Cisco ASR 9000 シリーズ アグリゲーション サービス ルータ システム セキュリティ コンフィギュレーション ガイド リリース 4.2

シスコシステムズ合同会社
〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー
http://www.cisco.com/jp
お問い合わせ先：シスコ コンタクトセンター
0120-092-255 （フリーコール、携帯・PHS含む）
電話受付時間：平日 10:00～12:00、13:00～17:00
http://www.cisco.com/jp/go/contactcenter/

Text Part Number: OL-26047-01-J

【注意】シスコ製品をご使用になる前に、安全上の注意（**www.cisco.com/jp/go/safety_warning/**）をご確認ください。本書は、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。また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

目 次

はじめに

このガイドでは、システム セキュリティの設定および例について説明します。 システム セキュリティ コマンドの説明、使用方法、タスク ID および例については、『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』を参照してください。

では、次のトピックについて取り上げます。

- マニュアルの変更履歴，xiii ページ
- マニュアルの入手方法およびテクニカル サポート，xiii ページ

マニュアルの変更履歴

次の表に、初版後、本書に行われた変更の履歴を示します。

表 *1*：マニュアルの変更履歴

リビジョン	日付	変更点
OL-26047-01	2011 年 12 月	このマニュアルの初版

マニュアルの入手方法およびテクニカル サポート

マニュアルの入手方法、テクニカル サポート、その他の有用な情報について、次の URL で、毎月更新される『*What's New in Cisco Product Documentation*』を参照してください。シスコの新規および改訂版の技術マニュアルの一覧も示されています。

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

『*What's New in Cisco Product Documentation*』は RSS フィードとして購読できます。また、リーダー アプリケーションを使用してコンテンツがデスクトップに直接配信されるように設定することもできます。 RSS フィードは無料のサービスです。シスコは現在、RSS バージョン 2.0 をサポートしています。

第 1 章

AAA サービスの設定

このモジュールでは、Cisco IOS XR ソフトウェア システムでのユーザ アクセスの制御に使用されるタスクベース認可の管理モデルの実装について説明します。 タスクベース認可の実装では、主にユーザ グループおよびタスク グループを設定する必要があります。

ユーザ グループおよびタスク グループは、認証、認可およびアカウンティング（AAA）サービスに使用される Cisco IOS XR ソフトウェア コマンド セットを介して設定されます。 認証コマンドは、ユーザまたはプリンシパルの ID の検証に使用されます。 認可コマンドは、認証ユーザ（またはプリンシパル）に特定のタスクを実行する権限があるか確認するときに使用されます。 アカウンティング コマンドは、セッションのログイン、および特定のユーザまたはシステムにより生成されるアクションを記録することで監査証跡を作成するときに使用されます。

AAA は、Cisco IOS XR ソフトウェア ベース パッケージの一部で、デフォルトで使用可能です。

（注） このモジュールで使用される AAA コマンドの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』の「*Authentication, Authorization, and Accounting Commands on* Cisco ASR 9000 シリーズ ルータ」モジュールを参照してください。 この章で使用される他のコマンドの説明については、コマンド リファレンスのマスター インデックスを参照するか、またはオンラインで検索してください。

AAA サービス設定の機能履歴

リリース	変更点
リリース 3.7.2	この機能を追加しました。
リリース 4.1.0	VRF 対応 TACACS+ のサポートが追加されました。

AAA サービスの設定に関する前提条件

次に、AAA サービスの設定に関する前提条件を示します。

- 適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。 ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。
- 初期設定ダイアログを使用してルート システム ユーザを確立します。 管理者は、特定の AAA 設定なしでいくつかのローカル ユーザを設定できます。 外部セキュリティ サーバは、ユーザ アカウントが管理ドメイン内の複数のルータで共有される場合に必要になります。一般的な設定では、外部サーバが到達不能になった場合のバックアップとしてローカル データベース オプションを使用して、外部 AAA セキュリティ サーバおよびデータベースを使用します。

AAA サービスの設定に関する制約事項

この項では、AAA サービスの設定に関する制約事項をリストします。

互換性

互換性は、Cisco フリーウェア TACACS+ サーバおよび FreeRADIUS のみで検証されています。

相互運用性

ルータ管理者は、

ルータおよび Cisco IOS XR ソフトウェア で実行されていないその他のシスコ デバイスに対して同じ AAA サーバ ソフトウェアとデータベース（たとえば、CiscoSecure ACS）を使用できます。ルータとタスク ID をサポートしていない外部 TACACS+ サーバ間の相互運用性をサポートするには、TACACS+ および RADIUS 認証ユーザのタスク ID, （14 ページ）の項を参照してください。

AAA サービスの設定について

この項では、Cisco IOS XR ソフトウェア ユーザが AAA でユーザ グループおよびタスク グループを設定する前、または Remote Authentication Dial-in User Service（RADIUS）や TACACS+ サーバ

を設定する前に理解しておく必要があるすべての概念情報をリストします。概念情報では、AAA について、およびなぜ重要なのかについても説明します。

ユーザ、ユーザグループおよびタスクグループ

Cisco IOS XR ソフトウェア ユーザ属性は、Cisco IOS XR ソフトウェア 管理モデルの基礎を形成します。 各ルータ ユーザには、次の属性が関連付けられます。

- 管理ドメイン内でユーザを一意に特定するユーザ ID（ASCII 文字列）
- 253 文字以下のパスワードおよび一方向の暗号化シークレット
- ユーザがメンバである（タスク ID などの属性をイネーブルにした）ユーザ グループ（1 つ以上）のリスト （タスク ID, （13 ページ）の項を参照）。

ユーザ カテゴリ

ルータ ユーザは、次のカテゴリに分類されます。

- ルート システム ユーザ（すべての管理権限）
- ルート SDR ユーザ（特定のセキュア ドメイン ルータ管理権限）
- セキュア ドメイン ルータ ユーザ（特定のセキュア ドメイン ルータ ユーザ アクセス）

ルート システム ユーザ

ルート システム ユーザは、ルータ シャーシ全体の「所有」が許可されたエンティティです。ルート システム ユーザは、すべてのルータ コンポーネントで最高の権限を持ち、システムのすべてのセキュア ドメイン ルータをモニタできます。 ルート システム ユーザ アカウントは、ルータ設定中に少なくとも 1 つ作成する必要があります。 ルート システム ユーザは複数作成できます。

ルート システム ユーザは、次のようなタスクの設定またはモニタリングを実行できます。

- セキュア ドメイン ルータを設定します。
- ルート SDR ユーザを作成、削除および変更します（セキュア ドメイン ルータにルート システム ユーザとしてログインした後） （ルート SDR ユーザ, （4 ページ）の項を参照）。
- セキュア ドメイン ルータ ユーザを作成、削除、変更し、ユーザ タスク権限を設定します（セキュア ドメイン ルータにルート システム ユーザとしてログインした後） （セキュア ドメイン ルータ ユーザ, （4 ページ）の項を参照）。
- セキュア ドメイン ルータに割り当てられていないファブリック ラックまたは任意のルータ リソースにアクセスします。これにより、セキュア ドメイン ルータの設定に関係なくルート システム ユーザが任意のルータ ノードに対して認証できます。

ルート SDR ユーザ

ルート SDR ユーザは、特定の SDR の設定およびモニタリングを制御します。 ルート SDR ユーザは、ユーザを作成し、SDR 内での権限を設定できます。 複数のルート SDR ユーザが独立して作業できます。 1 つの SDR に、複数の SDR ユーザを作成できます。

ルート SDR ユーザは、特定の SDR に対して次の管理タスクを実行できます。

- SDR のセキュア ドメイン ルータ ユーザおよび権限を作成、削除、変更します（セキュア ドメイン ルータ ユーザ, （4 ページ）の項を参照）。
- SDR にアクセスできるユーザ グループを作成、削除、変更します。
- SDR のほぼすべてを管理します。

ルート SDR ユーザは、ルート システム ユーザへのアクセスを拒否できません（ルート システム ユーザ, （3 ページ）の項を参照）。

セキュア ドメイン ルータ ユーザ

セキュア ドメイン ルータ ユーザには、ルート システム ユーザまたはルート SDR ユーザにより定義されている SDR への制限付きアクセス権があります。 セキュア ドメイン ルータ ユーザは、日常のシステムおよびネットワーク管理業務を行います。 セキュア ドメイン ルータ ユーザが実行できるタスクは、セキュア ドメイン ルータ ユーザが属するユーザ グループに関連付けられているタスク ID により決まります（ユーザ グループ, （4 ページ）の項を参照）。

ユーザ グループ

Cisco IOS XR ソフトウェアでは、システム管理者は、ユーザのグループ、およびユーザのグループで共通するジョブ特性を設定できます。 グループは、明示的にユーザに割り当てる必要があります。 ユーザは、デフォルトでは、グループに割り当てられていません。 ユーザは、複数のグループに割り当てることができます。

ユーザ グループは、アクセス権限など、属性のセットを共有するユーザの集まりです。各ユーザは、1 つ以上のユーザ グループに関連付けることができます。 ユーザ グループは、次の属性を持ちます。

- ユーザの認可を定義するタスク グループのリスト。 cisco-support 以外のすべてのタスクは、デフォルトで、ルート システム ユーザに許可されています（ルート システム ユーザ, （3 ページ）の項を参照）。
- 各ユーザ タスクには、読み取り、書き込み、実行またはデバッグ権限を割り当てることができます。

事前定義ユーザ グループ

Cisco IOS XR ソフトウェアには、属性を定義済みの一連のユーザ グループが用意されています。事前定義されているグループは次のとおりです。

- **cisco-support**：このグループは、Cisco サポート チームが使用します。
- **netadmin**：すべてのシステムおよびネットワーク パラメータを制御およびモニタできます。
- **operator**：基本権限を持つデモンストレーション グループ。
- **root-lr**：特定のセキュア ドメイン ルータを制御およびモニタできます。
- **root-system**：システム全体を制御およびモニタできます。
- **sysadmin**：すべてのシステム パラメータを制御およびモニタできますが、ネットワーク プロトコルを設定できません。
- **serviceadmin**：セッション ボーダー コントローラ（SBC）などのサービス管理タスク。

ユーザ グループ root-system には、唯一のメンバとしてルート システム ユーザが含まれます（ルート システム ユーザ, （3 ページ）の項を参照）。root-system ユーザ グループには認可が事前に定義されています。つまり、root-system ユーザ管理リソースのすべて、および他の SDR の一部を担当します。

ユーザ定義ユーザ グループ

管理者は、特定のニーズに合わせて、独自のユーザ グループを設定できます。

ユーザ グループ継承

ユーザ グループは、別のユーザ グループから属性を継承できます（同様に、タスク グループは、別のタスク グループから属性を継承できます）。 たとえば、ユーザ グループ A がユーザ グループ B から属性を継承すると、ユーザ グループ A のタスク属性の新しいセットは、A と B の属性の集合になります。 グループ A がグループ B から属性を継承した場合、グループ B で変更を行うと、明示的に再継承しなくても、その変更がグループ A にも影響を与えるため、ユーザ グループでの継承関係は動的といえます。

タスク グループ

タスク グループは、タスク ID の集合によって定義されます。 タスク グループには、各アクション クラスに対応したタスク ID リストが含まれます。

各ユーザ グループは、そのグループのユーザに適用できる一連のタスク グループが関連付けられます。 ユーザのタスク許可は、そのユーザが属するユーザ グループに関連付けられたタスク グループから継承されます。

事前定義タスク グループ

次に、管理者が通常の初期設定で使用できる事前定義タスク グループを示します。

- **cisco-support**：Cisco サポート担当タスク
- **netadmin**：ネットワーク管理者タスク

- **operator**：オペレータの日常業務（デモンストレーション目的）
- **root-lr**：セキュア ドメイン ルータ管理者タスク
- **root-system**：システム規模の管理者タスク
- **sysadmin**：システム管理者タスク
- **serviceadmin**：SBC などのサービス管理タスク

ユーザ定義タスク グループ

ユーザは、特定のニーズに合わせて、独自のタスク グループを設定できます。

グループ継承

タスク グループは、他のタスク グループからの継承をサポートします（同様に、ユーザ グループは、別のユーザ グループから属性を継承できます。ユーザ グループ，（4 ページ）の項を参照してください）。たとえば、タスク グループ A がタスク グループ B から継承すると、タスク グループ A の属性の新しいセットは、A と B の集合になります。

Cisco IOS XR ソフトウェア管理モデル

ルータは、管理（admin）プレーンとセキュア ドメイン ルータ（SDR）プレーンの 2 つのプレーンで機能します。 admin（共有）プレーンは、すべての SDR で共有されるリソースで構成され、SDR プレーンは、特定の SDR に固有なリソースで構成されます。

root-system ユーザには、ルータの最高レベルの権限があります。 このユーザは、セキュア ドメイン ルータをプロビジョニングし、ルート SDR ユーザを作成します。 作成すると、ルート SDR ユーザは、SDR の root-system ユーザの権限を利用します。 ルート SDR ユーザは、セキュア ドメイン ルータ ユーザを作成できます。 root-system ユーザおよびルート SDR ユーザには、ユーザが変更できない固定権限（タスク ID）があります。

各 SDR には、ローカル ユーザ、グループ、TACACS+ および RADIUS 設定など、独自の AAA 設定があります。 SDR で作成されたユーザは、同じユーザが他の SDR で設定されていない限り、他の SDR にアクセスできません。

管理アクセス

システムへの管理アクセスは、次の操作を十分理解していない場合、または注意して計画していない場合、失われる可能性があります。 すべての root-system ユーザのロックアウトは、パスワードの回復のためにシステム リロードが必要になる重大な問題です。

- 使用できないリモート AAA サーバを使用する認証（特にコンソールの認証）を設定する。

（注） 他の方式リストを指定しない **none** オプションの使用は、Cisco IOS XR ソフトウェアではサポートされていません。

- フラッシュ カードを disk0: から削除する、またはディスクが破損すると、補助ポート認証が拒否されることがあります。これにより、特定のシステムデバッグ機能に影響を与えることがあります。 ただし、コンソールを使用できる場合、システムにアクセスできます。
- コンソールでコマンド認可または EXEC 認可を設定する場合は十分に注意してください。これは、この設定により TACACS+ サーバが使用できなくなる、またはすべてのコマンドが拒否され、ユーザがロックアウトされる場合があるためです。 このロックアウトは、特に、TACACS+ サーバで認識されていないユーザで認証が行われる場合、あるいは TACACS+ ユーザで何らかの理由によりほとんど、またはすべてのコマンドが拒否される場合に発生します。

ロックアウトを回避するには、次のいずれか、または両方を推奨します。

- コンソールで TACACS+ コマンド認可または EXEC 認可を設定する前に、認可を設定するユーザが、TACACS+ プロファイルの適切なユーザ権限を使用してログインしていることを確認してください。
- サイトのセキュリティ ポリシーで許可されている場合、**none** オプションをコマンド認可または EXEC 認可に使用します。これにより、TACACS+ サーバが使用できない場合、AAA は **none** 方式にロールオーバーし、ユーザはコマンドを実行できるようになります。

AAA データベース

AAA データベースには、システムへのアクセスを制御するユーザ、グループおよびタスク情報が保存されます。 AAA データベースはローカルまたはリモートにできます。 特定の状況で使用されるデータベースは、AAA 設定により異なります。

ローカル データベース

ユーザ、ユーザ グループ、タスク グループなどの AAA データは、セキュア ドメイン ルータ内でローカルに保存できます。 このデータは、メモリ内データベースに保存され、コンフィギュレーション ファイルに保存されます。 保存されたパスワードは暗号化されます。

（注） データベースは、保存されている特定のセキュア ドメイン ルータ（SDR）に対してローカルで、定義されているユーザまたはグループは、同じシステムの他の SDR に表示されません。

残りすべてのユーザをローカルデータベースから削除できます。すべてのユーザを削除すると、ユーザが次にログインするときに、設定ダイアログが表示され、新しいユーザ名およびパスワードを入力するよう求められます。

（注） 設定ダイアログは、ユーザがコンソールにログインするときだけ表示されます。

リモート データベース

AAA データは、CiscoSecure ACS など、外部セキュリティ サーバに保存できます。 サーバに保存されるセキュリティ データは、任意のクライアント（ネットワーク アクセス サーバ（NAS））により使用できます。ただし、クライアントは、サーバ IP アドレスおよび共有秘密を知っている必要があります。

リモート AAA 設定

CiscoSecure ACS などの製品は、共有または外部 AAA データベースの管理に使用できます。 ルータは、標準の IP ベース セキュリティ プロトコル（TACACS+ または RADIUS など）を使用して、リモート AAA サーバと通信します。

クライアント設定

セキュリティ サーバは、ルータと共有するシークレット キーおよびクライアントの IP アドレスで設定する必要があります。

ユーザ グループ

外部サーバで作成されるユーザ グループは、ルータのローカル AAA データベース設定のユーザ グループとは関係がありません。 外部 TACACS+ サーバまたは RADIUS サーバ ユーザ グループの管理は別であるため、ルータはユーザ グループ構造を認識しません。 リモート ユーザまたはグループ プロファイルには、ユーザが属するグループ（ルータで定義）、および個々のタスク ID を指定する属性を含めることができます。 詳細については、TACACS+ および RADIUS 認証ユーザのタスク ID, （14 ページ）の項を参照してください。

外部サーバのユーザ グループの設定は、個々のサーバ製品の設計により異なります。 該当するサーバ製品のマニュアルを参照してください。

タスク グループ

タスク グループは、各操作のタイプ（読み取りや書き込みなど）で許可されるタスク ID のリストで定義されます。 タスク ID は、基本的にルータ システムで定義されます。 外部ソフトウェアのタスク グループを設定するには、タスク ID 定義がサポートされている必要があります。

タスク ID は、外部 TACACS+ または RADIUS サーバでも設定できます。

AAA 設定

この項では、AAA の設定について説明します。

方式リスト

AAA データは、さまざまなデータ ソースに保存できます。 AAA 設定は、方式リストを使用して、AAA データのソースの優先順位を定義します。 AAA は、複数の方式リストを定義でき、アプリケーション（ログインなど）は、これらのいずれかを選択できます。 たとえば、コンソールおよび補助ポートと vty ポートでは、それぞれ異なる方式リストを使用できます。 方式リストが指定されていない場合、アプリケーションは、デフォルトの方式リストを使用します。 デフォルトの方式リストが存在しない場合、AAA は、ローカル データベースとしてソースを使用します。

ロールオーバー メカニズム

AAA は、データベース オプションの優先順位リストを使用するよう設定できます。 システムがデータベースを使用できない場合、リストの次のデータベースに自動的にロールオーバーします。 認証、認可またはアカウンティング要件がデータベースで拒否されると、ロールオーバーは発生せず、要求が拒否されます。

次の方法が選択可能です。

- Local：ローカルで設定されるデータベースを使用します（アカウンティングや一部の認可には適していません）。
- TACACS+：TACACS+ サーバ（CiscoSecure ACS など）を使用します。
- RADIUS：RADIUS サーバを使用します。
- Line：回線パスワードおよびユーザ グループを使用します（認証のみに適しています）。
- None：要求を許可します（認証には適していません）。

サーバ グルーピング

サーバのシングル グローバル リストを保守する代わりに、ユーザは、異なる AAA プロトコル（RADIUS および TACACS+ など）のサーバ グループを形成して、AAA アプリケーション（PPP および EXEC など）に関連付けることができます。

認証

認証は、プリンシパル（ユーザまたはアプリケーション）がシステムへのアクセスを取得する最も重要なセキュリティ プロセスです。プリンシパルは、管理ドメインで一意であるユーザ名（またはユーザ ID）により定義されます。 ユーザにサービスを提供するアプリケーション（EXEC または管理エージェントなど）は、ユーザからユーザ名およびクレデンシャルを取得します。 AAA は、アプリケーションにより渡されたユーザ名およびクレデンシャルに基づいて認証を実行します。 認証ユーザのロールは、ユーザが属する 1 つ以上のグループにより決まります ユーザは、1 つ以上のユーザ グループのメンバにすることができます）。

ルート システム ユーザの認証

root-system ユーザは、システムのセキュア ドメイン ルータの任意のノードにログインできます。ユーザは、root-system グループに属する場合、root-system ユーザです。 root-system ユーザは、ローカルまたはリモート AAA データベースで定義できます。

所有者以外のセキュア ドメイン ルータ ユーザの認証

所有者以外のセキュア ドメイン ルータにログインする場合、ルート システム ユーザは、「@admin」サフィクスをユーザ名に追加する必要があります。 「@admin」サフィクスを使用すると、認証要求が所有者のセキュア ドメイン ルータに送信され、確認されます。所有者のセキュア ドメイン ルータは、認証方法の選択にリスト名 **remote** を使用します。 **remote** 方式リストは、**aaa authentication login remote** *method1 method2...* コマンドを使用して設定されます （AAA 方式リストの設定, （46 ページ）の項を参照）。

所有者のセキュア ドメイン ルータ ユーザの認証

所有者のセキュア ドメイン ルータ ユーザは、所有者のセキュア ドメイン ルータ ユーザに関連付けられている特定のセキュア ドメイン ルータに属するノードだけにログインできます。 ユーザが root-sdr グループのメンバである場合、ユーザは、所有者のセキュア ドメイン ルータ ユーザとして認証されます。

セキュア ドメイン ルータ ユーザの認証

セキュア ドメイン ルータ ユーザの認証は、所有者のセキュア ドメイン ルータ ユーザの認証に似ています。 指定された所有者のセキュア ドメイン ルータ ユーザ グループまたは root-system ユーザ グループのメンバで見つからないユーザは、セキュア ドメイン ルータ ユーザとして認証されます。

認証フロー制御

AAA は、次のプロセスに従い認証を実行します。

1. ユーザが、ユーザ名およびパスワード（またはシークレット）を提供して認証を要求します。
2. AAA が、ユーザのパスワードを検証して、パスワードがデータベースのものと一致しない場合ユーザを拒否します。
3. AAA が、ユーザのロールを決定します（ルート システム ユーザ、ルート SDR ユーザまたは SDR ユーザ）。

