （42 ページ） を参照してください。

M3UA インスペクション ポリシー マップの設定

M3UA インスペクション ポリシー マップを使用して、ポイント コードに基づくアクセス制御を設定します。また、クラスやタイプ別にメッセージをドロップおよびレート制限できます。

デフォルトのポイント コード形式は ITU です。別の形式を使用している場合は、ポリシー マップで要求される形式を指定します。

ポイント コードまたはメッセージ クラスに基づいてポリシーを適用しない場合は、M3UA ポリシー マップを設定する必要はありません。マップなしでインスペクションを有効にできます。

手順

ステップ 1 M3UA インスペクション ポリシー マップを作成します。**policy-map type inspect m3ua** *policy_map_name*

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。**description** *string*

ステップ 3 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 **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**] **message class** *class_id* [**id** *message_id*]：M3UA メッセージのクラスとタイプを照合します。次の表に、使用可能な値を示します。これらのメッセージの詳細については、M3UA の RFC およびドキュメンテーションを参照してください。

M3UA メッセージ クラス	メッセージ ID タイプ
0（管理メッセージ）	0 ～ 1
1（転送メッセージ）	1
2（SS7 シグナリング ネットワーク管理メッセージ）	1 ～ 6

M3UA メッセージ クラス	メッセージ ID タイプ
3（ASP 状態メンテナンス メッセージ）	1 ～ 6
4（ASP トラフィック メンテナンス メッセージ）	1 ～ 4
9（ルーティング キー管理メッセージ）	1 ～ 4

- **match** [**not**] **opc** *code*：データ メッセージ内の発信ポイント コード、つまりトラフィックの送信元を照合します。ポイント コードは *zone-region-sp* 形式で、各要素に使用可能な値は SS7 バリアントによって異なります。
  - **ITU**：ポイント コードは 3-8-3 形式の 14 ビット値です。値の範囲は、[0-7]-[0-255]-[0-7] です。
  - **ANSI**：ポイント コードは 8-8-8 形式の 24 ビット値です。値の範囲は、[0-255]-[0-255]-[0-255] です。
  - **Japan**：ポイント コードは 5-4-7 形式の 16 ビット値です。値の範囲は、[0-31]-[0-15]-[0-127] です。
  - **China**：ポイント コードは 8-8-8 形式の 24 ビット値です。値の範囲は、[0-255]-[0-255]-[0-255] です。
- **match** [**not**] **dpc** *code*：データ メッセージ内の宛先ポイント コードを照合します。ポイント コードは、**match opc** について説明しているとおり、*zone-region-sp* 形式です。
- **match** [**not**] **service-indicator** *number*：サービス インジケータ番号を照合します（0 ～ 15）。使用可能なサービス インジケータは次のとおりです。これらのサービス インジケータの詳細については、M3UA RFC およびドキュメントを参照してください。
  - 0：シグナリング ネットワーク管理メッセージ
  - 1：シグナリング ネットワーク テストおよびメンテナンス メッセージ
  - 2：シグナリング ネットワーク テストおよびメンテナンス特別メッセージ
  - 3：SCCP
  - 4：電話ユーザー部
  - 5：ISDN ユーザー部
  - 6：データ ユーザー部（コールおよび回線関連のメッセージ）
  - 7：データ ユーザー部（設備の登録およびキャンセル メッセージ）
  - 8：MTP テスト ユーザー部に予約済み
  - 9：ブロードバンド ISDN ユーザー部
  - 10：サテライト ISDN ユーザー部

- 11：予約済み
- 12：AAL タイプ 2 シグナリング
- 13：ベアラー非依存コール制御
- 14：ゲートウェイ制御プロトコル
- 15：予約済み

b) 次のコマンドのいずれかを入力して、一致する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。

- **drop** [**log**]：一致するすべてのパケットをドロップします。任意で、システム ログ メッセージを送信します。
- **rate-limit** *message_rate*：メッセージのレートを制限します。このオプションは **match message class** でのみ使用可能です。

ポリシー マップでは、複数の **match** コマンドを指定できます。match コマンドの順序については、複数のトラフィック クラスの処理方法 を参照してください。