   - ユーザが root-system ユーザ グループのメンバとして設定されている場合、AAA は、そのユーザを root-system ユーザとして認証します。
   - ユーザが所有者のセキュア ドメイン ルータ ユーザ グループのメンバとして設定されている場合、AAA は、そのユーザを所有者のセキュア ドメイン ルータ ユーザとして認証します。

- ユーザが root-system ユーザ グループまたは所有者のセキュア ドメイン ルータ ユーザ グループのメンバとして設定されていない場合、AAA は、そのユーザをセキュア ドメイン ルータ ユーザとして認証します。

クライアントは、ユーザの許可されているタスク ID を認証中に取得できます。この情報は、ユーザが属するユーザ グループで指定されているすべてのタスク グループ定義の集合を形成することで取得されます。このような情報を使用するクライアントは、通常、タスク ID セットが静的であるユーザのセッション（API セッションなど）を作成します。EXEC および外部 API クライアントは、両方ともこの機能を使用して、操作を最適化できます。EXEC は、該当しないコマンドを非表示にでき、EMS アプリケーションは、たとえば、該当しないグラフィカル ユーザ インターフェイス（GUI）メニューをディセーブルにできます。

ユーザ グループ メンバーシップなどのユーザの属性やタスク権限が変更されると、これらの変更された属性は、ユーザの現在アクティブなセッションでは反映されません。これらは、ユーザの次のセッションで有効になります。

Korn シェル認証

Korn シェル（ksh）は、ルート プロセッサ（RP）、スタンバイ RP、分散 RP カードの補助ポート、さらにラインカード（LC）とサービス プロセッサ（SP）のコンソールおよび補助ポートのプライマリ シェルです。次に、ksh 認証の特徴をいくつか示します。

- セキュリティのため、ksh 認証では、シークレットを設定できるのは root-system ユーザだけです。標準パスワードの root-system ユーザは認証されません。これは、標準パスワードは、二方向暗号化で、パスワード情報が簡単に復号化できるフラッシュ ディスクに保存され、セキュリティ リスクが発生するためです。
- シークレットを使用する root-system ユーザが標準 AAA CLI を使用して設定されるたびに、そのユーザは、有効な ksh ユーザになります。個別の設定は必要ありません。
- Ksh は、root-system ユーザであっても、TACACS+ または RADIUS ユーザを認証しません。
- Ksh 認証は、シングル ユーザ パスワード データベースを使用します。つまり、dSC の root-system ユーザが、標準 AAA CLI を使用して設定されると、そのユーザは、任意のカードのユーザ名パスワードを使用してログインできます。これには、RP、スタンバイ RP、LC および SP が含まれます。
- Ksh 認証は、カードのブート後に無効またはバイパスすることはできません。認証をバイパスするには、ユーザは、カードをリロードする必要があります（詳細については、「ksh 認証のバイパス」の項を参照してください）。
- ksh は、認証されないコンソールから実行します（**run** コマンドを使用します）。これは、**run** コマンドは、root-system タスク ID を必要とするためです。ユーザはすでに root-system であるため、再び認証されません。

ksh 認証のバイパス

ksh 認証は処理が軽量で、プロセスも多くありませんが、次の場合などは、ksh 認証をバイパスする必要があります。

- dSC（ACTIVE RP）disk0 の破損
- Qnet 接続の切断
- dSC（ACTIVE RP）のノード ID を決定できない

ksh 認証をバイパスするには、ユーザは、ROMMON 変数 AUX_AUTHEN_LEVEL を 0 に設定し、イメージをリロードする必要があります。 リブートは、認証のバイパスが必要なカードだけで必要です。

ROMMON 変数 AUX_AUTHEN_LEVEL には、次のいずれかの値を指定できます。

- 0：認証がカードでバイパスされます。
- 1：認証が失われます。 認証は、ベストエフォートの原則で実行されます。認証により、ユーザは、システムが認証情報に正常にアクセスできない場合に ksh にアクセスできます。
- 2：厳密な認証です。 これは、デフォルトの状態です。

  認証はバイパスされません。 認証インフラストラクチャがダウンしていても、システムはアクセスを拒否するだけです。

たとえば、カードの認証をバイパスするには、次のように入力します。

```
rommon1> AUX_AUTHEN_LEVEL=0
rommon2> sync
rommon2> boot tftp:/ ...
```

パスワード タイプ

ユーザおよびそのユーザのグループ メンバーシップを設定する場合、暗号化またはクリアテキストの 2 つのパスワードを指定できます。

ルータは、二方向および一方向（シークレット）の両方の暗号化ユーザ パスワードをサポートします。 オリジナルの暗号化されていないパスワード文字列が暗号化シークレットからは推測できないため、シークレット パスワードはユーザ ログイン アカウントに適しています。 アプリケーションによっては（PPP など）、パケットでのパスワードの送信など、独自の機能のための保存パスワードを復号化する必要があるので、二方向のみのパスワードが必要です。 ログインユーザでは、両方のタイプのパスワードを設定できますが、一方のパスワードがすでに設定されている状態でもう一方のパスワードを設定すると、警告メッセージが表示されます。

シークレットとパスワードの両方をユーザに設定すると、ログインなど、復号化できるパスワードを必要としないすべての操作で、シークレットが優先されます。 PPP などのアプリケーションでは、シークレットが存在する場合でも、二方向の暗号化パスワードが使用されます。

タスクベースの認可

AAA は、CLI または API を介した操作の任意の制御、設定またはモニタに「タスク許可」を使用します。 Cisco IOS ソフトウェアの特権レベルの概念は、Cisco IOS XR ソフトウェアでは、タスクベースの認可システムに置き換わりました。

タスク ID

ユーザによるCisco IOS XR ソフトウェアの制御、設定およびモニタを可能にする操作タスクは、タスク ID 別に表されます。 タスク ID は、コマンドで操作をする許可を定義します。 ユーザには、ルータに許可されているアクセスの範囲を定義するタスク IDのセットが関連付けられます。

タスク ID は、次のようにユーザに割り当てられます。 各ユーザは、1 つの以上のユーザ グループに関連付けられます。 すべてのユーザ グループは、1 つ以上のタスク グループに関連付けられ、すべてのタスク グループは、タスク ID のセットで定義されます。 つまり、ユーザと特定のユーザ グループを関連付けることで、そのユーザとタスク ID の特定のセットが関連付けられます。 タスク ID が関連付けられたユーザは、そのタスク ID に関連付けられている任意の操作を実行できます。

タスク ID に関する一般的な使用上のガイドライン

ほとんどのルータ制御、設定またはモニタリング操作（CLI または XML API）は、タスク ID の特定のセットが関連付けられます。 通常、特定の CLI コマンドまたは API イノベーションは、1 つ以上のタスク ID が関連付けられます。 config および commit コマンドでは、特定のタスク ID 許可は必要ありません。 設定およびコミット操作では、特定のタスク ID 許可は必要ありません（エイリアスでもタスク ID 許可は必要ありません）。 root-lr 許可が割り当てられるまで、config replace を実行できません。 コンフィギュレーション モードを開始しない場合、TACACS+ コマンド認可を使用して、config コマンドを拒否できます。 これらの関連付けは、ルータ内でハードコード化されていて、変更できません。 タスク ID は、特定のタスクを実行する許可を付与します。 タスク ID では、タスクを実行する許可は拒否されません。 タスク ID 操作は、次の表にリストされているクラスの 1 つ、すべて、または任意の組み合わせにすることができます。

表 *2*：タスク *ID* クラス

操作	説明
Read	読み取り専用操作を許可します。
Write	変更操作を許可、および読み取り操作を暗黙的に許可します。
Execute	ping や Telnet など、アクセス操作を許可します。
Debug	デバッグ操作を許可します。

システムは、各 CLI コマンドおよび API イノベーションがユーザのタスク ID 許可リストと一致しているか検証します。 CLI コマンドの使用時に問題が発生した場合、システム管理者に連絡してください。

スラッシュで区切られた複数のタスク ID 操作（read/write など）は、両方の操作が指定のタスク ID に適用されることを示します。

カンマで区切られた複数のタスク ID 操作（read, read/write など）は、両方の操作が個々のタスク ID に適用されることを示します。 たとえば、**copy ipv4 access-list** コマンドは、読み取りおよび書き込み操作を acl タスク ID に適用し、実行操作を *filesystem* タスク ID に適用できます。

タスク ID および操作の列が指定されていない場合、コマンドは、タスク ID および操作とユーザとの関連付けなしで使用されます。 また、ROM モニタ コマンドを使用するために、ユーザにタスク ID を関連付ける必要はありません。

コマンドが特定のコンフィギュレーション サブモードで使用される場合、そのコマンドを使用するための追加タスク ID をユーザに関連付ける必要があります。 たとえば、**show redundancy** コマンドを実行するには、ユーザに、system（read）タスク ID および操作を関連付ける必要があります（次の例を参照）。

```
RP/0/RSP0/CPU0:router# show redundancy
```

また、管理 EXEC モードでは、ユーザに、admin および system（read）タスク ID および操作を関連付ける必要があります（次の例を参照）。

```
RP/0/RSP0/CPU0:router# admin
RP/0/RSP0/CPU0:router(admin)# show redundancy
```

TACACS+ および RADIUS 認証ユーザのタスク ID

Cisco IOS XR ソフトウェア AAA では、TACACS+ および RADIUS 方式で認証されるユーザに次の方法でタスク許可を割り当てることができます。

- タスク マップのテキスト バージョンを、外部 TACACS+ および RADIUS サーバのコンフィギュレーション ファイルに直接指定します。

  詳細については、タスク マップ，（14 ページ）を参照してください。

- 外部 TACACS+ および RADIUS サーバのコンフィギュレーション ファイルで特権レベルを指定します。

  詳細については、特権レベル マッピング，（17 ページ）を参照してください。

- TACACS+ および RADIUS 方式で認証するユーザと同じユーザ名でローカル ユーザを作成します。

- 許可が TACACS+ および RADIUS 方式で認証する任意のユーザに適用されるデフォルト タスク グループを設定別に指定します。

タスク マップ

外部 TACACS+ サーバおよび RADIUS サーバを使用して認証されるユーザに対して、Cisco IOS XR ソフトウェア AAA は、タスク ID をリモートで定義する方式をサポートします。

タスク ストリングの形式

TACACS+ サーバのコンフィギュレーション ファイルのタスク文字列は、カンマ（,）で区切られたトークンで構成されます。各トークンは、タスク ID 名およびその許可、またはこの特定のユーザを含むユーザ グループのいずれかで構成されます（次の例を参照）。

task = “ *permissions* : *taskid name* , # *usergroup name* , ...”

（注） Cisco IOS XR ソフトウェアでは、タスク ID を外部 RADIUS または TACACS+ サーバの属性として指定できます。 サーバが非 Cisco IOS XR ソフトウェア システムと共有される場合、これらの属性には、サーバマニュアルで示されているように、オプションマークが付けられます。たとえば、CiscoSecure ACS および Cisco のフリーウェア TACACS+ サーバでは、オプション属性の属性値の前に等号記号（=）ではなく、アスタリスク（*）が必要です。属性をオプションとして設定する場合、TACACS+ サーバのマニュアルを参照してください。

たとえば、user1 BGP という名前のユーザに、read、write および execute 許可を付与し、user1 を operator という名前のユーザ グループに含める場合、外部サーバの TACACS+ コンフィギュレーション ファイルのユーザ名エントリは次のようになります。

```
user = user1{
member = some-tac-server-group
opap = cleartext "lab"
service = exec {
task = "rwx:bgp,#operator"
}
}
```

r、w、x、d はそれぞれ read、write、execute、debug に対応します。ポンド記号（#）はユーザ グループが続くことを示します。

（注） Cisco IOS ソフトウェアに基づいたシステムとの相互運用性をイネーブルにするには、「task」の前にオプション キーワードを追加する必要があります。

CiscoSecure ACS が使用される場合、次の手順を実行して、タスク ID とユーザ グループを指定します。

手順の概要

1. ユーザ名とパスワードを入力します。
2. [Group Setup] ボタンをクリックして、[Group Setup] ウィンドウを表示します。
3. [Group] ドロップダウン リストから、更新するグループを選択します。
4. [Edit Settings] ボタンをクリックします。
5. スクロール矢印を使用して、[Shell (exec)] チェックボックスを探します。
6. [Shell (exec)] チェックボックスを選択して、カスタム属性設定をイネーブルにします。
7. [Custom attributes] チェックボックスを選択します。
8. フィールドに空白や引用符を含めずに次のタスク文字列を入力します。
9. [Submit + Restart] ボタンをクリックしてサーバを再起動します。

手順の詳細

ステップ 1 ユーザ名とパスワードを入力します。

ステップ 2 [Group Setup] ボタンをクリックして、[Group Setup] ウィンドウを表示します。

ステップ 3 [Group] ドロップダウン リストから、更新するグループを選択します。

ステップ 4 [Edit Settings] ボタンをクリックします。

ステップ 5 スクロール矢印を使用して、[Shell (exec)] チェックボックスを探します。

ステップ 6 [Shell (exec)] チェックボックスを選択して、カスタム属性設定をイネーブルにします。

ステップ 7 [Custom attributes] チェックボックスを選択します。

ステップ 8 フィールドに空白や引用符を含めずに次のタスク文字列を入力します。

例：

```
task=rwx:bgp,#netadmin
```

ステップ 9 [Submit + Restart] ボタンをクリックしてサーバを再起動します。
次の RADIUS ベンダー固有属性（VSA）の例では、ユーザは、sysadmin 事前定義タスク グループに含まれ、BGP を設定でき、OSPF の設定を表示できます。

例：

```
user Auth-Type := Local, User-Password == lab
        Service-Type = NAS-Prompt-User,
        Reply-Message = "Hello, %u",
        Login-Service = Telnet,
        Cisco-AVPair = "shell:tasks=#sysadmin,rwx:bgp,r:ospf"
```

user1 が、ユーザ名 user1 および適切なパスワードを使用して、正常に外部 TACACS+ サーバに接続およびログインすると、**show user tasks** コマンドを EXEC モードで使用して、user1 が実行できるすべてのタスクを表示できます。 次に例を示します。

例：

```
Username:user1
Password:
RP/0/RSP0/CPU0:router# show user tasks

Task:      basic-services  :READ    WRITE    EXECUTEDEBUG
Task:                 bgp  :READ    WRITE    EXECUTE
Task:                 cdp  :READ
Task:                diag  :READ
Task:          ext-access  :READ             EXECUTE
Task:             logging  :READ
```

タスク文字列が指定されていない user2 という名前のユーザが外部サーバにログインすると、次の情報が表示されます。

例：

```
Username:user2
Password:
RP/0/RSP0/CPU0:router# show user tasks
No task ids available
```

特権レベル マッピング

タスク ID の概念をサポートしない TACACS+ デーモンとの互換性のために、AAA は、外部 TACACS+ サーバ コンフィギュレーション ファイルのユーザの特権レベルとローカル ユーザ グループのマッピングをサポートします。 TACACS+ 認証に従い、外部 TACACS+ サーバから返される特権レベルからマッピングされるユーザ グループのタスク マップがユーザに割り当てられます。 たとえば、特権レベル 5 が外部 TACACS サーバから返された場合、AAA は、ローカル ユーザ グループ priv5 のタスク マップを取得しようとします。 このマッピング プロセスは、1 ～ 13 までの他の特権レベルでも同様です。 特権レベル 15 の場合、root-system ユーザ グループが使用されます。特権レベル 14 は、ユーザ グループ owner-sdr にマッピングされます。

たとえば、シスコ フリーウェア tac plus サーバでは、コンフィギュレーション ファイルは、そのコンフィギュレーション ファイルで *priv_lv* を指定する必要があります（次の例を参照）。

```
user = sampleuser1{
    member = bar
    service = exec-ext {
        priv_lvl = 5
    }
}
```

この例の 5 という数値は、ユーザ *sampleuser* に割り当てる必要がある任意の特権レベルに置き換えることができます。

RADIUS サーバでは、タスク ID は、Cisco-AVPair を使用して定義されます（次の例を参照）。

```
user = sampleuser2{
    member = bar
    Cisco-AVPair = "shell:tasks=#root-system,#cisco-support"{
        Cisco-AVPair = "shell:priv-lvl=10"
```

```
        }
}
```

AAA サービスの XML スキーマ

Extensible Markup Language（XML）インターフェイスは、XML ドキュメント形式で要求と応答を使用して、AAA を設定およびモニタします。 AAA コンポーネントは、設定およびモニタリングに使用されるデータの内容と構造に対応する XML スキーマを発行します。 XML ツールおよびアプリケーションは、このスキーマを使用して、XML エージェントと通信して設定を実行します。

次のスキーマが、AAA により発行されます。

- 認証、許可、アカウンティングの設定
- ユーザ、ユーザ グループおよびタスク グループ設定
- TACACS+ サーバおよびサーバ グループ設定
- RADIUS サーバおよびサーバ グループ設定

RADIUS について

RADIUS は、不正なアクセスからネットワークのセキュリティを保護する分散クライアント/サーバ システムです。 シスコの実装では、RADIUS クライアントは Cisco ルータ上で稼働します。認証要求とアカウンティング要求は、すべてのユーザ認証情報とネットワーク サービスアクセス情報が格納されている中央の RADIUS サーバに送信されます。

RADIUS は完全にオープンなプロトコルであり、ソース コード形式で配布されているため、現在使用できる任意のセキュリティ システムと連携するように変更できます。

シスコは、AAA セキュリティ パラダイムの下で RADIUS をサポートしています。 RADIUS は、TACACS+、Kerberos、ローカル ユーザ名の検索など、他の AAA セキュリティ プロトコルと併用できます。

（注） RADIUS はすべての Cisco プラットフォームでサポートされますが、RADIUS でサポートされる一部の機能は、指定されたプラットフォームだけで実行されます。

RADIUS は、リモート ユーザのネットワーク アクセスを維持すると同時に高度なレベルのセキュリティを必要とするさまざまなネットワーク環境に実装されています。

RADIUS は、アクセスのセキュリティが必要な次のネットワーク環境で使用できます。

- それぞれが RADIUS をサポートする、マルチベンダー アクセス サーバによるネットワーク。たとえば、複数のベンダーのアクセス サーバが、1 つの RADIUS サーバベース セキュリティ データベースを使用します。 複数ベンダーのアクセス サーバからなる IP ベースのネットワークでは、ダイヤルイン ユーザは RADIUS サーバを通じて認証されます。RADIUS サーバは、Kerberos セキュリティ システムで動作するようにカスタマイズされています。

- アプリケーションが RADIUS プロトコルをサポートするターンキー ネットワーク セキュリティ環境。たとえば、「スマート カード」アクセス コントロール システムを使用するアクセス環境。 ある事例では、RADIUS と Enigma のセキュリティ カードを併用してユーザを検証し、ネットワーク リソースに対するアクセス権を付与しています。
- すでに RADIUS を使用中のネットワーク。 RADIUS 機能を持つ Cisco ルータをネットワークに追加できます。 Terminal Access Controller Access Control System Plus（TACACS+）サーバに移行する場合、これが最初の手順となります。
- ユーザが単一のサービスにだけアクセスする必要があるネットワーク。 RADIUS を使用すると、単一ホスト、単一ユーティリティ（Telnet など）、または単一プロトコル（ポイントツーポイントプロトコル（PPP））に対するユーザアクセスを制御できます。 たとえば、ユーザがログインすると、RADIUS は、IP アドレス 10.2.3.4 を使用してそのユーザが PPP を実行する権限を持っていることを識別し、定義済みのアクセス リストが開始されます。
- リソース アカウンティングが必要なネットワーク。 RADIUS アカウンティングは、RADIUS 認証または RADIUS 認可とは個別に使用できます。 RADIUS アカウンティング機能を使用すると、サービスの開始および終了時に、セッション中に使用したリソース（時間、パケット、バイトなど）の量を示すデータを送信できます。 インターネット サービス プロバイダー（ISP）は、RADIUS アクセス コントロールおよびアカウンティング ソフトウェアのフリーウェア バージョンを使用して、セキュリティおよび課金の独自ニーズを満たすこともできます。
- 事前認証のサポートを希望するネットワーク。 ネットワークに RADIUS サーバを導入すると、AAA 事前認証を設定し、事前認証のプロファイルを設定できます。 サービス プロバイダーが事前認証を使用すると、既存の RADIUS ソリューションを使用するポートの管理性が向上し、共有リソースを効率的に管理して、各種のサービス レベル契約を提供できるようになります。

RADIUS が適さないネットワーク セキュリティ状況

RADIUS は次のネットワーク セキュリティ状況には適していません。

- マルチプロトコル アクセス環境。 RADIUS は次のプロトコルをサポートしていません。
  - AppleTalk Remote Access（ARA）
  - NetBIOS Frame Control Protocol（NBFCP）
  - NetWare Asynchronous Services Interface（NASI）
  - X.25 PAD 接続
- ルータ間で接続している環境。 RADIUS は、双方向認証を行いません。 RADIUS は、ルータと RADIUS 認証を必要とするシスコ製以外のルータとの認証に使用できます。
- 各種のサービスを使用するネットワーク。 RADIUS は、一般に 1 人のユーザを 1 つのサービス モデルにバインドします。

RADIUS の動作

ユーザがログインを試行し、RADIUS を使用してアクセス サーバから認証を受ける場合、次の手順が発生します。

1. ユーザが、ユーザ名とパスワードの入力を求められ、入力します。
2. ユーザ名および暗号化されたパスワードが、ネットワーク経由でRADIUSサーバに送信されます。
3. ユーザは、RADIUS サーバから次のいずれかの応答を受信します。
   a. ACCEPT：ユーザが認証されたことを表します。
   a. REJECT：ユーザは認証されず、ユーザ名とパスワードの再入力を求められるか、アクセスを拒否されます。
   a. CHALLENGE：RADIUS サーバによってチャレンジが発行されます。 チャレンジは、ユーザから追加データを収集します。
   a. CHANGE PASSWORD：RADIUSサーバからユーザに対して新しいパスワードの選択を求める要求が発行されます。

   ACCEPT または REJECT 応答には、EXEC またはネットワーク認可に使用される追加データが含まれています。 RADIUS 認可を使用するには、まず RADIUS 認証を完了する必要があります。 ACCEPT または REJECT パケットに含まれる追加データの内容は次のとおりです。

   - Telnet、rlogin、またはローカルエリア トランスポート（LAT）、および PPP、Serial Line Internet Protocol（SLIP）、またはEXECサービスなどといった、ユーザがアクセスできるサービス。
   - ホストまたはクライアントの IP アドレス、アクセス リスト、ユーザ タイムアウトなどの接続パラメータ。

AAA サービスの設定方法

AAA サービスを設定するには、以下の項で説明する作業を実行します。

タスク グループの設定

タスクベースの認可では、その基本要素としてタスク *ID* の概念が使用されます。 タスク ID は、ユーザの操作実行許可を定義します。 各ユーザは、タスク ID で識別される許可されたルータ操作タスクのセットが関連付けられます。ユーザは、ユーザ グループに関連付けられることで許可が付与されます。ユーザ グループには、タスク グループが関連付けられます。 各タスクグループには、使用できるタスク ID の Cisco CRS-1 セットから選択された 1 つ以上のタスク ID が関連

付けられます。認可スキームを設定する場合、最初にタスク グループを設定します。次に、タスク グループ、個々のユーザの順に設定します。

タスク グループの設定

タスク グループには、アクション タイプごとに一連のタスク ID が設定されます。

no プレフィックスを使用した **task** コマンドを指定して、特定のタスク ID をタスク グループから削除できます。

タスク グループ自体は削除できます。 ドキュメント名のあるタスク グループを削除すると、エラーが発生します。

はじめる前に

タスク グループを作成して、タスク ID を関連付ける前に、タスク ID のルータ リストおよび各タスク ID の目的について理解しておく必要があります。 **show aaa task supported** コマンドを使用して、タスク ID の完全なリストを表示します。

（注） AAA タスク ID の write 許可を持っているユーザだけタスク グループを設定できます。

手順の概要

1. **configure**
2. **taskgroup** *taskgroup-name*
3. **description** *string*
4. **task** {**read** | **write** | **execute** | **debug**} *taskid-name*
5. ステップ 2 で指定したタスク グループに関連付ける各タスク ID で、ステップ 4 を繰り返します。
6. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**taskgroup** *taskgroup-name* 例： `RP/0/RSP0/CPU0:router(config)# taskgroup beta`	特定のタスク グループの名前を作成し、タスク グループ コンフィギュレーション サブモードを開始します。 • **no** 形式の **taskgroup** コマンドを指定すると、特定のタスク グループをシステムから削除できます。
ステップ 3	**description** *string* 例： `RP/0/RSP0/CPU0:router(config-tg)# description this is a sample task group description`	（任意）ステップ 2 で指定したタスク グループの説明を作成します。
ステップ 4	**task** {**read** | **write** | **execute** | **debug**} *taskid-name* 例： `RP/0/RSP0/CPU0:router(config-tg)# task read bgp`	ステップ 2 で指定したタスク グループに関連付けるタスク ID を指定します。 • そのタスク ID が関連付けられ、タスク グループのメンバにより実行される任意の CLI または API 呼び出しに **read** 許可を割り当てます。 • **no** プレフィックスを使用した **task** コマンドを指定して、特定のタスク ID をタスク グループから削除できます。
ステップ 5	ステップ 2 で指定したタスク グループに関連付ける各タスク ID で、ステップ 4 を繰り返します。	—

ステップ 6	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュ

	コマンドまたはアクション	目的
		レーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

次の作業

タスク グループのすべてのセットの設定が完了したら、ユーザ グループのフル セットを設定します（「ユーザ グループの設定」の項を参照）。

ユーザ グループの設定

ユーザ グループは、タスク グループなど一連のユーザに対するコマンド パラメータによって設定されます。 **usergroup** コマンドを入力すると、ユーザ グループ コンフィギュレーション サブモードにアクセスします。 **usergroup** コマンドの **no** 形式を使用すると、特定のユーザ グループを削除できます。 システムで参照されているユーザ グループを削除すると、警告が表示されます。

はじめる前に

（注） WRITE:AAA タスク ID が関連付けられているユーザだけ、ユーザ グループを設定できます。ユーザ グループは、root-system や owner-sdr などの事前定義されたグループのプロパティを継承できません。

手順の概要

1. **configure**
2. **usergroup** *usergroup-name*
3. **description** *string*
4. **taskgroup** *taskgroup-name*
5. ステップ 2, （24 ページ） で指定したユーザ グループを関連付ける各タスク グループで、ステップ 4, （26 ページ） を繰り返します。
6. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**usergroup** *usergroup-name* 例： `RP/0/RSP0/CPU0:router(config)# usergroup beta`	特定のユーザ グループの名前を作成し、ユーザ グループ コンフィギュレーション サブモードを開始します。 • **no** 形式の **usergroup** コマンドを指定すると、特定のユーザ グループをシステムから削除できます。
ステップ 3	**description** *string* 例： `RP/0/RSP0/CPU0:router(config-ug)# description this is a sample user group description`	（任意）ステップ 2, （24 ページ） で指定したユーザ グループの説明を作成します。
ステップ 4	**taskgroup** *taskgroup-name* 例： `RP/0/RSP0/CPU0:router(config-ug)# taskgroup beta`	ステップ 2, （26 ページ） で指定したユーザ グループを、この手順で指定したタスク グループに関連付けます。 • ユーザ グループは、入力したタスク グループに対してすでに定義されている設定属性（タスク ID リストと権限）を取ります。
ステップ 5	ステップ 2, （24 ページ） で指定したユーザ グループを関連付ける各タスク グループで、ステップ 4, （26 ページ） を繰り返します。	—

ステップ 6	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。

	コマンドまたはアクション	目的
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

次の作業

ユーザ グループのフル セットの設定が完了したら、個々のユーザを設定します（ユーザの設定, （25 ページ）の項を参照）。

ユーザの設定

このタスクを実行して、ユーザを設定します。

各ユーザは、管理ドメイン内で一意のユーザ名によって識別されます。 各ユーザは、少なくとも 1 つのユーザ グループのメンバーであることが必要です。 ユーザ グループを削除すると、そのグループに関連付けられたユーザが孤立する場合があります。 AAA サーバでは孤立したユーザも認証されますが、ほとんどのコマンドは許可されません。

手順の概要

1. **configure**
2. **username** *user-name*
3. 次のいずれかを実行します。
   - **password** {**0** | **7**} *password*
   - **secret** {**0** | **5**} *secret*
4. **group** *group-name*
5. ステップ 2, （26 ページ） で指定したユーザに関連付けられた各ユーザ グループで、ステップ 4, （26 ページ） を繰り返します。
6. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**username** *user-name* 例： `RP/0//CPU0:router(config)# username user1`	RSP0 新しいユーザの名前を作成（または現在のユーザを識別）して、ユーザ名コンフィギュレーション サブモードを開始します。 • *user-name* 引数には 1 つの単語だけ使用できます。 スペースや引用符は使用できません。

ステップ 3	次のいずれかを実行します。
	• **password** {**0** | **7**} *password*
	• **secret** {**0** | **5**} *secret*
	
	例：
	`RP/0/RSP0/CPU0:router(config-un)# password 0 pwd1`
	または
	`RP/0/RSP0/CPU0:router(config-un)# secret 0 sec1`
		ステップ 2, （26 ページ） で指定したユーザのパスワードを指定します。
		• **secret** コマンドを使用して、ステップ 2, （26 ページ） で指定したユーザ名の安全なログイン パスワードを作成します。
		• **password** コマンドに続けて **0** を入力した場合は、暗号化されていない（クリアテキストの）パスワードを続けます。 **password** コマンドに続けて **7** を入力した場合は、暗号化されたパスワードを続けます。
		• **secret** コマンドに続けて **0** を入力した場合は、暗号化されていない（クリアテキストの）安全なパスワードを続けます。 **secret** コマンドに続けて **5** を入力した場合は、暗号化された安全なパスワードを続けます。
		• タイプ **0** が、**password** コマンドおよび **secret** コマンドのデフォルトです。

ステップ 4	**group** *group-name* 例： `RP/0/RSP0/CPU0:router(config-un)# group sysadmin`	ステップ 2, （26 ページ） で指定したユーザ名を、**usergroup** コマンドを介して定義したユーザ グループに割り当てます。 • ユーザは、ユーザ グループのさまざまなタスク グループへの割り当てによって定義された内容に従って、ユーザ グループのすべての属性を受け取ります。 • 各ユーザは、少なくとも 1 つのユーザ グループに割り当てる必要があります。 ユーザは複数のユーザ グループに属することがあります。

	コマンドまたはアクション	目的
ステップ 5	ステップ 2, （26 ページ） で指定したユーザに関連付けられた各ユーザ グループで、ステップ 4, （26 ページ） を繰り返します。	—

ステップ 6	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

次の作業

ユーザのフル セットの設定が完了したら、RADIUS サーバ通信または TACACS+ サーバを使用するようにルータを設定します（RADIUS サーバ通信のルータの設定, （27 ページ）またはTACACS+ サーバの設定, （37 ページ）の項を参照）。

RADIUS サーバ通信のルータの設定

ルータと RADIUS サーバの通信を設定します。

通常、RADIUS ホストは、シスコ（CiscoSecure ACS）、Livingston、Merit、Microsoft、または他のソフトウェア プロバイダーの RADIUS サーバ ソフトウェアを実行するマルチユーザ システムです。 RADIUS サーバとの通信のためにルータを設定するには、次のような要素があります。

- ホスト名または IP アドレス
- 認証の宛先ポート
- アカウンティングの宛先ポート
- 再送信回数
- タイムアウト時間
- キー文字列

RADIUS セキュリティ サーバは、ホスト名または IP アドレス、ホスト名と特定のユーザ データグラム プロトコル（UDP）ポート番号、または IP アドレスおよび特定の UDP ポート番号により識別されます。 固有の識別情報は、IP アドレスと UDP ポート番号の組み合わせで構成されます。これにより、RADIUS ホストとして定義されているさまざまなポートが、固有の AAA サービスを提供できるようになります。 つまり、この固有識別情報を使用して、ある IP アドレスに位置する 1 台のサーバ上に複数の UDP ポートが存在する場合、それぞれの UDP ポートに対して RADIUS 要求を送信できます。 同じ RADIUS サーバ上の異なる 2 つのホスト エントリに同じサービス（たとえばアカウンティング）を設定した場合、2 番めに設定したホスト エントリは、最初に設定したホスト エントリのフェールオーバー バックアップとして動作します。 この場合、最初のホスト エントリがアカウンティング サービスを提供できなかった場合、ネットワーク アクセス サーバは同じ装置上でアカウンティング サービス用に設定されている 2 番めのホスト エントリを試行します （試行される RADIUS ホスト エントリの順番は、設定されている順序に従います）。

RADIUS サーバと Cisco ルータは、共有秘密テキスト ストリングを使用してパスワードを暗号化し、応答を交換します。RADIUS を設定して AAA セキュリティ コマンドを使用するには、RADIUS サーバ デーモンを実行するホストと、ルータと共有する秘密テキスト（キー）ストリングを指定する必要があります。

タイムアウト値、再送信値、および暗号キー値には、すべての RADIUS サーバを対象にしたグローバル設定、サーバ別設定、またはグローバル設定とサーバ別設定の組み合わせを使用できます。 すべての RADIUS サーバとルータとの通信にこのようなグローバル設定を適用するには、**radius-server timeout**、**radius-server retransmit**、および **radius-server key** という 3 つの固有なグローバル コンフィギュレーション コマンドを使用します。 特定の RADIUS サーバにこれらの値を適用するには、**radius-server host** コマンドをグローバル コンフィギュレーション モードで使用します。

（注） 同じシスコ製ネットワーク アクセス サーバで、タイムアウト、再送信、およびキー値のコマンドを同時に設定（グローバル設定およびサーバ別設定）できます。 ルータにグローバル機能とサーバ別機能の両方を設定する場合、サーバ別のタイマー、再送信、およびキー値のコマンドの方が、グローバルのタイマー、再送信、およびキー値のコマンドよりも優先されます。

手順の概要

1. **configure**
2. **radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
3. **radius-server retransmit** *retries*
4. **radius-server timeout** *seconds*
5. **radius-server key** {**0** *clear-text-key* | **7** *encrypted-key* | *clear-text-key*}
6. **radius source-interface** *type instance* [**vrf** *vrf-id*]
7. 設定する各外部サーバで、ステップ 2, （29 ページ）～ステップ 6, （30 ページ） を繰り返します。
8. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
9. show radius

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*] 例： `RP/0/RSP0/CPU0:router(config)# radius-server host host1`	リモート RADIUS サーバのホスト名または IP アドレスを指定します。 • **auth-port** *port-number* オプションを使用して、認証専用の RADIUS サーバに固有の UDP ポートを設定します。 • **acct-port** *port-number* オプションを使用して、アカウンティング専用の RADIUS サーバに固有の UDP ポートを設定します。 • ネットワーク アクセス サーバが単一の IP アドレスと関連付けられた複数のホストエントリを認識するように設定するには、このコマンドを必要な回数だけ繰り返します。その際、各 UDP ポート番号が異なっていることを確認してください。 各 RADIUS ホストで使用するタイムアウト、再送信回数、および暗号キーの値をそれぞれ設定してください。

	コマンドまたはアクション	目的
		• タイムアウトを設定しない場合、グローバル値が使用されます。設定する場合、値の範囲は 1 ～ 1000 です。再送信値を設定しない場合、グローバル値が使用されます。設定する場合、値の範囲は 1 ～ 100 です。 キー文字列を指定しない場合、グローバル値が使用されます。 （注） キーは、RADIUS サーバで使用する暗号化キーに一致するテキストストリングでなければなりません。 キーの先頭にあるスペースは無視されますが、キー内のスペースとキー末尾のスペースは使用されるため、キーは常に **radius-server host** コマンド構文の最後のアイテムとして設定してください。 キーにスペースを使用する場合、引用符をキーに含める場合を除き、引用符でキーを囲まないでください。
ステップ 3	**radius-server retransmit** *retries* 例： `RP/0/RSP0/CPU0:router(config)# radius-server retransmit 5`	Cisco IOS XR ソフトウェアで RADIUS サーバ ホストのリストを検索する回数を指定します。 • この例では、再転送の試行回数は 5 に設定されます。
ステップ 4	**radius-server timeout** *seconds* 例： `RP/0/RSP0/CPU0:router(config)# radius-server timeout 10`	タイムアウトになるまでルータがサーバ ホストの応答を待機する秒数を設定します。 • この例では、間隔タイマーは 10 秒に設定されます。
ステップ 5	**radius-server key** {**0** *clear-text-key* | **7** *encrypted-key* | *clear-text-key*} 例： `RP/0/RSP0/CPU0:router(config)# radius-server key 0 samplekey`	ルータおよび RADIUS デーモン間のすべての RADIUS コミュニケーションの認証キーおよび暗号キーを指定します。
ステップ 6	**radius source-interface** *type instance* [**vrf** *vrf-id*] 例： `RP/0/RSP0/CPU0:router(config)# radius source-interface GigabitEthernet 0/3/0/1`	（任意）RADIUS で、すべての発信 RADIUS パケットに指定のインターフェイスまたはサブインターフェイスの IP アドレスが使用されるようにします。 • 指定されたインターフェイスまたはサブインターフェイスには、IP アドレスが関連付けられている必要があります。 指定のインターフェイスまたはサブインターフェイスに IP アドレスが設定されていないか、そのインターフェイスがダウン状態にある場合、RADIUS はデフォルトに戻ります。 これを回避するには、インターフェイスまたはサブインターフェイスに IP アドレ

	コマンドまたはアクション	目的
		スを追加するか、そのインターフェイスをアップ状態にします。 **vrf** キーワードは、VRF ごとの指定をイネーブルにします。
ステップ 7	設定する各外部サーバで、ステップ 2, （29 ページ）～ステップ 6, （30 ページ） を繰り返します。	—

ステップ 8	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーションファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

ステップ 9	show radius 例： `RP/0/RSP0/CPU0:router# show radius`	（任意）システムに設定されている RADIUS サーバの情報を表示します。

次の作業

ルータと RADIUS サーバとの通信を設定したら、RADIUS サーバ グループを設定します（RADIUS サーバ グループの設定, （41 ページ）の項を参照）。

RADIUS Dead サーバ検出の設定

RADIUS Dead-Server Detection 機能を設定します。

RADIUS Dead-Server Detection 機能を使用すると、RADIUS サーバをデッド状態と指定するための条件を決定できます。 条件が明示的に設定されていない場合は、条件は未処理のトランザクションの数に基づいて動的に計算されます。 RADIUS Dead-Server Detection を設定すると、応答を停止している RADIUS サーバが即時検出されます。 この未応答 RADIUS サーバの即時検出、動きが鈍いサーバの誤検出の回避、デッド状態とライブ状態を繰り返す現象の回避が有効になると、デッドタイムが短くなり、パケット処理が高速になります。

つまり、ルータが RADIUS サーバから有効なパケットを最後に受け取ってから RADIUS サーバがデッド状態と指定されるまでに経過する必要がある最低時間を秒単位で設定することができます。ルータが起動してからパケットの受信がなく、タイムアウトになると、時間基準は満たされたものとして処理されます。

さらに、RADIUS サーバがデッド状態と指定されるまでにルータで発生する必要がある連続タイムアウト回数を設定することもできます。 サーバが認証とアカウンティングの両方を実行する場合、両方の種類のパケットがこの回数に含まれます。 正しく作成されていないパケットは、タイムアウトになっているものとしてカウントされます。 カウントされるのは再転送だけで、最初の転送はカウントされません。 たとえば、タイムアウトになるたびに再転送が 1 回行われることになります。

（注） 時間の条件と試行回数の条件の両方を満たしていないと、サーバはデッド状態と指定されません。

radius-server deadtime コマンドは、サーバがデッド状態と指定され、その状態を維持する時間を分数で指定します。この時間を過ぎると、サーバから応答がない場合でも、アライブ状態と指定されます。 デッド条件が設定されていても、**radius-server deadtime** コマンドが設定されない限り、サーバはモニタされません。

手順の概要

1. **configure**
2. **radius-server deadtime** *minutes*
3. **radius-server dead-criteria time** *seconds*
4. **radius-server dead-criteria tries** *tries*
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
6. **show radius dead-criteria host** *ip-addr* [**auth-port** *auth-port*] [**acct-port** *acct-port*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**radius-server deadtime** *minutes* 例： `RP/0/RSP0/CPU0:router(config)# radius-server deadtime 5`	いくつかのサーバが使用不能になったときの RADIUS サーバの応答時間を短くし、使用不能になったサーバがすぐにスキップされるようにします。
ステップ 3	**radius-server dead-criteria time** *seconds* 例： `RP/0/RSP0/CPU0:router(config)# radius-server dead-criteria time 5`	デッド状態と指定される RADIUS サーバの dead-criteria 条件の時間を確立します。
ステップ 4	**radius-server dead-criteria tries** *tries* 例： `RP/0/RSP0/CPU0:router(config)# radius-server dead-criteria tries 4`	デッド状態と指定される RADIUS サーバの dead-criteria 条件の試行回数を確立します。

ステップ 5	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。

	コマンドまたはアクション	目的
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 6	**show radius dead-criteria host** *ip-addr* [**auth-port** *auth-port*] [**acct-port** *acct-port*] 例： `RP/0/RSP0/CPU0:router# show radius dead-criteria host 172.19.192.80`	（任意）指定 IP アドレスで RADIUS サーバに要求された dead-server-detection 情報を表示します。

Per VRF AAA の設定

Per VRF AAA 機能を使用すると、AAA サービスを VPN VPN ルーティングおよび転送（VRF）インスタンスに基づかせることができます。 プロバイダー エッジ（PE）または仮想ホーム ゲートウェイ（VHG）は、カスタマーの RADIUS サーバと通信します。このサーバは、カスタマーの VPN と関連付けられているため、RADIUS プロキシを介する必要はありません。 RADIUS プロキシを使用する必要がないため、ISP は、VPN による提供サービスをより効率的に拡張でき、カスタマーにさらに柔軟性を提供できます。

新しいベンダー固有の属性（VSA）

インターネット技術特別調査委員会（IETF）ドラフト標準には、ネットワーク アクセス サーバと RADIUS サーバの間でベンダー固有の属性（属性 26）を使用してベンダー固有の情報を伝達する方法が規定されています。 属性 26 はベンダー固有属性をカプセル化します。このため、ベンダーは一般的な用途に適さない独自の拡張属性をサポートできます。

Cisco IOS XR ソフトウェアの RADIUS 実装は、この仕様で推奨される形式を使用して、1 つのベンダー固有オプションをサポートしています。 シスコのベンダー ID は 9 で、サポートするオプションはベンダー タイプ 1、名前は「cisco-av-pair」です。値は次の形式のストリングです。

```
protocol : attribute sep value *
```

「protocol」は、特定の認可タイプに使用するシスコのプロトコル属性の値です。 「attribute」および「value」は、シスコの RADIUS 仕様で定義されている適切な属性値（AV）ペアです。「sep」は、必須の属性の場合は「=」、任意指定の属性の場合は「*」です。