ステップ 4 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。

- **message-tag-validation** {**dupu** | **error** | **notify**}：特定のフィールドの内容が確認され、指定したメッセージ タイプが検証されます。検証で合格しなかったメッセージはドロップされます。検証はメッセージ タイプによって異なります。
  - 利用できない宛先ユーザー部（DUPU）：ユーザー/理由フィールドが存在し、有効な理由およびユーザー コードのみが含まれている必要があります。
  - エラー：すべての必須フィールドが存在し、許可された値のみが含まれている必要があります。各エラー メッセージには、そのエラー コードの必須フィールドが含まれている必要があります。
  - 通知：ステータス タイプおよびステータス情報フィールドには、許可された値のみが含まれている必要があります。
- **ss7 variant** {**ITU** | **ANSI** | **JAPAN** | **CHINA**}：ネットワーク内で使用されている SS7 のバリアントを特定します。このオプションによって、ポイント コードの有効な形式が決定します。オプションを設定して M3UA ポリシーを導入した後は、ポリシーを削除しない限り変更はできません。デフォルトのバリアントは ITU です。

- **strict-asp-state**：アプリケーション サーバー プロセス（ASP）状態の検証を実行します。システムは M3UA セッションの ASP の状態を維持し、検証結果に基づいて ASP メッセージをドロップします。ASP の厳密な状態検証を無効にすると、すべての ASP メッセージが検査されずに転送されます。厳密な ASP のステート チェックが必要なのは、ステートフル フェールオーバーが必要な場合、またはクラスタ内での動作が必要な場合です。ただし、厳密な ASP のステート チェックは、上書きモードでのみ動作し、ロードシェアリングまたはブロードキャスト モードで実行している場合は動作しません（RFC 4666 より）。インスペクションは、エンドポイントごとに ASP が 1 つだけあると仮定します。
- **timeout endpoint** *time*：M3UA エンドポイントの統計情報を削除するアイドル タイムアウトを設定します（hh:mm:ss 形式）。タイムアウトを付けない場合は、0 を指定してください。デフォルトは 30 分（0:30:00）です。
- **timeout session** *time*：厳密な ASP 状態の確認を有効にしている場合の、M3UA セッションを削除するためのアイドル タイムアウト（hh:mm:ss の形式）。タイムアウトを付けない場合は、0 を指定してください。デフォルトは 30 分（0:30:00）です。このタイムアウトを無効にすると、失効したセッションの削除を防止できます。

例

次は、M3UA ポリシー マップおよびサービス ポリシーの例です。

```
hostname(config)# policy-map type inspect m3ua m3ua-map
hostname(config-pmap)# match message class 2 id 6
hostname(config-pmap-c)# drop
hostname(config-pmap-c)# match message class 9
hostname(config-pmap-c)# drop
hostname(config-pmap-c)# match dpc 1-5-1
hostname(config-pmap-c)# drop log
hostname(config-pmap-c)# parameters
hostname(config-pmap-p)# ss7 variant ITU
hostname(config-pmap-p)# timeout endpoint 00:45:00

hostname(config)# policy-map global_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect m3ua m3ua-map

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

次のタスク

マップを使用するためのインスペクション ポリシーを設定できるようになりました。モバイル ネットワーク インスペクションのサービス ポリシーの設定 （42 ページ） を参照してください。

モバイル ネットワーク インスペクションのサービス ポリシーの設定

モバイルネットワークで使用されるプロトコルのインスペクションは、デフォルトのインスペクション ポリシーでは有効になっていないので、これらのインスペクションが必要な場合は有効にする必要があります。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、これらの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 mobile_class_map
hostname(config-cmap)# match access-list mobile
```

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、通過トラフィック用のレイヤ 3/4 クラス マップの作成 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 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 インスペクションをイネーブルにします。

次のコマンドでは、インスペクション ポリシー マップはオプションです。インスペクションをカスタマイズするためにこれらのマップのいずれかを作成した場合は、適切なコマンドで名前を指定します。 Diameterでは、TLSプロキシを指定して、暗号化されたメッセージのインスペクションを有効にすることもできます。