次の表で、Per VRF AAA で現在サポートされている VSA について説明します。

表 *3* : ***Per VRF AAA*** でサポートされている ***VSA***

VSA 名	値の種類	説明
（注） RADIUS VSA（rad-serv、rad-serv-source-if および rad-serv-vrf）は、VSA 名の前にプレフィックス「aaa:」が必要です。		

rad-serv	string	サーバおよびサーバのグループの IP アドレス、キー、タイムアウトおよび再転送回数を示します。
		次に、VSA 構文を示します。
		`rad-serv=a.b.c.d [key SomeKey] [auth-port X] [acct-port Y]`
		`[retransmit V] [timeout W].`
		IP アドレス以外、すべてのパラメータはオプションで、任意の順序で発行されます。オプションのパラメータが指定されていない場合、デフォルト値が使用されます。
		キーには、スペースを含めることはできません。「retransmit V」の「V」は、1 ～ 100 の値で、「timeout W」の「W」は 1 ～ 1000 の値です。

rad-serv-vrf	string	RADIUS パケットの転送に使用される VRF の名前を指定します。 VRF 名は、**vrf** コマンドを介して指定された名前と一致します。

VRF ごとの RADIUS サーバ グループを設定します。 VRF ごとの TACACS+ サーバ グループの設定については、TACACS+ サーバ グループの設定, （43 ページ）を参照してください。

手順の概要

1. **configure**
2. **aaa group server radius** *group-name*
3. **server-private** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
4. **vrf** *vrf-name*
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ 2	**aaa group server radius** *group-name*
	
	例：
	`RP/0/RSP0/CPU0:router(config)# aaa group server radius radgroup1`
	`RP/0/RSP0/CPU0:router(config-sg-radius)#`
		各種サーバ ホストを別個のリストにグループ化し、サーバ グループ コンフィギュレーション モードを開始します。

ステップ 3	**server-private** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
	
	例：
	`RP/0/RSP0/CPU0:router(config-sg-radius)# server-private 10.1.1.1 timeout 5`
	`RP/0/RSP0/CPU0:router(config-sg-radius)# server-private 10.2.2.2 retransmit 3`
		グループに対するプライベート RADIUS サーバの IP アドレスを設定します。
		プライベート サーバ パラメータが指定されていない場合、グローバル コンフィギュレーションが使用されます。 グローバル コンフィギュレーションが指定されていない場合、デフォルト値が使用されます。
		auth-port キーワードと **acct-port** キーワードのどちらを使用しても、RADIUS サーバグループ プライベート コンフィギュレーション モードが開始されます。

ステップ 4	**vrf** *vrf-name* 例： `RP/0/RSP0/CPU0:router(config-sg-radius)# vrf v2.44.com`	AAA RADIUS サーバ グループの VRF 参照を設定します。 （注） プライベート サーバ IP アドレスは、グローバルで設定されているアドレスとオーバーラップすることがあります。VRF 定義は、このような場合に、アドレスを区別するときに役に立ちます。
ステップ 5	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーションファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

TACACS+ サーバの設定

TACACS+ サーバを設定します。

ポートが指定されていない場合、標準ポート番号 49 がデフォルトで使用されます。 **timeout** および **key** パラメータは、すべての TACACS+ サーバに対してグローバルで指定できます。 **timeout** パラメータは、AAA サーバが TACACS+ サーバから応答を受信するまでの時間を指定します。 **key** パラメータは、AAA サーバと TACACS+ サーバ間で共有される認証および暗号キーを指定します。

手順の概要

1. **configure**
2. **tacacs-server host** *host-name* **port** *port-number*
3. **tacacs-server host** *host-name* **timeout** *seconds*
4. **tacacs-server host** *host-name* **key** [**0** | **7**] *auth-key*
5. **tacacs-server host** *host-name* **single-connection**
6. **tacacs source-interface** *type instance* **vrf** *vrf-name*
7. 設定する各外部サーバで、ステップ 2, （38 ページ）～ステップ 5, （39 ページ） を繰り返します。
8. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
9. **show tacacs**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ 2	**tacacs-server host** *host-name* **port** *port-number*
	
	例：
	
	`RP/0/RSP0/CPU0:router(config)# tacacs-server host 209.165.200.226 port 51`
	`RP/0/RSP0/CPU0:router(config-tacacs-host)#`
		TACACS+ ホスト サーバを指定し、オプションでサーバ ポート番号を指定します。
		
		• このオプションによって、デフォルトのポート 49 は上書きされます。 有効なポート番号の範囲は 1 ～ 65535 です。

ステップ 3	**tacacs-server host** *host-name* **timeout** *seconds*
	
	例：
	
	`RP/0/RSP0/CPU0:router(config-tacacs-host)# tacacs-server host 209.165.200.226 timeout 30`
	`RP/0/RSP0/CPU0:router(config)#`
		TACACS+ ホスト サーバを指定し、オプションで、AAA サーバが TACACS+ サーバからの応答を待機する時間の長さを設定するタイムアウト値を指定します。
		
		• このオプションによって、**tacacs-server timeout** コマンドで設定したグローバル タイムアウト値がこのサーバに限り上書きされます。 タイムアウト値は、タイムアウト間隔を指定する整数として表されます。 範囲は 1 ～ 1000 です。

	コマンドまたはアクション	目的
ステップ 4	**tacacs-server host** *host-name* **key** [**0** | **7**] *auth-key* 例： `RP/0/RSP0/CPU0:router(config)# tacacs-server host 209.165.200.226 key 0 a_secret`	TACACS+ ホスト サーバを指定し、オプションで、AAA サーバと TACACS+ サーバ間で共有される認証および暗号キーを指定します。 • TACACS+ パケットは、このキーを使って暗号化されます。 このキーは TACACS+ デーモンで使用されるキーと一致する必要があります。 このキーを指定すると、このサーバに限り、**tacacs-server key** コマンドで設定されているグローバル キーが上書きされます。 •（任意）**0** の入力により、暗号化されていない（クリアテキスト）キーが続くことを指定します。 •（任意）**7** の入力により、暗号キーが続くことを指定します。 • *auth-key* 引数は、AAA サーバと TACACS+ サーバ間で共有される暗号化または復号化されるキーを指定します。
ステップ 5	**tacacs-server host** *host-name* **single-connection** 例： `RP/0/RSP0/CPU0:router(config)# tacacs-server host 209.165.200.226 single-connection`	単一 TCP 接続を介してすべての TACACS+ 要求をこのサーバに多重化するようにルータを設定します。 デフォルトでは、セッションごとに別個の接続が使用されます。
ステップ 6	**tacacs source-interface** *type instance* **vrf** *vrf-name* 例： `RP/0/RSP0/CPU0:router(config)# tacacs source-interface GigabitEthernet 0/4/0/0 vrf abc`	（任意）すべての発信 TACACS+ パケットに対して、選択したインターフェイスの発信元 IP アドレスを指定します。 • 指定されたインターフェイスまたはサブインターフェイスには、IP アドレスが関連付けられている必要があります。 指定のインターフェイスまたはサブインターフェイスに IP アドレスが設定されていないか、そのインターフェイスがダウン状態にある場合、TACACS+ はデフォルト インターフェイスに戻ります。 これを回避するには、インターフェイスまたはサブインターフェイスに IP アドレスを追加するか、そのインターフェイスをアップ状態にします。

	コマンドまたはアクション	目的
		• **vrf** オプションは、AAA TACACS+ サーバ グループのバーチャル プライベート ネットワーク（VPN）ルーティングおよび転送（VRF）参照を指定します。
ステップ 7	設定する各外部サーバで、ステップ 2, （38 ページ）～ステップ 5, （39 ページ） を繰り返します。	—

ステップ 8	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

ステップ 9	**show tacacs** **例**： `RP/0/RSP0/CPU0:router# show tacacs`	（任意）システムに設定されている TACACS+ サーバの情報を表示します。

次の作業

TACACS+ サーバを設定したら、TACACS+ サーバ グループを設定します（TACACS+ サーバ グループの設定, （43 ページ）の項を参照）。

RADIUS サーバ グループの設定

RADIUS サーバ グループを設定します。

ユーザは、1 つ以上の **server** コマンドを入力できます。 **server** コマンドは、外部 RADIUS サーバのホスト名またはIPアドレスおよびポート番号を指定します。設定されている場合、このサーバグループは、AAA方式リスト（認証、認可またはアカウンティングの設定に使用されます）から参照できます（方式リスト, （9 ページ）の項を参照）。

はじめる前に

正常に設定を行うため、外部サーバが設定時にアクセスできる必要があります。

手順の概要

1. **configure**
2. **aaa group server radius** *group-name*
3. **server** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*]
4. ステップ 3, （42 ページ）で指定したサーバ グループに追加するすべての外部サーバで、ステップ 4, （42 ページ）を繰り返します。
5. **server-private** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
6. **deadtime** *minutes*
7. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
8. **show radius server-groups** [*group-name* [**detail**]]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**aaa group server radius** *group-name* 例： `RP/0/RSP0/CPU0:router(config)# aaa group server radius radgroup1`	各種サーバ ホストを別個のリストにグループ化し、サーバ グループ コンフィギュレーション モードを開始します。
ステップ 3	**server** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] 例： `RP/0/RSP0/CPU0:router(config-sg-radius)# server 192.168.20.0`	外部 RADIUS サーバのホスト名または IP アドレスを指定します。 • サーバ グループは、設定されると、AAA 方式リスト（認証、認可またはアカウンティングの設定に使用されます）から参照できます。
ステップ 4	ステップ 3, （42 ページ）で指定したサーバ グループに追加するすべての外部サーバで、ステップ 4, （42 ページ）を繰り返します。	—
ステップ 5	**server-private** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*] 例： `RP/0/RSP0/CPU0:router(config-sg-radius)# server-private 10.10.130.2 auth-port 1600 acct-port 1666 key code`	グループ サーバに対するプライベート RADIUS サーバの IP アドレスを設定します。 （注） プライベート サーバ パラメータが指定されていない場合、グローバル コンフィギュレーションが使用されます。 グローバル コンフィギュレーションが指定されていない場合、デフォルト値が使用されます。
ステップ 6	**deadtime** *minutes* 例： `RP/0/RSP0/CPU0:router(config-sg-radius)# deadtime 1`	RADIUS サーバ グループ レベルでデッドタイム値を設定します。 • *minutes* 引数は、RADIUS サーバがトランザクション要求によってスキップされる時間を最長 1440（24 時間）まで分単位で指定します。 有効な範囲は 1 ～ 1440 です。 この例では、RADIUS サーバ グループ radgroup1 が認証要求への応答に失敗したときの **deadtime** コマンドに対して、1 分のデッドタイムを指定します。 （注） グループを作成したら、グループ レベルのデッドタイムを設定できます。
ステップ 7	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ステップ 8	**show radius server-groups** [*group-name* [**detail**]] 例： `RP/0/RSP0/CPU0:router# show radius server-groups`	（任意）システムで設定されている各 RADIUS サーバ グループの情報を表示します。

次の作業

RADIUS サーバ グループを設定したら、認証、認可およびアカウンティングを設定して方式リストを定義します （AAA 方式リストの設定, （46 ページ）の項を参照）。

TACACS+ サーバ グループの設定

TACACS+ サーバ グループを設定します。

1 つ以上の **server** コマンドを入力できます。 **server** コマンドは、外部 TACACS+ サーバのホスト名または IP アドレスを指定します。 設定されている場合、このサーバ グループは、AAA 方式リスト（認証、認可またはアカウンティングの設定に使用されます）から参照できます （方式リスト, （9 ページ）の項を参照）。

はじめる前に

正常に設定を行うため、外部サーバが設定時にアクセスできる必要があります。 グローバルおよび vrf 設定で同じ IP アドレスを設定する場合、server-private パラメータが必要です。

手順の概要

1. **configure**
2. **aaa group server tacacs+** *group-name*
3. **server** {*hostname* | *ip-address*}
4. ステップ 2, （44 ページ）で指定したサーバ グループに追加するすべての外部サーバで、ステップ 3, （44 ページ）を繰り返します。
5. **server-private** {*hostname* | *ip-address*} [**port** *port-number*] [**timeout** *seconds*] [**key** *string*]
6. **vrf** *vrf-name*
7. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
8. **show tacacs server-groups**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**aaa group server tacacs+** *group-name* 例： `RP/0/RSP0/CPU0:router(config)# aaa group server tacacs+ tacgroup1`	各種サーバ ホストを別個のリストにグループ化し、サーバ グループ コンフィギュレーション モードを開始します。
ステップ 3	**server** {*hostname* | *ip-address*} 例： `RP/0/RSP0/CPU0:router(config-sg-tacacs+)# server 192.168.100.0`	外部 TACACS+ サーバのホスト名または IP アドレスを指定します。 • 設定されている場合、このグループは、AAA 方式リスト（認証、認可またはアカウンティングの設定に使用されます）から参照できます

	コマンドまたはアクション	目的
ステップ 4	ステップ 2, （44 ページ）で指定したサーバ グループに追加するすべての外部サーバで、ステップ 3, （44 ページ）を繰り返します。	—
ステップ 5	**server-private** {*hostname* | *ip-address*} [**port** *port-number*] [**timeout** *seconds*] [**key** *string*] 例： `RP/0/RSP0/CPU0:router(config-sg-tacacs+)# server-private 10.1.1.1 key a_secret`	グループ サーバに対するプライベート TACACS+ サーバの IP アドレスを設定します。 （注） プライベート サーバ パラメータが指定されていない場合、グローバル コンフィギュレーションが使用されます。グローバル コンフィギュレーションが指定されていない場合、デフォルト値が使用されます。
ステップ 6	**vrf** *vrf-name* 例： `RP/0/RSP0/CPU0:router(config-sg-tacacs+)# vrf abc`	AAA TACACS+ サーバ グループのバーチャル プライベート ネットワーク（VPN）ルーティングおよび転送（VRF）参照情報を設定します。

ステップ 7	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレー

	コマンドまたはアクション	目的
		ション セッションを継続するには、**commit** コマンドを使用します。
ステップ 8	**show tacacs server-groups** 例： `RP/0/RSP0/CPU0:router# show tacacs server-groups`	（任意）システムで設定されている各 TACACS+ サーバ グループの情報を表示します。

次の作業

TACACS+ サーバ グループを設定したら、認証、認可およびアカウンティングを設定して方式リストを定義します （AAA 方式リストの設定, （46 ページ）の項を参照）。

AAA 方式リストの設定

AAA データは、さまざまなデータ ソースに保存できます。 AAA 設定は、方式リストを使用して、AAA データのソースの優先順位を定義します。 AAA は、複数の方式リストを定義でき、アプリケーション（ログインなど）は、これらのいずれかを選択できます。 たとえば、コンソールおよびAUX ポートと VTY ポートで異なる方式リストを使用できます。 方式リストが指定されていない場合、アプリケーションは、デフォルトの方式リストを使用します。

この項では、次の手順について説明します。

認証方式リストの設定

認証の方式リストを設定します。

認証設定

認証は、ユーザ（またはプリンシパル）が検証されるプロセスです。 認証設定は、方式リストを使用して、さまざまなデータ ソースに保存されている、AAA データ ソースの優先順位を定義します。 認証を設定して、複数の方式リストを定義できます。アプリケーションは（ログインなど）、これらのいずれかを選択できます。 たとえば、コンソールおよびAUX ポートと VTY ポートで異なる方式リストを使用できます。 方式リストが指定されていない場合、アプリケーションは、デフォルトの方式リストを使用します。

（注） アプリケーションは、有効な方式リストを選択するため、定義済み方式リストを明示的に参照する必要があります。

認証は、**login authentication** 回線コンフィギュレーション サブモード コマンドを使用して、TTY 回線に適用できます。

一連の認証方式の作成

aaa authentication コマンドを使用して、一連の認証方式、つまり方式リストを作成します。 方式リストは、シーケンスで照会される認証方式（RADIUS、TACACS+など）を説明する単なる名前付きリストです。 方式は次のいずれかです。

- group radius：サーバ グループまたは RADIUS サーバを認証に使用します
- group tacacs+：サーバ グループまたは TACACS+ サーバを認証に使用します
- local：ユーザ名またはパスワードのローカル データベースを認証に使用します。
- line：回線パスワードまたはユーザ グループを認証に使用します。

方式が、サーバ グループではなく、RADIUS または TACACS+ サーバの場合、RADIUS または TACACS+ サーバは、設定されている RADIUS および TACACS+ サーバのグローバル プールから、設定順に選択されます。 このグローバル プールから選択されるサーバは、サーバ グループに追加できるサーバです。

後続の認証方式は、初期方式がエラーを返すか、要求が拒否された場合だけ使用されます。

はじめる前に

（注） デフォルトの方式リストは、認証のすべてのインターフェイスに適用されます。ただし、デフォルト以外の方式リストが明示的に設定されている場合は例外で、この場合は、指定されている方式リストが適用されます。

group radius、**group tacacs+** および **group** *group-name* 形式の **aaa authentication** コマンドは、事前定義されている RADIUS または TACACS+ サーバのセットを参照します。 ホスト サーバを設定するには、**radius-server host** コマンドまたは tacacs-server host コマンドを使用します。サーバの名前付きグループを作成するには、**aaa group server radius** コマンドまたは aaa group server tacacs+ コマンドを使用します。

手順の概要

1. **configure**
2. **aaa authentication** {**login** | **ppp**} {**default** | *list-name* | **remote**} *method-list*
3. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
4. 設定されるすべての認証方式リストに対して、ステップ 1 ～ 3 を繰り返します。

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**aaa authentication** {**login** | **ppp**} {**default** | *list-name* | **remote**} *method-list* 例： `RP/0/RSP0/CPU0:router(config)# aaa authentication login default group tacacs+`	一連の認証方式、つまり方式リストを作成します。 • **login** キーワードを使用すると、ログインの認証が設定されます。 **ppp** キーワードを使用すると、ポイントツーポイント プロトコルの認証が設定されます。 • **default** キーワードを入力すると、このキーワードの後のリストされている認証方式が、認証のデフォルト方式リストになります。 • *list-name* 文字列を入力すると、認証方式リストが識別されます。 • **remote** キーワードを入力すると、このキーワードの後のリストされている認証方式が、所有者以外のリモート SDR の管理認証のデフォルト方式リストになります。 （注） remote キーワードは管理プレーンでだけ使用できます。 • *method-list* 引数の後に方式リスト タイプを入力します。 方式リスト タイプは、目的の順序で入力します。 リストされる方式タイプは、次のいずれかのオプションです。 ◦ **group tacacs+**：サーバ グループまたは TACACS+ サーバを認証に使用します ◦ **group radius**：サーバ グループまたは RADIUS サーバを認証に使用します ◦ **group** *named-group*：TACACS+ サーバまたは RADIUS サーバの名前付きサブセットが認証に使用されます ◦ **local**：ユーザ名またはパスワードのローカル データベースを認証に使用します ◦ **line**：回線パスワードまたはユーザ グループを認証に使用します • この例では、**default** 方式リストが認証に使用されます
ステップ 3	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

ステップ 4	設定されるすべての認証方式リストに対して、ステップ 1 ～ 3 を繰り返します。	—

次の作業

認証方式リストを設定したら、認可方式リストを設定します。 （認可方式リストの設定, （49 ページ）の項を参照）。

認可方式リストの設定

認可方式リストを設定します。

（注） **radius** キーワードを **aaa authorization** コマンドで設定できます。

認可の設定

許可方式リストによって、許可の実行方法とこれらの方式の実行順序が定義されます。 方式リストは、一連の認可方式（TACACS+ など）を記述した名前付きリストです。 方式リストは、認可に使用するセキュリティ プロトコルを 1 つ以上指定できるため、最初の方式が失敗した場合のバックアップ システムを確保できます。 Cisco IOS XR ソフトウェアでは、特定のネットワーク サービスに対してユーザを許可するために、リスト内の最初の方式が使用されます。この方式が応答に失敗すると、Cisco IOS XR ソフトウェアでは方式リスト内の次の方式が選択されます。 このプロセスは、リスト内の認可方式との通信に成功するまで、または定義されている方式を使い果たすまで続行されます。

（注） Cisco IOS XR ソフトウェアでは、前の方式から応答がない（障害ではない）場合にだけ、次に指定された方式を使って認可が試みられます。 このサイクルの任意の時点で認可が失敗した場合（つまり、セキュリティ サーバまたはローカル ユーザ名データベースからユーザ サービスの拒否応答が返される場合）、認可プロセスは停止し、その他の認可方式は試行されません。

方式リストは、要求されている許可のタイプによって異なります。 Cisco IOS XR ソフトウェアは、次の 4 つのタイプの AAA 許可をサポートします。

- コマンドの認可：ユーザが実行する EXEC モード コマンドに適用されます。 コマンドの許可では、すべての EXEC モード コマンドに対する許可が試みられます。

  （注） 「コマンド」の認可は、認証中に確立されるタスク プロファイルに基づく「タスクベース」の認可とは異なります。

- EXEC の認可：EXEC セッションの開始に対する認可が適用されます。

  （注） **exec** キーワードは、障害マネージャ サービスの許可に使用されなくなりました。 障害マネージャ サービスの許可には、**eventmanager** キーワード（障害マネージャ）を使用します。 **exec** キーワードは、EXEC の許可に使用します。

- ネットワークの認可：IKE などのネットワーク サービスの認可が適用されます。
- イベント マネージャの認可：イベント マネージャ（障害マネージャ）を認可するための認可方式が適用されます。 RADIUS サーバは、イベント マネージャ（障害マネージャ）認可に設定できません。 TACACS+ を使用することも、locald を使用することもできます。

名前付き方式リストを作成すると、指定した許可タイプに対して特定の許可方式リストが定義されます。 方式リストを定義した場合、定義した方式のいずれかを実行するには、まず特定の回線またはインターフェイスに方式リストを適用する必要があります。 新しい方式リストを作成する場合、TACACS+ などの方式の名前は使用しないでください。

「コマンド」の認可は、コマンドの認可方式リストの回線テンプレートへの追加の結果として、ルータで自動的に実行される「タスクベース」の認可とは区別されます。 コマンド認可のデフォルト動作はnoneです。デフォルトの方式リストが設定されている場合でも、この方式リストを使用するために回線テンプレートに追加する必要があります。

aaa authorization commands コマンドを使用すると、一連の属性値（AV）ペアを含む要求パケットが、認可プロセス中に TACACS+ デーモンに送信されます。 デーモンは、次のいずれかを実行できます。

- 要求をそのまま受け取る。
- 認可を拒否する。

一連の認可方式の作成

aaa authorization コマンドを使用して、認可パラメータを設定し、各回線またはインターフェイスで使用できる特定の認可方式を定義する名前付きの方式リストを作成します。

Cisco IOS XR ソフトウェアは、次の許可方式をサポートします。

- **none**：ルータから認可情報の要求はありません。この回線やインターフェイスに対する認可は行われません。
- **local**：ローカル データベースを認可に使用します。
- **group tacacs+**：設定されているすべての TACACS+ サーバを認可に使用します。
- **group radius**：設定されているすべての RADIUS サーバのリストを認可に使用します。
- **group** *group-name*：TACACS+ サーバまたは サーバの名前付きサブセットを認可に使用します。

手順の概要

1. **configure**
2. **aaa authorization** {**commands** | **eventmanager** | **exec** | **network**} {**default** | *list-name*} {**none** | **local** | **group** {**tacacs+** | **radius** | *group-name*}}
3. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**aaa authorization** {**commands** | **eventmanager** | **exec** | **network**} {**default** | *list-name*} {**none** | **local** | **group** {**tacacs+** | **radius** | *group-name*}} 例： `RP/0/RSP0/CPU0:router(config)# aaa authorization commands listname1 group tacacs+`	一連の認可方式、つまり方式リストを作成します。 • **commands** キーワードは、すべての EXEC シェル コマンドの認可を設定します。 コマンドの認可は、ユーザにより発行される EXEC モード コマンドに適用されます。 コマンドの許可では、すべての EXEC モード コマンドに対する許可が試みられます。 • **eventmanager** キーワードは、イベント マネージャ（障害マネージャ）を認可するための認可方式を適用します。 • **exec** キーワードは、インタラクティブ（EXEC）セッションの認可を設定します。 • **network** キーワードは、PPP または IKE のようなネットワーク サービスの認可を設定します。 • **default** キーワードを入力すると、このキーワードの後のリストされている認可方式が、認可のデフォルト方式リストになります。 • *list-name* 文字列を入力すると、認可方式リストが識別されます。 方式リスト自体は、方式リスト名に続きます。 方式リスト タイプは、目的の順序で入力します。 リストされる方式リスト タイプは、次のいずれかにできます。 ◦ **none**：ネットワーク アクセス サーバ（NAS）は、認可情報を要求しません。 認可は常に成功します。 以降の認可方式は試行されません。 ただし、タスク ID の許可は常に必要であり、ディセーブルにはできません。 ◦ **local**：ローカル データベースを認可に使用します。 • **group tacacs+**：設定されているすべての TACACS+ サーバを認可に使用します。 NAS は、認可情報を TACACS+ セキュリティ デーモンと交換します。 TACACS+ 認可は、AV ペアを関連付けることでユーザに特定の権限を定義します。AV は適切なユーザとともに TACACS+ セキュリティ サーバのデータベースに保存されます。 • **group radius**：設定されているすべての RADIUS サーバのリストを認可に使用します。

	コマンドまたはアクション	目的
		• **group** *group-name*：**aaa group server tacacs+** または **aaa group server radius** コマンドで定義されているとおりに、TACACS+ サーバまたは RADIUS サーバのサブセット、名前付きサーバ グループを認可に使用します。

ステップ 3	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

次の作業

認可方式リストを設定したら、アカウンティング方式リストを設定します（アカウンティング方式リストの設定, （53 ページ）の項を参照）。

アカウンティング方式リストの設定

アカウンティング方式リストを設定します。

（注） **radius** キーワードを **aaa accounting** コマンドで設定できます。

アカウンティングの設定

現在、Cisco IOS XR ソフトウェアは、アカウンティングで TACACS+ および RADIUS 方式の両方をサポートしています。 ルータは、アカウンティング レコードの形式で TACACS+ または RADIUS セキュリティ サーバにユーザ アクティビティを報告します。 各アカウンティング レコードは、アカウンティング AV ペアが含まれ、セキュリティ サーバ上で保管されます。

アカウンティング方式リストには、アカウンティングの実行方法が定義されます。このリストを使用して、特定のタイプのアカウンティング サービスに固有の回線またはインターフェイスに使用する特定のセキュリティプロトコルを指定できます。方式リストの名前を付ける場合、TACACS+ などの方式の名前を使用しないでください。

最小のアカウンティングの場合、**stop-only** キーワードを指定して、要求されたユーザプロセスの終了時に「stop accounting」通知を送信します。最小より大きいアカウンティングの場合、**start-stop** キーワードを使用できます。これにより、外部 AAA サーバが、要求されたプロセスの開始時に「start accounting」通知を送信し、プロセスの終了時に「stop accounting」通知を送信します。 また、**aaa accounting update** コマンドを使用して、累積情報による更新レコードを定期的に送信できます。 アカウンティング レコードは、TACACS+ または RADIUS サーバだけに格納されます。

AAA アカウンティングをアクティブにすると、ルータは、これらの属性をアカウンティング レコードとして報告します。そのアカウンティング レコードは、セキュリティ サーバ上のアカウンティング ログに格納されます。

一連のアカウンティング方式の作成

aaa accounting コマンドを使用して、各回線またはインターフェイスで使用できる特定のアカウンティング方式を定義するデフォルトまたは名前付き方式リストを作成します。

Cisco IOS XR ソフトウェアは、次のアカウンティング方式をサポートします。

- none：アカウンティングは、この回線またはインターフェイスで実行されません。
- group tacacs+：設定されているすべての TACACS+ サーバをアカウンティングに使用します。
- group radius：設定されているすべての RADIUS サーバのリストをアカウンティングに使用します。

手順の概要

1. **configure**
2. 次のいずれかを実行します。
   - **aaa accounting** {**commands** | **exec** | **network**} {**default** | *list-name*} {**start-stop** | **stop-only**}
   - {none | method}
3. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	次のいずれかを実行します。 • **aaa accounting** {**commands** | **exec** | **network**} {**default** | *list-name*} {**start-stop** | **stop-only**} • {none | method} 例： `RP/0/RSP0/CPU0:router(config)# aaa accounting commands default stop-only group tacacs+`	一連のアカウンティング方式、つまり方式リストを作成します。 • **commands** キーワードは、EXEC シェル コマンドでアカウンティングをイネーブルにします。 • **exec** キーワードを使用して、対話型（EXEC）セッションに対するアカウンティングをイネーブルにします。 • **network** キーワードは、ポイントツーポイント プロトコル（PPP）など、すべてのネットワーク関連サービス要求のアカウンティングをイネーブルにします。 • **default** キーワードを入力すると、このキーワードの後のリストされているアカウンティング方式が、アカウンティングのデフォルト方式リストになります。 • *list-name* 文字列を入力すると、アカウンティング方式リストが識別されます。 • **start-stop** キーワードは、プロセスの開始時に「start accounting」通知を送信し、プロセスの終了時に「stop accounting」通知を送信します。 要求されたユーザ プロセスは、「start accounting」通知をアカウンティング サーバから受信したかどうかにかかわらず開始されます。 • **stop-only** キーワードは、要求されたユーザ プロセスの終了時に「stop accounting」通知を送信します。 • **none** キーワードは、アカウンティングが実行されないことを示します。 • 方式リスト自体は、**start-stop** キーワードの後に続きます。 方式リスト タイプは、目的の順序で入力します。 方式引数は、次のタイプをリストします。 ◦ **group tacacs+**：設定されているすべての TACACS+ サーバをアカウンティングに使用します。 ◦ **group radius**：設定されているすべての RADIUS サーバのリストをアカウンティングに使用します。 ◦ **group** *group-name*：**aaa group server tacacs+** または **aaa group server radius** コマンドで定義されているとおりに、TACACS+ サーバまた

	コマンドまたはアクション	目的
		はRADIUS サーバのサブセット、名前付きサーバ グループをアカウンティングに使用します。 • 次の例では、アカウンティング サービスが TACACS+セキュリティ サーバで提供され、stop-only 制限がある **default** コマンド アカウンティング方式リストの定義を示します。

ステップ 3	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

次の作業

方式リストを設定したら、これらの方式リストを適用します（アプリケーションの方式リストの適用，（58 ページ）の項を参照）。

中間アカウンティング レコードの生成

アカウンティング サーバに送信される定期的中間アカウンティング レコードをイネーブルにします。 **aaa accounting update** コマンドをアクティブにすると、Cisco IOS XR ソフトウェアは、システム上のすべてのユーザに中間アカウンティング レコードを発行します。

（注） 中間アカウンティング レコードは、インターネット キー交換（IKE）アカウンティングなど、ネットワーク セッションだけで生成されます。これは、**network** キーワードを指定した **aaa accounting** コマンドで制御されます。 システム、コマンドまたはEXECアカウンティング セッションでは、中間レコードは生成されません。

手順の概要

1. **configure**
2. **aaa accounting update** {**newinfo** | **periodic** *minutes*}
3. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**aaa accounting update** {**newinfo**	**periodic** *minutes*} 例： `RP/0/RSP0/CPU0:router(config)# aaa accounting update periodic 30`	アカウンティング サーバに送信される定期的中間アカウンティング レコードをイネーブルにします。 • **newinfo** キーワードを使用すると、報告する新しいアカウンティング情報があるたびに中間アカウンティング レコードがアカウンティング サーバに送信されます。 たとえば、IPCP がリモート ピアとの間でIPアドレスのネゴシエーションを完了したときなどです。 中間アカウンティング レコードには、リモート ピアに使用されるネゴシエート済み IP アドレスが含まれます。 • **periodic** キーワードを使用すると、中間アカウンティング レコードは number 引数で定義されているとおりに定期的に送信されます。中間アカウンティング レコードには、中間アカウンティング レコードが送信される時間までに、そのユーザについて記録されたすべてのアカウンティング情報が含まれます。 注意 **periodic** キーワードを使用すると、多数のユーザがネットワークにログインしているときに、大きな輻輳が生じる場合があります。

	コマンドまたはアクション	目的

ステップ 3	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

アプリケーションの方式リストの適用

認可およびアカウンティング サービスの方式リストを設定したら、これらのサービスを使用するアプリケーション（コンソール、vty、補助など）に、設定した方式リストを適用できます。 方式リストの適用するには、AAA 認可およびアカウンティングをイネーブルします。

この項では、次の手順について説明します。

AAA 認可のイネーブル化

AAA 認可を特定の回線または回線のグループに対してイネーブルにします。

方式リストの適用

aaa authorization コマンドを使用して、特定のタイプの認可に対して名前付き認可方式リストを定義（またはデフォルトの方式リストを使用）したあと、認可を実行する該当の回線に、定義済みのリストを適用する必要があります。 **authorization** コマンドを使用して、指定の方式リスト

（または、方式リストを指定していない場合はデフォルトの方式リスト）を選択した回線または回線グループに適用します。

手順の概要

1. **configure**
2. **line** {**aux** | **console** | **default** | **template** *template-name*}
3. **authorization** {**commands** | **exec**} {**default** | *list-name*}
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**line** {**aux**	**console**	**default**	**template** *template-name*} 例： `RP/0/RSP0/CPU0:router(config)# line console`	回線テンプレート コンフィギュレーション モードを開始します。
ステップ 3	**authorization** {**commands**	**exec**} {**default**	*list-name*} 例： `RP/0/RSP0/CPU0:router(config-line)# authorization commands listname5`	AAA認可を特定の回線または回線のグループに対してイネーブルにします。 • **commands** キーワードは、すべてのコマンドに対して、選択した回線における認可をイネーブルにします。 • **exec** キーワードを使用して、対話型（EXEC）セッションに対する認可をイネーブルにします。 • **default** キーワードを入力し、**aaa authorization** コマンドで定義されているように、デフォルトの方式リストの名前を適用します。 • 使用する認可方式リストの名前を入力します。 リスト名を指定しない場合は、デフォルト名が使用されます。 リストは**aaa authorization** コマンドで作成されます。 • 次に、方式リスト listname5 を使用したコマンド認可の例を示します。

	コマンドまたはアクション	目的

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

次の作業

AAA 認可をイネーブルにして認可方式リストを適用したら、AAA アカウンティングをイネーブルにしてアカウンティング方式リストを適用します（アカウンティング サービスのイネーブル化，（60 ページ）の項を参照）。

アカウンティング サービスのイネーブル化

アカウンティング サービスを特定の回線または回線のグループに対してイネーブルにします。

手順の概要

1. **configure**
2. **line {aux | console ｜ default | template template-name}**
3. **accounting** {**commands** | **exec**} {**default** | *list-name*}
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**line {aux | console ｜ default | template template-name}** 例： `RP/0/RSP0/CPU0:router(config)# line console`	回線テンプレート コンフィギュレーション モードを開始します。
ステップ 3	**accounting** {**commands** | **exec**} {**default** | *list-name*} 例： `RP/0/RSP0/CPU0:router(config-line)# accounting commands listname7`	AAA アカウンティングを特定の回線または回線のグループに対してイネーブルにします。 • **commands** キーワードは、すべての EXEC シェル コマンドに対して、選択した回線におけるアカウンティングをイネーブルにします。 • **exec** キーワードを使用して、対話型（EXEC）セッションに対するアカウンティングをイネーブルにします。 • **default** キーワードを入力し、**aaa accounting** コマンドで定義されているように、デフォルトの方式リストの名前を適用します。 • 使用するアカウンティング方式リストの名前を指定します。 リスト名を指定しない場合は、デフォルト名が使用されます。 リストは、**aaa accounting** コマンドで作成されます。 • 次に、方式リスト listname7 を使用したコマンド アカウンティングの例を示します。

	コマンドまたはアクション	目的

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

次の作業

AAA アカウンティング サービスをイネーブルにしてアカウンティング方式リストを適用したら、ログイン パラメータを設定します（ログイン パラメータの設定, （62 ページ）の項を参照）。

ログイン パラメータの設定

サーバがログインの応答を待機する間隔を設定します。

手順の概要

1. **configure**
2. **line template** *template-name*
3. **timeout login response** *seconds*
4. 次のいずれかのコマンドを使用します。
   • **end**
   • **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**line template** *template-name* 例： `RP/0/RSP0/CPU0:router(config)# line template alpha`	設定する回線を指定し、回線テンプレート コンフィギュレーション モードを開始します。
ステップ 3	**timeout login response** *seconds* 例： `RP/0/RSP0/CPU0:router(config-line)# timeout login response 20`	サーバがログインの応答を待機する間隔を設定します。 • *seconds* 引数は、0 ～ 300 のタイムアウト間隔（秒数）を指定します。 デフォルトは 30 秒です。 • この例では、インターバル タイマーを 20 秒に変更します。

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

AAA サービスの設定の設定例

この項では、次の設定例について説明します。

AAA サービスの設定：例

次に、AAA サービスを設定する例を示します。

認証方式リスト vty-authen が設定されます。 この例では、設定されたすべての TACACS+ サーバのリストを認証に使用する方式リストを指定します。 この方式が失敗した場合、ローカルユーザ名データベース方式が認証に使用されます。

```
configure
aaa authentication login vty-authen group tacacs+ local
```

PPP のデフォルトの方式リストは、ローカル方式を使用するように設定されます。

```
aaa authentication ppp default local
```

ユーザ名 user1 が、ログイン目的で作成され、安全なログイン パスワードが割り当てられ、user1 が root-system ユーザになります。 ユーザ名 user2 でも同様に設定します。

```
username user1
secret lab
group root-system
exit

username user2
secret lab
exit
```

タスク グループ tga が作成され、タスクが tga に追加されます。ユーザ グループ uga が作成され、uga が、タスク グループ tga から権限を継承するように設定されます。 説明がタスク グループ uga に追加されます。

```
taskgroup tga
task read bgp
task write ospf
exit

usergroup uga
taskgroup tga
description usergroup uga
exit
```

ユーザ名 user2 が、ユーザ グループ uga から継承されます。

```
username user2
group uga
exit
```

3 台の TACACS サーバが設定されます。

```
tacacs-server host 10.1.1.1 port 1 key abc
tacacs-server host 10.2.2.2 port 2 key def
tacacs-server host 10.3.3.3 port 3 key ghi
```

ユーザ グループ priv5 が作成されます。これは、TACACS+ 方式で認証され、外部 TACACS+ デーモン コンフィギュレーション ファイルでのエントリの特権レベルは 5 です。

```
usergroup priv5
taskgroup operator
exit
```

認可方式リスト vty-author が設定されます。 次に、設定されているすべての TACACS+ サーバのリストを使用してコマンドが認可される例を示します。

```
aaa authorization commands vty-author group tacacs+
```

アカウンティング方式リスト vty-acct が設定されます。 次に、設定されているすべての TACACS+ サーバのリストを使用して start-stop コマンド アカウンティングが行われる例を示します。

```
aaa accounting commands vty-acct start-stop group tacacs+
```

TACACS+ 認証では、たとえば、特権レベル 8 が返され、ローカル ユーザ グループ priv8 が存在せず、同じ名前のローカル ユーザも存在しない場合、*taskgroup-name* 引数で tga を指定した **aaa default-taskgroup** コマンドを使用すると、このようなユーザにタスク グループ tga のタスクマップが提供されます。

```
aaa default-taskgroup tga
```

回線テンプレート vty に、回線パスワードが割り当てられます。これは、回線認証で使用され、ユーザ グループ uga を回線認証（使用される場合）に割り当てられるグループに指定します。また、vty-authen、vty-author および vty-acct をそれぞれ、認証、認可およびアカウンティングで使用される方式リストに指定します。

```
line template vty
password lab
users group uga
login authentication vty-authen
authorization commands vty-author
accounting commands vty-acct
exit
```

TACACS+ サーバ グループ abc が作成され、すでに設定されている TACACS+ サーバが追加されます。

```
aaa group server tacacs+ abc
server 10.3.3.3
exit
```

参考資料

ここでは、AAA サービスの設定に関連する参考資料について説明します。

関連資料

関連項目	ドキュメント名
AAA サービスのコマンド：完全なコマンド構文、コマンドモード、コマンド履歴、デフォルト値、使用上の注意、例	「*Authentication, Authorization, and Accounting Commands on* Cisco ASR 9000 シリーズ ルータ」

標準

標準	タイトル
この機能でサポートが追加または変更された標準はありません。また、この機能で変更された既存の標準のサポートはありません。	—

MIB

MIB	MIB リンク
—	Cisco IOS XR ソフトウェアを使用して MIB を検出およびダウンロードするには、URL（http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml）にある Cisco MIB Locator を使用して、[Cisco Access Products] メニューでプラットフォームを選択します。

RFC

RFC	タイトル
この機能でサポートが追加または変更された RFC はありません。また、この機能で変更された既存の RFC のサポートはありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカルサポート Web サイトには、数千ページに及ぶ検索可能な技術情報があります。製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクもあります。 Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	http://www.cisco.com/en/US/support/index.html

第 2 章

認証局相互運用性の実装

認証局（CA）相互運用性は、IP Security（IPSec）、Secure Socket Layer（SSL）および Secure Shell（SSH）プロトコルのサポートとして提供されます。 このモジュールでは、CA 相互運用性を実装する方法について説明します。

CA 相互運用性は、デバイスが CA からデジタル証明書を取得および使用できるように、Cisco ASR 9000 Series Router デバイスと CA の通信を許可します。 IPSec は CA を使用せずにネットワークで実装できますが、CA を使用すると、IPSec の管理性と拡張性が提供されます。

（注） このモジュールで使用される公開キーインフラストラクチャ（PKI）コマンドの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』の「*Public Key Infrastructure Commands on* Cisco ASR 9000 シリーズ ルータ」モジュールを参照してください。 このモジュールで言及する他のコマンドについては、コマンド リファレンス マスター インデックス（オンライン検索）を使用して、該当するマニュアルを参照してください。

認証局相互運用性の実装に関する機能履歴

リリース	変更点
リリース 3.7.2	この機能を追加しました。

• 参考資料，83 ページ

認証局の実装に関する前提条件

CA 相互運用性を実装するには、次の前提条件を満たす必要があります。

- 適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。
- セキュリティ ソフトウェアのパッケージインストレーション エンベロープ（PIE）をインストールしてアクティブにする必要があります。

  オプションの PIE インストールの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Management Configuration Guide*』を参照してください。
- この相互運用性機能を設定する前に、ネットワークでCA を使用可能にする必要があります。CA は、Cisco Systems PKI プロトコル、Simple Certificate Enrollment Protocol（SCEP）（以前の Certificate Enrollment Protocol（CEP））をサポートする必要があります。

認証局の実装に関する制約事項

Cisco IOS XR ソフトウェアは、2048 ビットを超える CA サーバ公開キーをサポートしません。

認証局の実装について

CA を実装するには、次の概念を理解する必要があります。

認証局相互運用性のサポートされている標準

シスコでは次の標準をサポートしています。

- IPsec：IP Security Protocol（IP セキュリティ プロトコル）。 IPSec は、データ保護、参加しているピア間のデータ整合性およびデータ認証を提供するオープン スタンダードです。 IPSec は、IP レイヤでこれらのセキュリティ サービスを提供し、インターネット キー交換（IKE）を使用して、ローカル ポリシーに基づいてプロトコルとアルゴリズムのネゴシエーションを処理し、IPSec で使用する暗号化および認証キーを生成します。 IPSec を使用することにより、ホスト ペア間、セキュリティ ゲートウェイ ペア間、またはセキュリティ ゲートウェイとホスト間の 1 つ以上のデータ フローを保護できます。
- IKE：Oakley キー交換や Skeme キー交換をインターネット セキュリティ アソシエーションおよびキー管理プロトコル（ISAKMP）フレームワーク内部に実装したハイブリッド プロトコルです。 IKE は他のプロトコルで使用できますが、その初期実装時は IPSec プロトコルで