- **inspect gtp** [*map_name*]：GTP インスペクションをイネーブルにします。
- **inspect sctp**[*map_name*]：SCTP インスペクションをイネーブルにします。
- **inspect diameter** [*map_name*] [**tls-proxy** *proxy_name*]：Diameter インスペクションをイネーブルにします。

（注） Diameter インスペクション用の TLS プロキシを指定し、Diameter サーバー トラフィックに NAT ポート リダイレクションを適用した場合（たとえば、ポート 5868 から 3868 にサーバー トラフィックをリダイレクトするなど）は、グローバルに、または入力インターフェイスのみでインスペクションを設定します。出力インターフェイスにインスペクションを適用すると、NATed Diameter トラフィックはインスペクションをバイパスします。

- **inspect m3ua** [*map_name*]：M3UA インスペクションをイネーブルにします。

例：

```
hostname(config-class)# inspect gtp
hostname(config-class)# inspect sctp
hostname(config-class)# inspect diameter
hostname(config-class)# inspect m3ua
```

（注） 別のインスペクション ポリシー マップを使用するためにデフォルト グローバル ポリシー（またはすべての使用中のポリシー）を編集する場合は、コマンドの **no inspect** バージョンを使用してインスペクションを削除してから、新しいインスペクション ポリシー マップの名前で再追加します。たとえば、GTP のポリシー マップを変更するには：

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

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** はポリシーを 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。

RADIUS アカウンティング インスペクションの設定

RADIUSアカウンティングインスペクションはデフォルトではイネーブルになっていません。RADIUS アカウンティング インスペクションが必要な場合は設定してください。

手順

ステップ 1 RADIUS アカウンティング インスペクション ポリシー マップの設定 （44 ページ）。
ステップ 2 RADIUS アカウンティング インスペクションのサービス ポリシーの設定 （46 ページ）。

RADIUS アカウンティング インスペクション ポリシー マップの設定

検査に必要な属性を設定する RADIUS アカウンティング インスペクション ポリシー マップを作成します。

手順

ステップ 1 RADIUS アカウンティング インスペクション ポリシー マップを作成します。**policy-map type inspect radius-accounting** *policy_map_name*

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。**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。

- **send response**：Accounting-Request の Start および Stop メッセージを、それらのメッセージの送信元（**host** コマンド内で識別されています）へ送信するよう ASA に指示します。
- **enable gprs**：GPRS 過剰請求の保護を実装します。セカンダリ PDP コンテキストを適切に処理するため、ASA は、Accounting-Request の Stop および Disconnect メッセージの 3GPP VSA 26-10415 属性をチェックします。この属性が存在する場合、ASA は、設定インターフェイスのユーザー IP アドレスに一致するソース IP を持つすべての接続を切断します。
- **validate-attribute** *number*：Accounting-Request Start メッセージを受信する際、ユーザー アカウントのテーブルを作成する場合に使用する追加基準。これらの属性は、ASA が接続を切断するかどうかを決定する場合に役立ちます。

  検証する追加属性を指定しない場合は、Framed IP アドレス属性の IP アドレスのみに基づいて決定されます。追加属性を設定し、ASA が現在追跡されているアドレスを含むが、その他の検証する属性が異なるアカウンティング開始メッセージを受信すると、古い属性を使用して開始するすべての接続は、IP アドレスが新しいユーザーに再割り当てされたという前提で、切断されます。

  値の範囲は 1 ～ 191 で、このコマンドは複数回入力できます。属性番号および説明のリストについては、http://www.iana.org/assignments/radius-types を参照してください。
- **host** *ip_address* [**key** *secret*]：RADIUS サーバーまたは GGSN の IP アドレスです。ASA がメッセージを許可できるよう、任意で秘密キーを含めることができます。キーがない場合、IP アドレスだけがチェックされます。複数の RADIUS と GGSN のホストを識別するため、このコマンドは繰り返し実行できます。ASA は、これらのホストから RADIUS アカウンティング メッセージのコピーを受信します。
- **timeout users** *time*：ユーザーのアイドル タイムアウトを設定します（hh: mm: ss 形式）。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。デフォルトは 1 時間です。

例

```
policy-map type inspect radius-accounting radius-acct-pmap
  parameters
    send response
    enable gprs
    validate-attribute 31
    host 10.2.2.2 key 123456789
    host 10.1.1.1 key 12345
class-map type management radius-class
  match port udp eq radius-acct
```

```
policy-map global_policy
  class radius-class
    inspect radius-accounting radius-acct-pmap
```

RADIUS アカウンティング インスペクションのサービス ポリシーの設定

デフォルトのインスペクション ポリシーでは、RADIUS アカウンティング インスペクションはイネーブルにされてないため、この検査が必要な場合はイネーブルにします。RADIUS アカウンティング インスペクションは ASA のトラフィック用に指示されますので、標準ルールではなく、管理インスペクション ルールとして設定してください。

手順

ステップ 1 検査を適用するトラフィックを識別するため L3/L4 マネジメント クラス マップを作成し、一致するトラフィックを識別します。

```
class-map type management name
match {port | access-list} parameter
```

例：

```
hostname(config)# class-map type management radius-class-map
hostname(config-cmap)# match port udp eq radius-acct
```

この例では、一致は radius acct UDP ポート（1646）です。ポートの範囲（**match port udp range** *number1 number2*）または **match access-list** *acl_name* と ACL を使って異なるポートを指定できます。

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 RADIUS アカウンティング インスペクションに使用する L3/L4 管理クラス マップを特定します。**class** *name*

例：

```
hostname(config-pmap)# class radius-class-map
```

ステップ 4 RADIUS アカウンティング インスペクションを設定します。**inspect radius-accounting**[*radius-accounting_policy_map*]

radius_accounting_policy_map はRADIUS アカウンティング インスペクション ポリシー マップの設定 （44 ページ）で作成した RADIUS アカウンティング インスペクション ポリシー マップです。

例：

```
hostname(config-class)# no inspect radius-accounting
hostname(config-class)# inspect radius-accounting radius-class-map
```

（注） 別のインスペクション ポリシー マップを使用するために使用中のポリシーを編集する場合、**no inspect radius-accounting** コマンドで RADIUS アカウンティング インスペクションを削除してから、新しいインスペクション ポリシー マップの名前で再追加します。

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** はポリシーを 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。

モバイル ネットワーク インスペクションのモニタリング

ここでは、モバイル ネットワーク インスペクションをモニタリングする方法について説明します。

GTP インスペクションのモニタリング

GTP コンフィギュレーションを表示するには、特権 EXEC モードで show service-policy inspect gtp コマンドを入力します。

show service-policy inspect gtp statistics コマンドを使用して、GTP インスペクションの統計情報を表示します。次にサンプル出力を示します。

```
firewall(config)# show service-policy inspect gtp statistics
GPRS GTP Statistics:
  version_not_support               0     msg_too_short             0
  unknown_msg                       0     unexpected_sig_msg        0
  unexpected_data_msg               0     ie_duplicated             0
  mandatory_ie_missing              0     mandatory_ie_incorrect    0
  optional_ie_incorrect             0     ie_unknown                0
  ie_out_of_order                   0     ie_unexpected             0
  total_forwarded                  67     total_dropped             1
  signalling_msg_dropped            1     data_msg_dropped          0
  signalling_msg_forwarded         67     data_msg_forwarded        0
  total created_pdp                33     total deleted_pdp         32
  total created_pdpmcb             31     total deleted_pdpmcb      30
  total dup_sig_mcbinfo             0     total dup_data_mcbinfo    0
  no_new_sgw_sig_mcbinfo            0     no_new_sgw_data_mcbinfo   0
  pdp_non_existent                  1
```

show service-policy inspect gtp statistics *ip_address* コマンドに IP アドレスを入力すると、特定の GTP エンドポイントの統計情報を取得できます。

```
firewall(config)# show service-policy inspect gtp statistics 10.9.9.9
1 in use, 1 most used, timeout 0:30:00
 GTP GSN Statistics for 10.9.9.9, Idle 0:00:34, restart counter 0
  Tunnels Active                    0
  Tunnels Created                   1
  Tunnels Destroyed                 0
  Total Messages Received           1
                         Signalling Messages        Data Messages
  total received                   1                          0
  dropped                          0                          0
  forwarded                        1                          0
```