使用します。 IKE は、IPSec ピアの認証を提供し、IPSec キーを交渉し、IPSec セキュリティ アソシエーション（SA）を交渉します。

- Public-Key Cryptography Standard #7（PKCS #7）：証明書登録メッセージの暗号化および署名に使用される RSA Data Security Inc. の標準。
- Public-Key Cryptography Standard #10（PKCS #10）：証明書要求のための RSA Data Security Inc. の標準構文。
- RSA キー：RSA は公開キー暗号化システムで、Ron Rivest、Adi Shamir、Leonard Adelman の 3 名によって開発されました。 RSA キーは、1 つの公開キーと 1 つの秘密キーのペアになっています。
- SSL：Secure Socket Layer プロトコル。
- X.509v3 証明書：同等のデジタル ID カードを各デバイスに提供することで、IPSec で保護されたネットワークの拡張を可能にする証明書サポート。 2 台の装置が通信する際、デジタル証明書を交換することで ID を証明します（これにより、各ピアで公開キーを手動で交換したり、各ピアで共有キーを手動で指定したりする必要がなくなります）。 これらの証明書は CA から取得されます。 X.509 は、ITU の X.500 標準の一部です。

認証局

次の項では、CA の背景情報を説明します。

CA の目的

CA は、証明書要求を管理し、参加する IPSec ネットワーク デバイスへの証明書の発行します。これらのサービスは、参加デバイスのキー管理を一元化して行います。

CA は、IPSec ネットワーク デバイスの管理を簡素化します。 CA は、ルータなど、複数の IPSec 対応デバイスを含むネットワークで使用できます。

Public Key Cryptography によりイネーブルにされたデジタル署名は、デバイスおよび個人ユーザをデジタル認証します。 RSA 暗号化システムなどの Public Key Cryptography では、各ユーザは、公開キーと秘密キーの両方を含むキーペアを使用します。 これらのキーは、補足として機能し、一方で暗号化されたものは、もう一方で復号化できます。 つまり、シグニチャは、データがユーザの秘密キーで暗号化されるときに形成されます。 受信側は、送信側の公開キーを使用してメッセージを復号化することで、シグニチャを検証します。 メッセージは、送信側の公開キーを使用して復号化できるため、秘密キーの所有者、つまり送信者がメッセージを作成することになります。 このプロセスは、受信者が送信者の公開キーのコピーを持っていて、これが本当に送信者のものであり、送信者を騙る他人のものではないことを高い確実性を持って知っていることを基盤としています。

デジタル証明書はリンクを提供します。 デジタル証明書には、名前、シリアル番号、企業、部署またはIPアドレスなど、ユーザまたはデバイスを特定する情報を含んでいます。また、エンティティの公開キーのコピーも含んでいます。 証明書自体は、受信者が身元を証明し、デジタル証明書を作成するうえで確実に信頼できるサード パーティである、CA により署名されます。

CA のシグニチャを検証するには、受信者は、CA の公開キーを認識している必要があります。 通常、このプロセスは、アウトオブバンドで、またはインストールで行われる操作により処理されます。 たとえば、通常の Web ブラウザでは、デフォルトで、複数の CA の公開キーが設定されています。 IKE は、IPSec の必須要素で、デジタル証明書を使用して、SA を設定する前にピア デバイスの拡張性を認証します。

デジタル シグニチャがない場合、ユーザは、IPSec を使用するデバイスの各ペア間で公開キーまたはシークレットを手動で交換して、通信を保護する必要があります。 証明書がない場合、ネットワークに新しいデバイスが追加されるたびに、安全に通信を行う他のすべてのデバイスで設定を変更する必要があります。 デジタル証明書がある場合、各デバイスは、CA に登録されます。 2 台のデバイスが通信する場合、証明書を交換し、データをデジタル署名して、お互いを認証します。 新しいデバイスがネットワークに追加されると、ユーザは、そのデバイスを CA に登録します。他のデバイスでは変更の必要はありません。 新しいデバイスが IPSec 接続を試行すると、証明書が自動的に交換され、デバイスを認証できます。

CA がない IPSec

CA を使用せずに、2 つの Cisco ルータ間で IPSec サービス（暗号化など）をイネーブルにする場合、最初に、各ルータにもう一方のルータのキー（RSA 公開キーや共有キー）が存在するか確認する必要があります。 つまり、次のいずれかの操作を手動で実行する必要があります。

- 各ルータで、もう一方のルータの RSA 公開キーを入力します。
- 各ルータで、両方のルータで使用される共有キーを指定します。

複数の Cisco ルータをメッシュ トポロジで配置し、すべてのルータ間で IPSec トラフィックを交換させる場合には、最初に、すべてのルータ間に共有キーまたは RSA 公開キーを設定する必要があります。

IPSec ネットワークに新しいルータを追加するごとに、新しいルータと既存の各ルータ間にキーを設定する必要があります

したがって、IPSec サービスを必要とするデバイスが増えるほど、キー管理は複雑になります。 このアプローチでは、より大型で複雑な暗号化ネットワークには拡張できません。

CA がある IPSec

CA を使用する場合、すべての暗号化ルータ間でキーを設定する必要はありません。 代わりに、加入させる各ルータを CA に個別に登録し、各ルータの証明書を要求します。 この登録が完了していれば、各加入ルータは、他のすべての加入ルータをダイナミックに認証できます。

ネットワークに新しい IPSec ルータを追加する場合、新しいルータが CA に証明書を要求するように設定するだけでよく、既存の他のすべての IPSec ルータとの間に複数のキー設定を行う必要はありません。

複数のトラストポイント CA がある IPSec

複数のトラストポイント CA がある場合、証明書をピアに発行した CA にルータを登録する必要はありません。 その代わり、信頼できる複数の CA にルータを設定します。 そのため、ルータは、設定された CA（信頼できるルート）を使用して、ルータ ID で定義されている同じ CA により発行されていない証明を、ピアが提供したかどうかを検証できます。

複数の CA を設定することにより、IKE を使用して IPSec トンネルを確立する場合に、異なるドメイン（異なる CA）に登録した 2 台以上のルータ間で相互の ID を確認できます。

SCEP では、各ルータは、CA（登録 CA）で設定されます。 CA は、CA の秘密キーで署名されるルータに証明書を発行します。同じドメインのピアの証明書を確認するため、ルータは、登録CA のルート証明書でも設定されます。

異なるドメインからピアの証明書を確認するには、そのピアのドメインの登録 CA のルート証明書をルータで安全に設定する必要があります。

IKE フェーズ I の署名の検証中、発信側は CA 証明書のリストを応答側に送信します。応答側は、リストのいずれかの CA により発行される証明書を送信する必要があります。 証明書が検証されたら、証明書に含まれる公開キーを公開キー リングに保存します。

複数のルート CA がある場合、バーチャル プライベート ネットワーク（VPN）ユーザは、一方のドメインで信頼を確立して、もう一方のドメインで簡単かつ安全に配布できます。 そのため、異なるドメインで認証されるエンティティ間の必要なプライベート通信チャネルが発生します。

IPSec デバイスにより CA 証明書の使用方法

2 台の IPSec ルータが IPSec で保護されたトラフィックを交換するには、最初に相互に認証しあう必要があります。認証されていない場合、IPSec 保護が適用されません。 この認証を行うには、IKE を使用します。

CA を使用しない場合、ルータは、RSA 暗号化ナンスまたは事前共有キーを使用してリモート ルータに自身を認証します。 いずれの方式でも、2 つのルータ間でキーを事前に設定しておく必要があります。

CA を使用する場合、ルータはリモート ルータに証明書を送信し、何らかの公開キー暗号法を実行することによって、ルータ スイッチに対して自身を認証します。 各ルータは、CA により発行されて検証された、ルータ固有の証明書を送信する必要があります。 このプロセスが有効なのは、各ルータの証明書にルータの公開キーがカプセル化され、各証明書が CA によって認証されることにより、すべての加入ルータが CA を認証局として認識するからです。 この機構は、RSA シグニチャを使用する IKE と呼ばれます。

ルータは、証明書が期限切れになるまで、複数の IPSec ピアに対して、複数の IPSec セッション用に自身の証明書を継続的に送信できます。 証明書が期限満了になったときは、ルータの管理者は新しい証明書を CA から入手する必要があります。

ルータが別のドメイン（異なる CA）のピアから証明書を受信した場合、ルータの CA からダウンロードした証明書失効リスト（CRL）には、そのピアの証明書情報は含まれません。 そのため、Lightweight Directory Access Protocol（LDAP）URL で設定したトラストポイントで発行された CRL をチェックして、ピアの証明書が失効しているかどうかを確認します。

LDAP URL で設定されているトラストポイントにより発行された CRL を照会するには、トラストポイント コンフィギュレーション モードで **query url** コマンドを使用します。

CA 登録局

CA によっては、実装の一部として登録局（RA）を使用します。 RA は CA のプロキシの役割を果たすサーバであるため、CA が使用できないときも CA 機能は継続しています。

CA 相互運用性の実装方法

この項では、次の手順について説明します。

ルータのホスト名および IP ドメイン名の設定

この作業では、ルータのホスト名および IP ドメイン名を設定します。

ルータのホスト名および IP ドメイン名が未設定の場合には、これらを設定する必要があります。ホスト名および IP ドメイン名が必要なのは、ルータが完全修飾ドメイン名（FQDN）を IPSec により使用されるキーおよび証明書に割り当て、ルータに割り当てられたホスト名および IP ドメイン名に FQDN が基づいているためです。 たとえば、router20.example.com という名前の証明書は、router20 というルータのホスト名と example.com というルータの IP ドメイン名に基づいています。

手順の概要

1. **configure**
2. **hostname** *name*
3. **domain name domain-name**
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**hostname** *name* 例： `RP/0/RSP0/CPU0:router(config)# hostname myhost`	ルータのホスト名を設定します。
ステップ 3	**domain name domain-name** 例： `RP/0/RSP0/CPU0:router(config)# domain name mydomain.com`	ルータの IP ドメイン名を設定します。

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータがEXECモードに戻ります。
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。
		
		• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

RSA キー ペアの生成

RSA キー ペアを生成します。

RSA キー ペアはIKE キー交換管理メッセージの署名および暗号化に使用されます。また、ルータの証明書を取得する際に必要です。

手順の概要

1. **crypto key generate rsa** [**usage keys** | **general-keys**] [**keypair-label**]
2. **crypto key zeroize rsa** [**keypair-label**]
3. show crypto key mypubkey rsa

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**crypto key generate rsa** [**usage keys** | **general-keys**] [**keypair-label**] 例： `RP/0/RSP0/CPU0:router# crypto key generate rsa general-keys`	RSA キー ペアを生成します。 • **usage keys** キーワードを使用して、特殊用途キーを指定します。**general-keys** キーワードを使用して、汎用 RSA キーを指定します。 • *keypair-label* 引数は、RSA キー ペアを指定する RSA キー ペア ラベルです。
ステップ 2	**crypto key zeroize rsa** [**keypair-label**] 例： `RP/0/RSP0/CPU0:router# crypto key zeroize rsa key1`	（任意）ルータからすべての RSA を削除します。 • 場合によっては、すべての RSA キーをルータから削除します。 たとえば、何らかの原因で RSA キーペアの信用性が失われ、使用しなくなった場合、そのキーペアを削除します。 • 特定の RSA キー ペアを削除するには、*keypair-label* 引数を使用します。
ステップ 3	show crypto key mypubkey rsa 例： `RP/0/RSP0/CPU0:router# show crypto key mypubkey rsa`	（任意）ルータの RSA 公開キーを表示します。

公開キーのルータへのインポート

公開キーをルータにインポートします。

公開キーがルータにインポートされ、ユーザが認証されます。

手順の概要

1. **crypto key import authentication rsa** [**usage keys** | **general-keys**] [**keypair-label**]
2. show crypto key mypubkey rsa

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**crypto key import authentication rsa** [**usage keys** | **general-keys**] [**keypair-label**] 例： `RP/0/RSP0/CPU0:router# crypto key import authentication rsa general-keys`	RSA キー ペアを生成します。 • **usage keys** キーワードを使用して、特殊用途キーを指定します。**general-keys** キーワードを使用して、汎用 RSA キーを指定します。 • *keypair-label* 引数は、RSA キー ペアを指定する RSA キー ペア ラベルです。
ステップ 2	show crypto key mypubkey rsa 例： `RP/0/RSP0/CPU0:router# show crypto key mypubkey rsa`	（任意）ルータの RSA 公開キーを表示します。

認証局の宣言および信頼できるポイントの設定

CA を宣言し、信頼できるポイントを設定します。

手順の概要

1. **configure**
2. **crypto ca trustpoint ca-name**
3. **enrollment url CA-URL**
4. **query url LDAP-URL**
5. **enrollment retry period minutes**
6. **enrollment retry count number**
7. **rsakeypair keypair-label**
8. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**crypto ca trustpoint ca-name** 例： `RP/0/RSP0/CPU0:router(config)# crypto ca trustpoint myca`	CA を宣言します。 • ルータがピアに対して発行された証明書を確認できるように、選択した名前で信頼できるポイントを設定します。 • トラストポイント コンフィギュレーション モードを開始します。
ステップ 3	**enrollment url CA-URL** 例： `RP/0/RSP0/CPU0:router(config-trustp)# enrollment url http://ca.domain.com/certsrv/mscep/mscep.dll`	CA の URL を指定します。 • URL には、非標準 cgi-bin スクリプトの場所が含まれている必要があります。
ステップ 4	**query url LDAP-URL** 例： `RP/0/RSP0/CPU0:router(config-trustp)# query url ldap://my-ldap.domain.com`	（任意）CA システムにより LDAP プロトコルがサポートされている場合、LDAP サーバの位置を指定します。
ステップ 5	**enrollment retry period minutes** 例： `RP/0/RSP0/CPU0:router(config-trustp)# enrollment retry period 2`	（任意）再試行期間を指定します。 • 証明書の要求後、ルータは CA からの証明書の受け取りを待機します。ルータが期間（再試行期間）内に証明書を受け取らない場合、ルータは、別の証明書要求を送信します。 • 範囲は 1 ～ 60 分です。 デフォルトは 1 分です。
ステップ 6	**enrollment retry count number** 例： `RP/0/RSP0/CPU0:router(config-trustp)# enrollment retry count 10`	（任意）失敗した証明書要求送信を続行する回数を指定します。 • 範囲は 1 ～ 100 です。

	コマンドまたはアクション	目的
ステップ 7	**rsakeypair keypair-label** 例： `RP/0/RSP0/CPU0:router(config-trustp)# rsakeypair mykey`	（任意）このトラストポイントに **crypto key generate rsa** コマンドを使用して生成した指定 RSA キー ペアを指定します。 • このキーペアを設定しない場合、トラストポイントは現在の設定のデフォルトの RSA キーを使用します。

ステップ 8	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

CA の認証

CA をルータに対して認証します。

ルータは、CA の公開キーを含む CA の自己署名証明書を取得して CA を認証する必要があります。 この CA の証明書は自己署名（CA が自身の証明書に署名したもの）であるため、CA の公開キーは、CA アドミニストレータに連絡し、CA 証明書のフィンガープリントを比較して手動で認証する必要があります。

手順の概要

1. **crypto ca authenticate ca-name**
2. show crypto ca certificates

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**crypto ca authenticate ca-name** 例： `RP/0/RSP0/CPU0:router# crypto ca authenticate myca`	CA の公開キーを含む CA 証明書を取得することで、ルータに対して CA を認証します。
ステップ 2	show crypto ca certificates 例： `RP/0/RSP0/CPU0:router# show crypto ca certificates`	（任意）CA証明書に関する情報を表示します。

独自の証明書の要求

証明書を CA から要求します。

ルータの各 RSA キー ペアに対して、署名された証明書を CA から取得する必要があります。 汎用 RSA キーを生成した場合、ルータの RSA キー ペアは 1 つだけなので、必要な証明書は 1 つだけです。 特殊用途 RSA キーを生成した場合、ルータには 2 つの RSA キー ペアがあるので、必要な証明書は 2 つです。

手順の概要

1. **crypto ca enroll ca-name**
2. show crypto ca certificates

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**crypto ca enroll ca-name** 例： `RP/0/RSP0/CPU0:router# crypto ca enroll myca`	すべての RSA キー ペアの証明書を要求します。 • このコマンドでは、ルータは存在する RSA キー ペアと同数の証明書を要求するため、特定目的の RSA キー ペアがある場合にも、このコマンドは 1 回しか実行する必要はありません。

	コマンドまたはアクション	目的
		• このコマンドでは、設定に保存されないチャレンジパスワードを作成する必要があります。証明書を失効させる必要が生じた場合、このパスワードが要求されるので、このパスワードを覚えておく必要があります。 • 証明書はすぐに発行できます。または、登録リトライ時間に達し、タイムアウトが発生するまで、ルータが証明書要求を毎分送信します。タイムアウトが発生した場合、システム管理者に要求承認を依頼して、このコマンドを再入力します。
ステップ 2	show crypto ca certificates 例： `RP/0/RSP0/CPU0:router# show crypto ca certificates`	（任意）CA 証明書に関する情報を表示します。

カットアンドペーストによる証明書登録の設定

ルータが使用するトラストポイント認証局（CA）を宣言して、このトラストポイント CA をカットアンドペーストによる手動登録に設定します。

手順の概要

1. **configure**
2. **crypto ca trustpoint** *ca-name*
3. enrollment terminal
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
5. **crypto ca authenticate** *ca-name*
6. **crypto ca enroll** *ca-name*
7. **crypto ca import** ca- *name* **certificate**
8. show crypto ca certificates

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ 2	**crypto ca trustpoint** *ca-name*
	
	例：
	`RP/0/RSP0/CPU0:router(config)# crypto ca trustpoint myca`
	`RP/0/RSP0/CPU0:router(config-trustp)#`
		ルータが使用するCAを宣言し、CAトラストポイントコンフィギュレーション モードを開始します。
		• *ca-name* 引数を使用して、CA の名前を指定します。

ステップ 3	enrollment terminal 例： `RP/0/RSP0/CPU0:router(config-trustp)# enrollment terminal`	カットアンドペーストによる手動での証明書登録を指定します。

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 5	**crypto ca authenticate** *ca-name* 例： `RP/0/RSP0/CPU0:router# crypto ca authenticate myca`	CA の証明書を取得することにより、CA を認証します。 • *ca-name* 引数を使用して、CA の名前を指定します。ステップ 2, （80 ページ） で入力した名前と同じ名前を使用します。
ステップ 6	**crypto ca enroll** *ca-name* 例： `RP/0/RSP0/CPU0:router# crypto ca enroll myca`	CA からルータの証明書を取得します。 • *ca-name* 引数を使用して、CA の名前を指定します。ステップ 2 で入力した名前と同じ名前を使用します。
ステップ 7	**crypto ca import** ca- *name* **certificate** 例： `RP/0/RSP0/CPU0:router# crypto ca import myca certificate`	端末で証明書を手動でインポートします。 • *ca-name* 引数を使用して、CA の名前を指定します。ステップ 2 で入力した名前と同じ名前を使用します。 （注） 用途キー（シグニチャおよび暗号キー）を使用する場合は、**crypto ca import** コマンドを 2 回入力する必要があります。このコマンドを最初に入力した場合は、認証の 1 つがルータにペーストされます。2 回目に入力した場合は、他の認証がルータにペーストされます（どの認証が最初にペーストされるかは重要ではありません）。
ステップ 8	show crypto ca certificates 例： `RP/0/RSP0/CPU0:router# show crypto ca certificates`	証明書と CA 証明書に関する情報を表示します。

認証局相互運用性の実装の設定例

この項では、次の設定例について説明します。

認証局相互運用性の設定：例

次に、CA 相互運用性を設定する例を示します。

さまざまなコマンドを説明するコメントが設定に含まれます。

```
configure
hostname myrouter
domain name mydomain.com
end
```

```
Uncommitted changes found, commit them? [yes]:yes

crypto key generate rsa mykey

The name for the keys will be:mykey
Choose the size of the key modulus in the range of 360 to 2048 for your General Purpose
Keypair
Choosing a key modulus greater than 512 may take a few minutes.
How many bits in the modulus [1024]:
Generating RSA keys ...
Done w/ crypto generate keypair
[OK]

show crypto key mypubkey rsa

Key label:mykey
Type     :RSA General purpose
Size     :1024
Created  :17:33:23 UTC Thu Sep 18 2003
Data     :
 30819F30 0D06092A 864886F7 0D010101 05000381 8D003081 89028181 00CB8D86
 BF6707AA FD7E4F08 A1F70080 B9E6016B 8128004C B477817B BCF35106 BC60B06E
 07A417FD 7979D262 B35465A6 1D3B70D1 36ACAFBD 7F91D5A0 CFB0EE91 B9D52C69
 7CAF89ED F66A6A58 89EEF776 A03916CB 3663FB17 B7DBEBF8 1C54AF7F 293F3004
 C15B08A8 C6965F1E 289DD724 BD40AF59 E90E44D5 7D590000 5C4BEA9D B5020301
 0001

! The following commands declare a CA and configure a trusted point.

configure
crypto ca trustpoint myca
enrollment url http://xyz-ultra5
enrollment retry count 25
enrollment retry period 2
rsakeypair mykey
end

Uncommitted changes found, commit them? [yes]:yes

! The following command authenticates the CA to your router.

crypto ca authenticate myca

Serial Number  :01
Subject Name   :
cn=Root coax-u10 Certificate Manager,ou=HFR,o=Cisco Systems,l=San Jose,st=CA,c=US
Issued By      :
cn=Root coax-u10 Certificate Manager,ou=HFR,o=Cisco Systems,l=San Jose,st=CA,c=US
Validity Start :07:00:00 UTC Tue Aug 19 2003
Validity End   :07:00:00 UTC Wed Aug 19 2020
Fingerprint:58 71 FB 94 55 65 D4 64 38 91 2B 00 61 E9 F8 05
Do you accept this certificate?? [yes/no]:yes

! The following command requests certificates for all of your RSA key pairs.

crypto ca enroll myca

% Start certificate enrollment ...
% Create a challenge password. You will need to verbally provide this
  password to the CA Administrator in order to revoke your certificate.
% For security reasons your password will not be saved in the configuration.
% Please make a note of it.

Password:
Re-enter Password:
    Fingerprint: 17D8B38D ED2BDF2E DF8ADBF7 A7DBE35A

! The following command displays information about your certificate and the CA certificate.

show crypto ca certificates

Trustpoint       :myca
```

```
============================================================
CA certificate
  Serial Number  :01
  Subject Name   :
        cn=Root coax-u10 Certificate Manager,ou=HFR,o=Cisco Systems,l=San Jose,st=CA,c=US
  Issued By      :
        cn=Root coax-u10 Certificate Manager,ou=HFR,o=Cisco Systems,l=San Jose,st=CA,c=US
  Validity Start :07:00:00 UTC Tue Aug 19 2003
  Validity End   :07:00:00 UTC Wed Aug 19 2020
Router certificate
  Key usage      :General Purpose
  Status         :Available
  Serial Number  :6E
  Subject Name   :
        unstructuredName=myrouter.mydomain.com,o=Cisco Systems
  Issued By      :
        cn=Root coax-u10 Certificate Manager,ou=HFR,o=Cisco Systems,l=San Jose,st=CA,c=US
  Validity Start :21:43:14 UTC Mon Sep 22 2003
  Validity End   :21:43:14 UTC Mon Sep 29 2003
  CRL Distribution Point
        ldap://coax-u10.cisco.com/CN=Root coax-u10 Certificate Manager,O=Cisco Systems
```