show service-policy inspect gtp pdp-context コマンドを使用して、PDP コンテキストに関する情報を表示します。GTPv2 の場合、これはベアラー コンテキストです。次に例を示します。

```
ciscoasa(config)# show service-policy inspect gtp pdp-context
4 in use, 5 most used

Version v1,   TID 050542012151705f,  MS Addr 2005:a00::250:56ff:fe96:eec,
SGSN Addr 10.0.203.22,      Idle 0:52:01,   Timeout 3:00:00,   APN ssenoauth146

Version v2,   TID 0505420121517056,  MS Addr 100.100.100.102,
SGW Addr 10.0.203.24,      Idle 0:00:05,   Timeout 3:00:00,   APN ssenoauth146

Version v2,   TID 0505420121517057,  MS Addr 100.100.100.103,
SGW Addr 10.0.203.25,      Idle 0:00:04,   Timeout 3:00:00,   APN ssenoauth146

Version v2,   TID 0505420121517055,  MS Addr 100.100.100.101,
SGW Addr 10.0.203.23,      Idle 0:00:06,   Timeout 3:00:00,   APN ssenoauth146

ciscoasa(config)# show service-policy inspect gtp pdp-context detail
1 in use, 1 most used

Version v1,   TID 050542012151705f,  MS Addr 2005:a00::250:56ff:fe96:eec,
SGSN Addr 10.0.203.22,      Idle 0:06:14,   Timeout 3:00:00,   APN ssenoauth146

    user_name (IMSI):  50502410121507    MS address: 2005:a00::250:56ff:fe96:eec
    nsapi: 5                linked nsapi: 5
    primary pdp: Y         sgsn is Remote
```

```
    sgsn_addr_signal: 10.0.203.22    sgsn_addr_data: 10.0.203.22
    ggsn_addr_signal: 10.0.202.22    ggsn_addr_data: 10.0.202.22
    sgsn control teid:     0x00000001    sgsn data teid:        0x000003e8
    ggsn control teid:     0x000f4240    ggsn data teid:        0x001e8480
    signal_sequence:               18     state:    Ready
...
```

PDP またはベアラー コンテキストは、IMSI と NSAPI（GTPv0-1）または IMSI と EBI（GTPv2）の値の組み合わせであるトンネル ID（TID）によって識別されます。GTP トンネルは、それぞれ別個の GSN または SGW/PGW ノードにある、2 つの関連するコンテキストによって定義され、トンネル ID によって識別されます。GTP トンネルは、外部パケット データ ネットワークとモバイル サブスクライバ（MS）ユーザーとの間でパケットを転送する場合に必要です。

SCTP のモニタリング

次のコマンドを使用して、SCTP をモニターできます。

- **show service-policy inspect sctp**

  SCTP インスペクションの統計情報を表示します。sctp-drop-override カウンタは、PPID がドロップ アクションに一致するたびに増加しますが、パケットには PPID が異なるデータのかたまりが含まれていたのでパケットはドロップされません。次に例を示します。

```
ciscoasa# show service-policy inspect sctp
Global policy:
  Service-policy: global_policy
    Class-map: inspection_default
      Inspect: sctp sctp, packet 153302, lock fail 0, drop 20665, reset-drop 0,
5-min-pkt-rate 0 pkts/sec, v6-fail-close 0, sctp-drop-override 4910
        Match ppid 30 35
           rate-limit 1000 kbps, chunk 2354, dropped 10, bytes 21408, dropped-bytes
 958
        Match: ppid 40
           drop, chunk 5849
        Match: ppid 55
           log, chunk 9546
```

- **show sctp** [**detail**]

  現在の SCTP Cookie およびアソシエーションを表示します。SCTP アソシエーションに関する詳細情報を表示するには、**detail** キーワードを追加します。詳細ビューには、マルチホーミング、複数のストリーム、およびフラグメント再構成に関する情報も表示されます。

```
ciscoasa# show sctp

AssocID: 71adeb15
Local:  192.168.107.12/50001 (ESTABLISHED)
Remote: 192.168.108.122/2905 (ESTABLISHED)
Secondary Conn List:
    192.168.108.12(192.168.108.12):2905 to 192.168.107.122(192.168.107.122):50001
    192.168.107.122(192.168.107.122):50001 to 192.168.108.12(192.168.108.12):2905
    192.168.108.122(192.168.108.122):2905 to 192.168.107.122(192.168.107.122):50001

    192.168.107.122(192.168.107.122):50001 to 192.168.108.122(192.168.108.122):2905
```

```
192.168.108.12(192.168.108.12):2905 to 192.168.107.12(192.168.107.12):50001
192.168.107.12(192.168.107.12):50001 to 192.168.108.12(192.168.108.12):2905
```