次の作業

CA 相互運用性の設定が終了したら、IKE、IPSec および SSL を設定する必要があります。 IKE 設定については、「*Implementing Internet Key Exchange Security Protocol on* Cisco ASR 9000 シリーズ ルータ」モジュール、「IPSec in the *Implementing IPSec Network Security on* Cisco ASR 9000 シリーズ ルータ」モジュールおよび「SSL in the *Implementing Secure Socket Layer on* Cisco ASR 9000 シリーズルータ」モジュールを参照してください。 これらのモジュールは、『*Cisco ASR 9000 Series Aggregation Services Router System Security Configuration Guide*』（この資料）にあります。

参考資料

次の項では、認証局相互運用性の実装に関連する参考資料を提供します。

関連資料

関連項目	ドキュメント名
PKI コマンド：コマンド構文の詳細、コマンド モード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』の「*Public Key Infrastructure Commands on* Cisco ASR 9000 シリーズ ルータ」モジュール

標準

標準	タイトル
この機能でサポートが追加または変更された標準はありません。また、この機能で変更された既存の標準のサポートはありません。	—

MIB

MIB	MIB リンク
—	Cisco IOS XR ソフトウェアを使用して MIB を検出およびダウンロードするには、URL（http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml）にある Cisco MIB Locator を使用して、[Cisco Access Products] メニューでプラットフォームを選択します。

RFC

RFC	タイトル
この機能でサポートが追加または変更された RFC はありません。また、この機能で変更された既存の RFC のサポートはありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトには、数千ページに及ぶ検索可能な技術情報があります。製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクもあります。 Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	http://www.cisco.com/en/US/support/index.html

第 3 章

インターネット キー交換セキュリティ プロトコルの実装

インターネット キー交換（IKE）は、IP Security（IPSec）標準と組み合わせて使用されるキー管理プロトコル標準です。 IPsec は、IP パケットに対して強力な認証や暗号化を実現する機能です。

IKE は、Oakley キー交換や Skeme キー交換をインターネット セキュリティ アソシエーションおよびキー管理プロトコル（ISAKMP）フレームワーク内部に実装したハイブリッドプロトコルです （ISAKMP、Oakley、および Skeme は、IKE により実装されるセキュリティ プロトコルです）。

IPSec の設定には必ずしも IKE は必要ありませんが、IKE では、IPSec 標準に対する新機能が追加されているほか、設定をより柔軟かつ容易に行えるよう、IPSec のサポートが強化されています。

このモジュールでは、Cisco ASR 9000 シリーズ ルータ で IKE を実装する手順について説明します。

（注） このモジュールで使用される IKE コマンドの詳細については、の「*Internet Key Exchange Security Protocol Commands on* Cisco ASR 9000 シリーズ ルータ」モジュールを参照してください。 このモジュールで使用される他のコマンドの説明については、コマンド リファレンスのマスター インデックスを参照するか、またはオンラインで検索してください。

インターネット キー交換セキュリティ プロトコルの実装に関する機能履歴

リリース	変更点
リリース 3.7.2	この機能を追加しました。

- インターネット キー交換の実装に関する前提条件，86 ページ

インターネット キー交換の実装に関する前提条件

インターネット キー交換を実装するには、次の前提条件を満たす必要があります。

- 適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。 ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。
- セキュリティ ソフトウェアのパッケージインストレーション エンベロープ（PIE）をインストールしてアクティブにする必要があります。

  オプションの PIE インストールの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Management Configuration Guide*』を参照してください。

IPSec ネットワークでの IKE セキュリティ プロトコル設定の実装について

IKE を実装するには、次の概念について理解しておく必要があります。

サポートされている標準

シスコでは次の標準を採用しています。

- **IKE**：**インターネット キー交換。** ハイブリッド プロトコルで、Oakley キー交換と SKEME キー交換を ISAKMP フレームワーク内部に実装しています。 IKE は他のプロトコルで使用できますが、その初期実装時は IPSec プロトコルで使用します。 IKE は、IPSec ピアの認証を提供し、IPSec キーを交渉し、IPSec セキュリティ アソシエーション（SA）を交渉します。

  IKE は、RFC 2409『*The Internet Key Exchange*』に従い実装されます。
- **IPSec**：**IP Security プロトコル。** IPSec は、データ保護、参加しているピア間のデータ整合性およびデータ認証を提供するオープン スタンダードです。 IPSec は、これらのセキュリ

ティ サービスを IP レイヤで提供します。IPSec は、IKE を使用して、ローカル ポリシーに基づいてプロトコルのネゴシエーションおよびアルゴリズムを処理し、IPSec で使用される暗号キーと認証キーを生成します。 IPSec では、一対のホスト間、一対のセキュリティ ゲートウェイ間、または一対のセキュリティ ゲートウェイとホストの間で1つ以上のデータ フローを保護できます。

IPSec の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Security Configuration Guide*』の「IPSec Network Security」モジュールを参照してください。

- **ISAKMP**：インターネット セキュリティ アソシエーションおよびキー管理プロトコル。 ペイロード形式、キー交換プロトコル実装の方法、およびセキュリティ アソシエーションのネゴシエーションを定義するプロトコル フレームワークです。

  ISAKMP は、最新バージョンの『*Internet Security Association and Key Management Protocol (ISAKMP)*』Internet Draft（RFC 2408）に従い実装されます。

- **Oakley**：キー交換プロトコルの 1 つで、認証済みのキー関連情報を取得する方法を定義します。

- **Skeme**：キー交換プロトコルの 1 つで、キーをすばやく更新しながら認証済みのキー関連情報を取得する方法を定義します。

IKE での使用に備えて実装されているコンポーネント テクノロジーには次のものがあります。

- **DES**：データ暗号規格。 パケット データの暗号化に使用されるアルゴリズムです。 IKE は Explicit IV 標準の 56 ビット DES-CBC を実装しています。 Cipher Block Chaining（CBC）では、暗号化の開始に初期ベクター（IV）が必要です。 IV は IPSec パケットに明示的に指定されます。

  また Cisco IOS XR ソフトウェアは、特定のプラットフォームで使用可能なソフトウェア バージョンに応じて、トリプル DES（168 ビット）暗号化も実装します。 トリプル DES（3DES）は強力な暗号化方式であり、これにより、機密性の高い情報を非信頼ネットワーク上で送信できます。 この暗号化方式を使用することで、（特に金融業界の）お客様はネットワーク層での暗号化を実現できます。

- **AES**：高度暗号化規格。 128 ビット、192 ビットおよび 256 ビットの規格がサポートされています。

（注） 強力な暗号化（56 ビット データ暗号化機能セットを含むがこれに限らない）Cisco IOS XR イメージは、米国 政府により輸出が規制されるため、配布が制限されます。 米国以外の国でインストールされるイメージには、輸出許可が必要です。 お客様のご注文は、米国政府の規制により拒否される、または遅延することがあります。 詳細については、営業担当者または販売業者、あるいは export@cisco.com までお問い合わせください。

- **Diffie-Hellman**：公開キー暗号法プロトコルの 1 つで、2 者間に、セキュアでない通信チャネルによる共有秘密を確立できます。 Diffie-Hellman は、IKE 内でセッション キーを確立する

ために使用されます。 768 ビット、1024 ビット、1536 ビットの各 Diffie-Hellman グループがサポートされています。

- **MD5（HMAC バリアント）**：メッセージ ダイジェスト 5。 パケット データの認証に使用するハッシュ アルゴリズム。 HMAC は、追加レベルのハッシュを提供するバリアントです。
- **SHA（HMAC バリアント）**：セキュア ハッシュ アルゴリズム。 パケット データの認証に使用するハッシュ アルゴリズム。 HMAC は、追加レベルのハッシュを提供するバリアントです。
- **RSA シグニチャおよび RSA 暗号化ナンス**：RSA は、ロナルド・リベスト、アディ・シャミア、レオナルド・エーデルマンの 3 人によって開発された公開キー暗号化システムです。 RSA シグニチャは否認防止を実行し、RSA 暗号化ナンスは否認を実行します （否認および否認防止は、トレーサビリティと関連付けられます）。

IKE は、X.509v3 証明書標準と相互運用します。 これは、認証で公開キーが必要な場合に IKE プロトコルにより使用されます。 この証明書サポートを使用すると、各デバイスに同等のデジタル ID カードを付与することで、保護されたネットワークを拡張できます。 2 つのデバイスが通信する際、デジタル証明書を交換することで ID を証明します。これにより、各ピアで公開キーを手動で交換したり、各ピアで共有キーを手動で指定したりする必要がなくなります。

IKE をイネーブルにしない場合の譲歩

IKE は、デフォルトでは、Cisco IOS XR ソフトウェアでディセーブルです。 IKE をイネーブルにしない場合、ピアでこれらの接続を確立する必要があります。

- すべてのピアの暗号プロファイルですべての IPSec セキュリティ アソシエーションを手動で指定する必要があります （暗号プロファイル設定については、『*Cisco ASR 9000 Series Aggregation Services Router System Security Configuration Guide*』のモジュール「*Implementing IPSec Network Security on* Cisco ASR 9000 シリーズ ルータ」を参照してください）。
- ピアの IPSec セキュリティ アソシエーションは、指定の IPSec セッション中はタイムアウトになりません。
- ピア間の IPSec セッション中、暗号キーは変更されません。
- アンチリプレイ サービスはピア間で使用できません。
- 認証局（CA）サポートは使用できません。

IKE ポリシー

各ピアに IKE ポリシーを作成する必要があります。 IKE ポリシーを使い、IKE ネゴシエーション中に使用するセキュリティ パラメータの組み合わせを定義します。

IKE ポリシーを作成および設定する前に、次の概念について理解しておく必要があります。

IKE ポリシーの作成

IKE ネゴシエーションは保護する必要があるため、各 IKE ネゴシエーションは、共有（共通）の IKE ポリシーについて両ピアが同意することで開始されます。 このポリシーには、次の IKE ネゴシエーションを保護するために使用するセキュリティ パラメータとピアの認証方法を記述します。

両ピアがポリシーに同意すると、各ピアに確立されているセキュリティ アソシエーションによってポリシーのセキュリティ パラメータが識別され、ネゴシエーションにおける以降すべての IKE トラフィックに適用されます。

各ピアにおいて、複数のポリシーを優先順位付きで作成して、少なくとも 1 つのポリシーがリモート ピアのポリシーに一致するようにできます。

ポリシー パラメータの定義

次の表に、各 IKE ポリシーで定義する 5 つのパラメータを示します。

表 *4*：*IKE* ポリシー パラメータの定義

パラメータ	許容値	キーワード	デフォルト値
encryption algorithm	56 ビット DES-CBC 168 ビット DES 128 ビット AES 192 ビット AES 256 ビット AES	des 3des aes aes 192 aes 256	56 ビット DES-CBC
Hash algorithm	SHA-1（HMAC バリアント） MD5（HMACバリアント）	sha md5	SHA-1
Authentication method	RSA シグニチャ RSA 暗号化ナンス 事前共有キー	rsa-sig rsa-encr pre-share	RSA シグニチャ
Diffie-Hellman group identifier	768 ビット Diffie-Hellman または 1024 ビット Diffie-Hellman 1536 ビット Diffie-Hellman	1 2 5	768 ビット Diffie-Hellman

パラメータ	許容値	キーワード	デフォルト値
Lifetime of the security association このライフタイムおよび使用方法については、**lifetime** コマンドのコマンド説明を参照してください。	任意の秒数	—	86400 秒（1 日）

これらのパラメータは、IKE セキュリティ アソシエーションが確立されるときに IKE ネゴシエーションに適用されます。

ポリシー一致の IKE ピア同意

IKE ネゴシエーションが開始されると、IKE は、両ピア上で同一の IKE ポリシーを検索します。ネゴシエーションを開始したピアがすべてのポリシーをリモート ピアに送信し、リモート ピアの方では一致するポリシーを探そうとします。 リモート ピアは、独自の優先順位が最も高いポリシー（最も小さい優先順位番号）と他のピアから受信したポリシーを比較することで、一致するポリシーを探します。一致するポリシーが見つかるまで、リモート ピアは優先順位が高い順に各ポリシーをチェックします。

2 つのピアのポリシーが一致するのは、2 つのピアが同じ暗号化、ハッシュ、認証、Diffie-Hellman パラメータの各値を持ち、リモート ピアのポリシーに指定されているライフタイムが、比較しているポリシーのライフタイム以下の場合です（ライフタイムが同一でない場合は、リモート ピアのポリシーのライフタイムよりも短いライフタイムが使用されます）。

一致するポリシーが見つからなかった場合は、IKE はネゴシエーションを拒否し、IPSec は確立されません （関連情報については、特定のポリシー セットへの IKE ピアの制限, （91 ページ）の項を参照してください）。

一致するポリシーが見つかった場合、IKE は、ネゴシエーションを完了し、ISAKMP セキュリティ アソシエーション（SA）が作成されます。ISAKMP SA 事前共有キーまたは証明書を確立するには、一致するポリシーを設定する必要があります。 一致するポリシーがない場合、ISAKMP SA を確立できません。

（注） ポリシーに指定する認証方式によっては、追加の設定が必要な場合があります（IKE ポリシーに必要な追加設定, （93 ページ）の項を参照）。 ピアのポリシーに必要な比較設定がされていないと、一致するポリシーをリモート ピアで検索するときに、ピアはポリシーを送信しません。

特定のポリシー セットへの IKE ピアの制限

Cisco VPN クライアントは、使用可能なすべてのポリシーで事前設定されていて、ハブに接続するときにこれらのすべてのポリシーを提案します。 次に、ハブは、「最初に見つかった」ポリシーを選択する必要があります。 ただし、ユーザによっては、IPSec ゲートウェイを介して接続するときにローカル ピアとリモート ピア間で強力な暗号化アルゴリズムの使用を制限する必要があります。 Cisco VPN クライアントでは、ユーザは、使用するポリシー（および暗号化アルゴリズム）を選択できませんが、このような制限を実装するポリシー セットを代わりに使用できます。 ピアとポリシー セット間の一致は、ポリシー セットで識別されるローカル IP アドレス（または SVI で設定されるトンネル ソース）との一致に基づいて、制限または許可されます。

たとえば、IPSec ハブは 6 つのポリシーが設定されていて、ポリシー セットがこれらの 6 つのうち 3 つだけ設定されているとします。 この場合、リモート クライアントがトンネルを開始しようとし、この SVI トンネル ソース アドレスを参照するときに、ポリシー セットから一致するポリシーがあるか確認されます。 IKE は、ポリシー セットで示された 3 つのポリシーで、優先順位の高いものから順に（小さい数字ほど、優先順位は高くなります）、一致するポリシーを探します。これらの 3 つのポリシーから一致するポリシーが見つからなかった場合、トンネルは確立できません。

ローカル IP アドレスが特定の IKE ポリシーに制限されていない SVI に、リモート ピアが接続しようとすると、ポリシー一致の IKE ピア同意, （90 ページ）で説明されているデフォルトの動作が実行されます。

単一のポリシー セットには最大 5 つの ISAKMP ポリシーを設定できます。

IKE ピアを特定のポリシーセットに制限する方法については、このモジュールの「特定のポリシー セットへの IKE ピアの制限」を参照してください。

パラメータ値の選択

IKE 標準に従い、各パラメータの特定の値を選択できます。 ここでは、値を選択する基準について説明します。

サポートされているパラメータの値が 1 つしかないデバイスを使用する場合は、もう一方のデバイスでサポートされている値を設定する必要があります。 この制限を別にすれば、セキュリティとパフォーマンスには通常トレードオフの関係があり、パラメータ値の多くにはこのトレードオフがあります。 ネットワークのセキュリティ リスクのレベルと、そのリスクに対する許容度を評価する必要があります。 次のヒントは、各パラメータに指定する値を選択するときの参考にしてください。

- 暗号化アルゴリズムには、56 ビット DES-CBC、168 ビット DES、128 ビット AES、192 ビット AES および 256 ビット AES の 5 つのオプションがあります。
- ハッシュ アルゴリズムには、SHA-1 と MD5 の 2 つのオプションがあります。

  MD5 のダイジェストの方が小さく、SHA-1 よりもやや速いと見なされています。 証明されている成功した（極端に難しい）攻撃は MD5 におけるものですが、IKE により使用される HMAC バリアントはこの攻撃を防ぎます。

- 認証方式には、RSA シグニチャ、RSA 暗号化ナンスおよび事前共有キーの 3 つのオプションがあります。
  - RSA シグニチャにより、IKE ネゴシエーションで否認防止が可能になります（さらに、リモート ピアとのIKE ネゴシエーションを実際に実行することで、第三者に対する証明が可能になります）。

  RSA シグニチャでは、CA を使用できます。 CA を使用すると、IPSec ネットワークの管理性と拡張性が劇的に改善されます。 また、RAS シグニチャ ベースの認証で使用できる公開キー操作は 2 つだけです。これに対し、RSA 暗号化では 4 つの公開キー操作を使用しますが、その分だけ全体のパフォーマンスが下がります。
  - RSA 暗号化ナンスでは IKE ネゴシエーションを否認できます。ただし、RSA シグニチャとは異なり、リモート ピアと IKE ネゴシエーションを実行したことを第三者に対して証明はできません。

  RSA 暗号化ナンスでは、ピアが認証局を使用せずにお互いの公開キーを処理する必要があります。 その代わり、ピアは、互いの公開キーを次の 2 つの方法で取得できます。
  - ローカル ピアが、リモート ピアとの成功した IKE ネゴシエーション中に、RSA シグニチャと証明書を使用していた場合、ローカル ピアは、リモート ピアの公開キーをすでに処理しています（証明書を使用すると、RSA シグニチャ ベースの IKE ネゴシエーション中にピアの公開キーが交換されます）。
  - 事前共有キーは、大規模なセキュア ネットワークでは、成長するネットワークにうまく対応できないため、適していません。 ただし、RSA シグニチャのように認証局を使用する必要がないため、10 ノード未満の規模の小さいネットワークでは設定が簡単です。また、事前共有キーによる認証に比べ、RSA シグニチャによる認証の方が安全です。

- Diffie-Hellman グループ ID には、768 ビット、1024 ビット Diffie-Hellman および 1536 ビット Diffie Hellman の 3 つのオプションがあります。

  1024 ビットおよび 1536 ビットの Diffie-Hellman オプションを使用すると、「解読」がより困難になる一方、実行に必要な CPU 時間が増えます。

- セキュリティ アソシエーションのライフタイムは、任意の値に設定できます。

  一般的に、一定の限度に達するまで、ライフタイムが短いほど、IKE ネゴシエーションがセキュアになります。 ただし、ライフタイムを長くすれば、以後の IPSec セキュリティ アソシエーションをそれだけ速くセットアップできます。 このパラメータおよび使用方法の詳細については、**lifetime** コマンドのコマンド説明を参照してください。

ポリシーの作成

それぞれが異なるパラメータ値の組み合わせを持つ、複数の IKE ポリシーを作成できます。 作成する各ポリシーに対して、一意の優先順位を割り当てます（1 ～ 10,000 で指定し、1 が最大の優先順位）。

各ピアで複数のポリシーを設定できますが、これらのうち少なくとも 1 つのポリシーに、リモート ピアのいずれかと同じ暗号化、ハッシュ、認証および Diffie-Hellman パラメータ値が含まれている必要があります（ライフタイム パラメータは同じである必要はありません。詳細については、ポリシー一致の IKE ピア同意, （90 ページ）を参照してください）。

ポリシーを設定しない場合は、デフォルト ポリシー（常に最小優先順位に設定され、各パラメータのデフォルト値を格納しているポリシー）が使用されます。

IKE ポリシーに必要な追加設定

IKE ポリシーで指定した認証方式によっては、IKE および IPSec で正常に IKE ポリシーが使用できるようにするため、特定の追加の設定作業を実行する必要があります。