- **show conn protocol sctp**

  現在の SCTP 接続に関する情報を表示します。

- **show local-host** [**connection sctp** *start*[*-end*]]

  インターフェイスごとに、ASA を経由して SCTP 接続を行うホストに関する情報を表示します。特定の数または範囲の SCTP 接続を持つホストのみを表示するには、**connection sctp** キーワードを追加します。

- **show traffic**

  sysopt traffic detailed-statistics コマンドを有効にしている場合は、インターフェイスごとの SCTP 接続とインスペクションの統計情報が表示されます。

Diameter のモニタリング

次のコマンドを使用して、Diameter をモニターできます。

- **show service-policy inspect diameter**

  Diameter インスペクションの統計情報を表示します。次に例を示します。

```
ciscoasa# show service-policy inspect diameter
Global policy:
  Service-policy: global_policy
    Class-map: inspection_default
      Inspect: Diameter Diameter_map, packet 0, lock fail 0, drop 0, -drop 0,
5-min-pkt-rate 0 pkts/sec, v6-fail-close 0
       Class-map: log_app
           Log: 5849
        Class-map: block_ip
            drop-connection: 2
```

- **show diameter**

  各 Diameter 接続のステータス情報を表示します。次に例を示します。

```
ciscoasa# show diameter
Total active diameter sessions: 5
Session 3638
       ==========
       ref_count: 1 val = .; 1096298391; 2461;
           Protocol : diameter Context id : 0
           From inside:211.1.1.10/45169 to outside:212.1.1.10/3868
...
```

- **show conn detail**

  接続情報を表示します。Diameter 接続は、Q フラグを使用してマークされます。

- **show tls-proxy**

  TLS プロキシを Diameter インスペクションで使用する場合は、そのプロキシに関する情報が表示されます。

M3UA のモニタリング

次のコマンドを使用して、M3UA をモニターできます。

- **show service-policy inspect m3ua drops**

  M3UA インスペクションに対するドロップの統計情報を表示します。

- **show service-policy inspect m3ua endpoint** [*IP_address*]

  M3UA エンドポイントの統計情報を表示します。エンドポイントの IP アドレスを指定して、特定のエンドポイントに関する情報を表示できます。ハイアベイラビリティまたはクラスタ化されたシステムでは、統計情報はユニットごとに提供され、ユニット間で同期されません。次に例を示します。

```
ciscoasa# sh service-policy inspect m3ua endpoint
 M3UA Endpoint Statistics for 10.0.0.100, Idle : 0:00:06 :
                    Forwarded        Dropped          Total Received
 All Messages       21                5                 26
 DATA Messages      9                 5                 14
 M3UA Endpoint Statistics for 10.0.0.110, Idle : 0:00:06 :
                    Forwarded        Dropped          Total Received
 All Messages       21                8                 29
 DATA Messages      9                 8                 17
```

- **show service-policy inspect m3ua session**

  厳密なアプリケーション サーバー プロセス（ASP）状態の確認を有効にすると、M3UA セッションに関する情報が表示されます。情報には、送信元アソシエーション ID、セッションがシングルまたはダブルいずれの交換であるか、また、クラスタの場合はクラスタ オーナーセッションとバックアップセッションのいずれであるかが含まれます。3 つ以上のユニットを持つクラスタでは、ユニットがクラスタから抜けた後に戻って来る場合、古いバックアップ セッションが表示されることがあります。これらの古いセッションは、セッション タイムアウトを無効にしていなければ、タイムアウト時に削除されます。

```
Ciscoasa# show service-policy inspect m3ua session
0 in use, 0 most used
Flags: o - cluster owner session, b - cluster backup session
       d - double exchange      , s - single exchange
AssocID: cfc59fbe in Down state, idle:0:00:05, timeout:0:01:00, bd
AssocID: dac2e123 in Active state, idle:0:00:18, timeout:0:01:00, os
```

- **show service-policy inspect m3ua table**

  分類ルールを含むランタイム M3UA インスペクション テーブルを表示します。

- **show conn detail**

接続情報を表示します。M3UA 接続は、v フラグを使用してマークされます。

モバイル ネットワーク インスペクションの履歴

機能名	リリース	機能情報
GTPv2 インスペクションと GTPv0/1 インスペクションの改善	9.5(1)	GTP インスペクションは GTPv2 を処理できるようになりました。また、すべてのバージョンの GTP インスペクションで IPv6 アドレスがサポートされるようになりました。 **match message id** コマンドが **match message** {**v1** | **v2**} **id** *message_id* に変更されました。**timeout gsn** コマンドが **timeout endpoint** に置き換えられました。**clear/show service-policy inspect gtp statistics** コマンドから **gsn** キーワードが削除され、エンドポイント ID を入力するだけでこれらの統計情報を確認またはクリアできるようになりました。**clear/show service-policy inspect gtp request** および **pdpmcb** コマンドに **version** キーワードが追加され、特定の GTP バージョンに関する情報を表示できるようになりました。
SCTP インスペクション	9.5(2)	ペイロード プロトコル ID（PPID）に基づいてアクションを適用するために、アプリケーション層インスペクションを Stream Control Transmission Protocol（SCTP）トラフィックに適用できるようになりました。 **clear conn protocol sctp**、**inspect sctp**、**match ppid**、**policy-map type inspect sctp**、**show conn protocol sctp**、**show local-host connection sctp**、**show service-policy inspect sctp** の各コマンドが追加または変更されました。
Diameter インスペクション	9.5(2)	アプリケーション層インスペクションを Diameter トラフィックに適用できるようになり、アプリケーション ID、コマンドコード、および属性値ペア（AVP）のフィルタリングに基づいてアクションを適用できるようになりました。 **class-map type inspect diameter**、**diameter**、**inspect diameter**、**match application-id**、**match avp**、**match command-code**、**policy-map type inspect diameter**、**show conn detail**、**show diameter**、**show service-policy inspect diameter**、**unsupported** の各コマンドが追加または変更されました。

機能名	リリース	機能情報
Diameter インスペクションの改善	9.6(1)	TCP/TLS トラフィック上の Diameter を検査し、厳密なプロトコル準拠チェックを適用し、クラスタ モードで SCTP 上の Diameter を検査できるようになりました。 **client clear-text**、**inspect diameter**、**strict-diameter** の各コマンドが追加または変更されました。
クラスタ モードでの SCTP ステートフル インスペクション	9.6(1)	SCTP ステートフル インスペクションがクラスタ モードで動作するようになりました。また、クラスタ モードで SCTP ステートフル インスペクション バイパスを設定することもできます。 導入または変更されたコマンドはありません。
MTP3 User Adaptation（M3UA）インスペクション。	9.6(2)	M3UA トラフィックを検査できるようになりました。また、ポイント コード、サービス インジケータ、およびメッセージのクラスとタイプに基づいてアクションを適用できるようになりました。 **clear service-policy inspect m3ua** {**drops** | **endpoint** [*IP_address*]}、**inspect m3ua**、**match dpc**、**match opc**、**match service-indicator**、**policy-map type inspect m3ua**、**show asp table classify domain inspect-m3ua**、**show conn detail**、**show service-policy inspect m3ua** {**drops** | **endpoint** [*IP_address*]}、**ss7 variant**、**timeout endpoint** の各コマンドが追加または変更されました。
SCTP マルチストリーミングの並べ替えとリアセンブル、およびフラグメンテーションのサポート。SCTP エンドポイントに複数の IP アドレスが設定された SCTP マルチホーミングのサポート。	9.7(1)	このシステムは、SCTP マルチストリーミングの並べ替え、リアセンブル、およびフラグメンテーションを完全にサポートしており、これにより SCTP トラフィックに対する Diameter および M3UA インスペクションの有効性が改善されています。このシステムは、各エンドポイントに複数の IP アドレスが設定された SCTP マルチホーミングもサポートしています。マルチホーミングでは、セカンデリ アドレスに必要なピンホールをシステムが開くので、セカンデリ アドレスを許可するためのアクセス ルールをユーザーが設定する必要はありません。SCTP エンドポイントは、それぞれ 3 つの IP アドレスに制限する必要があります。 **show sctp detail** コマンドの出力が変更されました。

機能名	リリース	機能情報
M3UA インスペクションの改善。	9.7(1)	M3UAインスペクションは、ステートフルフェールオーバー、半分散クラスタリング、およびマルチホーミングをサポートするようになりました。また、アプリケーション サーバー プロセス（ASP）の状態の厳密な検証や、さまざまなメッセージの検証も設定できます。ASP状態の厳密な検証は、ステートフルフェールオーバーとクラスタリングに必要です。 次のコマンドが追加または変更されました。**clear service-policy inspect m3ua session** [**assocID** *id*]、**match port sctp**、**message-tag-validation**、**show service-policy inspect m3ua drop**、**show service-policy inspect m3ua endpoint**、**show service-policy inspect m3ua session**、**show service-policy inspect m3ua table**、**strict-asp-state**、**timeout session**。
TLS プロキシ サーバーの SSL 暗号スイートの設定サポート	9.8(1)	ASA が TLS プロキシ サーバーとして動作している場合は、SSL 暗号スイートを設定できるようになりました。以前は、 **ssl cipher** コマンドを使用した ASA のグローバル設定のみが可能でした。 次のコマンドが導入されました。 **server cipher-suite**
MSISDN および選択モードのフィルタリング、アンチリプレイ、およびユーザー スプーフィング保護に対する GTP インスペクションの機能拡張。	9.10(1)	モバイル ステーション国際サブスクライバ電話番号（MSISDN）または選択モードに基づいて PDP コンテキストの作成メッセージをドロップするようにGTPインスペクションを設定できるようになりました。また、アンチリプレイとユーザー スプーフィング保護も実装できます。 **anti-replay**、**gtp-u-header-check**、**match msisdn**、**match selection-mode** の各コマンドが追加されました。
GTPv1 リリース 10.12 のサポート	9.12(1)	システムで GTPv1 リリース 10.12 がサポートされるようになりました。以前は、リリース 6.1 がサポートされていました。新しいサポートでは、25 件の GTPv1 メッセージおよび 66 件の情報要素の認識が追加されています。 さらに、動作の変更もあります。不明なメッセージ ID が許可されるようになりました。以前は、不明なメッセージはドロップされ、ログに記録されていました。 追加または変更されたコマンドはありません。

機能名	リリース	機能情報
モバイル端末の場所のロギング（GTPインスペクション）。	9.13(1)	GTPインスペクションを設定すると、モバイル端末の初期の場所とそれ以降の場所の変更をログに記録できます。場所の変更を追跡すると、不正なローミング請求を識別するのに役立つ場合があります。 **location-logging** コマンドが追加されました。
GTPv2 および GTPv1 リリース 15 がサポートされています。	9.13(1)	システムで GTPv2 3GPP 29.274 V15.5.0 がサポートされるようになりました。GTPv1 の場合、3GPP 29.060 V15.2.0 までサポートしています。新しいサポートでは、2 件のメッセージおよび 53 件の情報要素の認識が追加されています。 追加または変更されたコマンドはありません。
GTP インスペクションでドロップされる IMSI プレフィックスを指定する機能です。	9.16(1)	GTP インスペクションでは、許可する Mobile Country Code/Mobile Network Code（MCC/MNC）の組み合わせを識別するために、IMSI プレフィックス フィルタリングを設定できます。ドロップする MCC/MNC の組み合わせに対して IMSI フィルタリングを実行できるようになりました。これにより、望ましくない組み合わせをリストにして、デフォルトで他のすべての組み合わせを許可することができます。 **drop mcc** コマンドが追加されました。
キャリアライセンスの Secure Firewall 3100 サポート	9.18(1)	キャリアライセンスは、Diameter、GTP/GPRS、SCTP 検査を有効にします。 新規/変更されたコマンド： **feature carrier**

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。