各認証方式では、次の設定が追加で必要です。

- RSA シグニチャ方式。 RSA シグニチャをポリシーの認証方式として指定する場合、CA から証明書を取得するようにピアを設定できます （CA は、証明書を発行するように正しく設定する必要があります）。この証明書サポートは、「認証局相互運用性の実装」モジュールで説明されているように設定します。

  証明書は公開キーを安全に交換するために各ピアで使用されます（RSA シグニチャでは、各ピアが、リモート ピアの公開シグニチャ キーを持っている必要があります）。双方のピアが有効な証明書を持っている場合、RSA シグニチャを使用する IKE ネゴシエーションの一環として、ピアの間で公開キーが自動的に交換されます。

- RSA 暗号化ナンス方式。 RSA 暗号化ナンスをポリシーの認証方式として指定する場合、各ピアがお互いのピアの公開キーを交換する必要があります。

  RSA シグニチャとは異なり、RSA 暗号化ナンス方式では、証明書を使って公開キーを交換できません。その代わりに各ピアが他のピアの公開キーを持つようにする必要があります。それには次の方法のいずれかを実行します。

  - 証明書を使用する RSA シグニチャを使って IKE 交換がピア間で実行されていることを確認する （証明書を使用すると、RSA シグニチャ ベースの IKE ネゴシエーション中にピアの公開キーが交換されます）。

  IKE 交換が実行されるようにするには、RSA 暗号化ナンスによる優先順位の高いポリシーと、RSA シグニチャによる優先順位の低いポリシーの 2 つのポリシーを指定します。 RSA シグニチャは IKE ネゴシエーションが実行されるときに初めて使用されます。これは、各ピアが他のピアの公開キーをまだ持っていないためです。公開キーが交換されることで、以後の IKE ネゴシエーションで RSA 暗号化ナンスを使用できるようになります。

  この方法では、認証局サポートをあらかじめ設定しておく必要があります。

- 事前共有キー認証方式。 事前共有キーをポリシーの認証方式として指定する場合、これらの事前共有キーを設定する必要があります（ISAKMP 事前共有キーの ISAKMP キーリングでの設定, （110 ページ）を参照）。

RSA 暗号化を設定し、シグニチャ モードがネゴシエーションされ、シグニチャ モードに証明書が使用されると、ピアはシグニチャと暗号キーを要求します。 基本的に、ルータは、設定でサポー

トされているだけキーを要求します。 RSA 暗号化が設定されていない場合は、ルータはシグニチャ キーだけを要求します。

ISAKMP 識別情報

IKE ポリシーで事前共有キーを使用するピアそれぞれについて ISAKMP ID を設定する必要があります。

2 つのピアが IKE を使って IPSec セキュリティ アソシエーションを確立する場合、各ピアが自分の ID をリモート ピアに送信します。 各ピアは、ルータの ISAKMP ID の設定に従い、ホスト名または IP アドレスを送信します。

デフォルトでは、ピアの ISAKMP ID は、ピアの IP アドレスです。 必要に応じて ID をピアのホスト名に変更します。 一般的に、すべてのピアの ID は同じ設定にします（すべてのピアで IP アドレスを設定するか、すべてのピアでホスト名を設定）。 お互いの識別にホスト名を使うピアと IP アドレスを使うピアが混在していると、リモート ピアの ID が識別されない場合にドメイン ネーム サーバ（DNS）ルックアップで ID を解決できなくなり、IKE ネゴシエーションが失敗することがあります。

ISAKMP プロファイルの概要

ISAKMP プロファイルは、インターネット セキュリティ アソシエーションおよびキー管理プロトコル（ISAKMP）設定に関する拡張です。 これにより、フェーズ 1 ネゴシエーションに対する ISAKMP 設定のモジュール性がイネーブルになります。 このモジュール性により、各種 ISAKMP パラメータを各種 IP セキュリティ（IPSec）トンネルに適用し、各種 IPSec トンネルを各種 VPN 転送およびルーティング（VRF）インスタンスにマッピングできます。 現在、ISAKMP プロファイルは、Quality Of Service（QoS）、ルータ証明書管理およびマルチプロトコル ラベル スイッチング（MPLS）VPN 設定など多くのアプリケーションおよび拡張で使用されます。

ISAKMP プロファイルは、一連のピアの IKE フェーズ 1 および IKE フェーズ 1.5（）設定のリポジトリです。 ISAKMP プロファイルは、一致識別基準の概念において一意に識別される着信 IPSec 接続にパラメータを適用します。 これらの基準は、着信 IKE 接続により提供され、IP アドレス、完全修飾ドメイン名（FQDN）およびグループ（バーチャル プライベート ネットワーク（VPN）リモート クライアント グルーピング）を含む IKE アイデンティティに基づきます。 一致識別基準のレベルにより、指定パラメータの適用範囲が決まります。 ISAKMP プロファイルは、トラストポイント、ピア アイデンティティ、および XAUTH 認証、許可、アカウンティング（AAA）リストなど、各プロファイルに固有なパラメータを提供します。 次に、ISAKMP プロファイルを使用する場合のガイドラインを示します。

- ルータでは、サイトごとに異なるフェーズ 1 パラメータを必要とする複数の IPSec 接続を使用します（たとえば、同じルータでサイト間およびリモートアクセスを設定する場合です）。
- VRF 対応 IPSec を使用して IPSec を設定します。これにより、単一の IP アドレスを使用して、異なる IKE フェーズ 1 パラメータの異なるピアに接続できます。 この設定の例については、「VRF 対応の設定：例」を参照してください。

- 異なるカスタム インターネット キー交換（IKE）ピア 1 ポリシーが、各ピアで必要になる場合があります。 たとえば、XAUTH をすべての接続ではなく、特定のピアに適用する場合です。

（注） リモート アクセス IPSec、VRF 対応 IPSec および Xauth はサポートされていません。

インターネット キー交換拡張認証

Xauth を設定するには、次の手順を実行します。

- AAA を設定します（認証リストを設定する必要があります）。 『*Cisco ASR 9000 Series Aggregation Services Router System Security Configuration Guide*』の「*Configuring AAA Services*」モジュールを参照してください。
- **スタティック** クリプト ISAKMP プロファイルを設定します（必須）。 設定の詳細については、ISAKMP プロファイルの設定方法, （120 ページ）を参照してください。
- **ダイナミック** クリプト ISAKMP プロファイルを設定します（任意）。 設定の詳細については、ISAKMP プロファイルの設定方法, （120 ページ）を参照してください。
- ISAKMP ポリシーを設定します（必須）。 設定の詳細については、IKE ポリシーの設定, （100 ページ）を参照してください。

コール アドミッション制御

インターネット キー公開（IKE）のコール アドミッション制御（CAC）は、Cisco IOS XR ソフトウェアの IKE プロトコルへの CAC の適用について記述します。 CAC は、主に、深刻なリソース減少からルータを保護し、クラッシュを防止します。 CAC は、ルータが同時に確立できる IKE セキュリティ アソシエーション（SA）（つまり、CAC へのコール）の数を制限します。 また、システムで許可されるアクティブな IKE SA の最大数、および IKE プロセスまたはグローバル CPU で消費される CPU 使用率を制限するオプションがあります。

IKE セッション

crypto isakmp call admission limit コマンドを使用して、絶対 IKE SA 制限を設定できます。 制限値に達すると、ルータは新しい IKE SA 要求をドロップします。

セキュリティ アソシエーション制限

IKE は接続のパラメータを識別するために、SA を使用します。 IKE では、独自に SA をネゴシエーションして確立できます。 IKE SA は双方向で、IKE だけに使用されます。 IKE SA は IPSec を制限できません。

システムで許可するアクティブな IKE SA の最大数を設定し、**crypto isakmp call admission limit** コマンドを使用することで IKE プロセスまたはグローバル CPU で消費される CPU リソースを制限できます。

IKE は、ユーザが設定した SA 制限値に基づいて SA 要求をドロップします。 IKE SA 制限値を設定するには、**crypto isakmp call admission limit** コマンドを使用します。 ピア ルータから新しい SA 要求があると、IKE はネゴシエーション中のアクティブな IKE SA の数が、設定された SA 制限値を満たしているか、超えているかを判別します。 この数が制限値より大きい、または等しい場合、新しい SA 要求は拒否され、syslog が生成されます。 このログには、SA 要求の送信元および宛先 IP アドレスが含まれます。

IKE SA は IPSec を制限できません。

IP Security VPN モニタリングについて

IP Security（IPSec）VPN Monitoring モニタリング機能では、VPN セッション モニタリング拡張機能によって、バーチャルプライベートネットワーク（VPN）のトラブルシューティングを行い、エンドユーザ インターフェイスをモニタリングできます。 セッション モニタリングには、次の拡張機能が含まれます。

- コンフィギュレーション ファイル内のインターネット キー交換（IKE）ピアの説明を指定する機能。
- 暗号化セッション ステータスの一覧。
- 1 つのコマンドライン インターフェイス（CLI）を使用して、IKE と IP Security（IPSec）の両方のセキュリティ アソシエーション（SA）をクリアする機能。
- **show crypto session** コマンドのオプションを使用してフィルタリング メカニズムを拡張する機能。

IPSec VPN セキュリティ モニタリングを実装するには、次の概念を理解する必要があります。

暗号化セッション バックグラウンド

暗号化セッションは、2 つの暗号エンドポイント間における一連の IPSec 接続（フロー）です。 2 つの暗号エンドポイントで、IKE をキーイング プロトコルとして使用している場合、それらの暗号エンドポイントは互いに対して IKE ピアになります。 一般に、暗号化セッションは、1 つの IKE セキュリティ アソシエーション（制御トラフィック用）と、少なくとも 2 つの IPSec セキュリティ アソシエーション（データ トラフィック用、各方向に 1 つ）で構成されています。 キー再生成中、または両サイドから同時に設定要求が行われたことにより、同じセッションの IKE セ

キュリティ アソシエーション（SA）と IPSec SA が重複したり、IKE SA または IPSec SA が重複したりする可能性があります。

Per-IKE ピアの説明

Per-IKE Peer Description 機能を使用すれば、IKE ピアの選択に関する説明を入力できます。 一意なピアの説明（最大 80 文字）は、特定の IKE ピアを参照する場合に使用されます。 ピアの説明を追加するには、**description (ISAKMP peer)** コマンドを使用します。

この説明フィールドの主要な利用目的はモニタリングです（たとえば、**show** コマンドを使用するときや、ロギング（シスログ メッセージ）などのためです）。 説明フィールドは情報目的です。

暗号化セッション ステータスのサマリー リスト

アクティブ暗号化セッションのステータス情報のリストを取得するには、**show crypto session** コマンドを使用します。 このリストには、暗号化セッションに関する次の概要ステータスが含まれます。

- Interface
- IPSec SA を作成したピアに関連付けられた IKE SA
- セッションのフローにサービスを提供する IPSec SA

（同じセッションの）同じピアには、最大 2 つの IKE SA または複数の IPSec SA が確立されます。その場合、IKE ピアの説明は、ピアと関連付けられている IKE SA と、セッションのフローのサービスを提供している IPSec SA 用に、異なる値で繰り返されます。

また、**detail** キーワードを指定して **show crypto session** コマンドを使用すると、セッションに関する詳細を取得できます。

IKE および IPSec セキュリティ交換のクリア コマンド

clear crypto session コマンドを使用すると、1 つのコマンドで IKE と IPSec の両方をクリアできます。 特定の暗号化セッションや、すべてのセッションのサブセット（たとえば、あるリモート サイトへの単一のトンネル）をクリアするには、ローカルまたはリモート IP アドレス、ローカルまたはリモート ポート、フロント ドア VPN ルーティングおよび転送（FVRF）名、内部 VRF（IVRF）名といった、セッション固有のパラメータを指定する必要があります。 削除する単一のトンネルを指定する場合、リモート IP アドレスを使用するのが一般的です。

clear crypto session コマンドを使用する際にローカル IP アドレスをパラメータとして指定した場合、その IP アドレスをローカル暗号エンドポイント（IKE ローカル アドレス）として共有しているすべてのセッション（およびそれらの IKE SA と IPSec SA）がクリアされます。 パラメータを提供しない場合、ルータのすべての IPSec SA および IKE SA が削除されます。

IPSec Dead Peer Detection 定期メッセージ オプション

ピアとは、IKE チャネルを確立し、それ自体と他のピア間で SA をネゴシエートできる IPSec 対応ノードのことです。 ピアと他のピアとの IP 接続は、ルーティング問題やピアのリロードなどの状況により失われ、パケット トラフィックが損失する場合があります（「ブラック ホール」と呼ばれることがあります）。

IPSec ネットワークの IKE セキュリティ プロトコル設定の実装方法

IPSec ネットワークの IKE セキュリティ プロトコルを設定するには、次の項で説明されている作業を実行します。 最初の 2 つの項の作業は必須です。残りの作業は、設定するパラメータに応じて、実行してください。

- ローカル AAA 方式サーバでの Cisco Easy VPN の設定（任意）
- RSA キーの手動設定, （102 ページ） （IKE パラメータに応じて任意）
- ISAKMP 事前共有キーの ISAKMP キーリングでの設定, （110 ページ） （IKE パラメータに応じて任意）

IKE のイネーブル化またはディセーブル化

インターネット キー交換セキュリティ プロトコルをイネーブルまたはディセーブルにします。

IKE はデフォルトでディセーブルに設定されています。 IKE は個々のインターフェイスに対してイネーブルにする必要はありませんが、ルータのすべてのインターフェイスに対してグローバルにイネーブルにします。

手順の概要

1. **configure**
2. crypto isakmp
3. no crypto isakmp
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	crypto isakmp 例： `RP/0/RSP0RP00/CPU0:router(config)# crypto isakmp`	ピア ルータで IKE をグローバルにイネーブルにします。
ステップ 3	no crypto isakmp 例： `RP/0/RSP0RP00/CPU0:router(config)# no crypto isakmp`	（任意）ピア ルータで IKE をディセーブルにします。

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

IKE ポリシーの設定

IKE ポリシーを設定します。

手順の概要

1. **configure**
2. **crypto isakmp policy** *priority*
3. 次のいずれかを実行します。
   - **encryption** {**192-aes** | **256-aes** | **3des** | **aes** | **des** *}*
   - **encryption {192-aes** *AES - Advanced Encryption Standard (192-bit keys)* | **256-aes** *AES - Advanced Encryption Standard (256-bit keys)* | **3des** *3DES - Three-key triple DES* | **aes** *AES - Advanced Encryption Standard (128 bit keys)* | **des** *DES - Data Encryption Standard (56 bit keys)}*
4. **hash** {**sha** | **md5**}
5. **authentication** {**pre-share** | **rsa-sig** | **rsa-encr**}
6. **group** {**1** | **2** | **5**}
7. **lifetime** *seconds*
8. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
9. show crypto isakmp policy

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**crypto isakmp policy** *priority* 例： `RP/0/RSP0RP00/CPU0:router(config)# crypto isakmp policy 5`	作成するポリシーを識別します。 各ポリシーは、割り当てられた優先順位（1 ～ 10000）により一意に識別されます。 このコマンドによって、ルータは ISAKMP ポリシー コンフィギュレーション モードになります。
ステップ 3	次のいずれかを実行します。 • **encryption** {**192-aes** | **256-aes** | **3des** | **aes** | **des** *}*	暗号化アルゴリズムを指定します。 • **192-aes**：Advanced Encryption Standard、192 ビット キー

	コマンドまたはアクション	目的
	• **encryption {192-aes** *AES - Advanced Encryption Standard (192-bit keys)* | **256-aes** *AES - Advanced Encryption Standard (256-bit keys)* | **3des** *3DES - Three-key triple DES* | **aes** *AES - Advanced Encryption Standard (128 bit keys)* | **des** *DES - Data Encryption Standard (56 bit keys)}* 例： `RP/0/RSP0RP00/CPU0:router(config-isakmp)# encryption aes`	• **256-aes**：Advanced Encryption Standard、256 ビット キー • **3des**：3 キー トリプル Data Encryption Standard • **aes**：Advanced Encryption Standard、128 ビット キー • **des**：Data Encryption Standard、56 ビット キー
ステップ 4	**hash** {**sha** | **md5**} 例： `RP/0/RSP0RP00/CPU0:router(config-isakmp)# hash md5`	ハッシュ アルゴリズムを指定します。 • SHA：セキュア ハッシュ アルゴリズム • MD5：メッセージ ダイジェスト 5 （注） SHA および MD5 は、Hashed Message Authentication Coding（HMAC）の計算に使用できます。
ステップ 5	**authentication** {**pre-share** | **rsa-sig** | **rsa-encr**} 例： `RP/0/RSP0RP00/CPU0:router(config-isakmp)# authentication rsa-sig`	このポリシーの認証方式を事前共有キー、RSA暗号化または RSA シグニチャを指定します。
ステップ 6	**group** {**1** | **2** | **5**} 例： `RP/0/RSP0RP00/CPU0:router(config-isakmp)# group 5`	Diffie-Hellman グループ ID を指定します。
ステップ 7	**lifetime** *seconds* 例： `RP/0/RSP0RP00/CPU0:router(config-isakmp)# lifetime 50000`	セキュリティ アソシエーションのライフタイムを指定します。 範囲は 60 ～ 86400 秒です。
ステップ 8	次のいずれかのコマンドを使用します。 • **end** • **commit**	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`

	コマンドまたはアクション	目的

	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ステップ 9	show crypto isakmp policy 例： `RP/0/RSP0RP00/CPU0:router# show crypto isakmp policy`	（任意）既存の IKE ポリシーをすべて表示します。

RSA キーの手動設定

RSA 暗号化ナンスを IKE ポリシーの認証方式として指定し、CA を使用しない場合、手動で RSA キーを設定します。

RSA キーを手動で設定するには、IKE ポリシーで RSA 暗号化ナンスを使用する IPSec ピアそれぞれについて、この作業を実行します。

RSA キーの生成

RSA キーの生成方法については、「認証局相互運用性の実装」モジュールのRSA キー ペアの生成, （73 ページ）を参照してください。

ISAKMP ID の設定

ピアの ISAKMP ID を設定します。

IKE ポリシーで事前共有キーを使用する各ピアに対して、これらの手順を繰り返します。

手順の概要

1. **configure**
2. **crypto isakmp identity** {**address** | **hostname**}
3. **host** *hostname address1 [address2...address8]*
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**crypto isakmp identity** {**address**	**hostname**} 例： `RP/0/RSP0RP00/CPU0:router(config)# crypto isakmp identity address`	（ローカル ピア）IPアドレスまたはホスト名により、ピアのISAKMP ID を指定します。 IP アドレスおよびホスト名の使用方法については、**crypto isakmp identity** コマンドの説明を参照してください。
ステップ 3	**host** *hostname address1 [address2...address8]* 例： `RP/0/RSP0RP00/CPU0:router(config)# host host1 10.0.0.5`	（すべてのリモート ピア）すべてのリモート ピアでピアのホスト名をその IP アドレスにマッピングします。 • このコマンドは、ローカル ピアの ISAKMP ID がホスト名を使用して指定された場合に使用されます。 • ホスト名または IP アドレスが DNS サーバでマップ済みの場合はこの手順は不要です。
ステップ 4	次のいずれかのコマンドを使用します。 • **end** • **commit** 例： `RP/0/RSP0/CPU0:router(config)# end`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` ◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。

	コマンドまたはアクション	目的
	または `RP/0/RSP0/CPU0:router(config)# commit`	◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 ◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

その他のすべてのピアの RSA 公開キーの設定

この作業では、その他のすべてのピアの RSA 公開キーを設定します。

IKE ポリシーで RSA 暗号化ナンスを使用するピアそれぞれについて、必ず上記の作業を繰り返してください。

手順の概要

1. **configure**
2. **crypto keyring** *keyring-name* [**vrf** *fvrf-name*]
3. **rsa-pubkey** {**address** *address* | **name** *fqdn*} [**encryption** | **signature**]
4. **address** *ip-address*
5. **key-string** **key-string**
6. **quit**
7. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
8. **show crypto pubkey-chain rsa** [**name key-name** | **address key-address**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ 2	**crypto keyring** *keyring-name* [**vrf** *fvrf-name*]
	
	例：
	
	`RP/0//CPU0:router(config)# crypto keyring vpnkeyring`
	`RP/0//CPU0:router(config-keyring)#`
		IKE 認証時の暗号キーリングを定義します
		• キーリング コンフィギュレーション モードを開始します。
		• *keyring-name* 引数を使用して、暗号化キーリングの名前を指定します。
		• （任意）**vrf** キーワードを使用して、前面扉仮想ルーティングおよび転送（FVRF）の名前が、参照されているキーリングであることを指定します。

ステップ 3	**rsa-pubkey** {**address** *address* | **name** *fqdn*} [**encryption** | **signature**]
	
	例：
	
	`RP/0//CPU0:router(config-keyring)# rsa-pubkey name host.vpn.com`
	`RP/0//CPU0:router(config-pubkey)`
		インターネット キー交換（IKE）認証時の暗号化またはシグニチャに使用される Rivest, Shamir, and Adelman（RSA）手動キーを定義します。
		• **address** キーワードを使用して、リモート ピアの RSA 公開キーの IP アドレスを指定します。 address 引数は、手動で設定するリモート ピアのリモート RSA 公開キーの IP アドレスです。
		• **name** キーワードを使用して、ピアの完全修飾ドメイン名（FQDN）を指定します。
		• **encryption** キーワードを使用して、キーが暗号化に使用されることを指定します。
		• **signature** キーワードを使用して、キーがシグネチャに使用されることを指定します。 **signature** キーワードがデフォルトです。

ステップ 4	**address** *ip-address* 例： `RP/0//CPU0:router(config-pubkey)# address 10.5.5.1`	リモート ピアの IP アドレスを指定します。 • このコマンドは、リモートピアの名前に完全修飾ドメイン名を使用した場合に使用できます。
ステップ 5	**key-string key-string** 例： `RP/0//CPU0:router(config-pubkey)# key-string 005C300D 06092A86 4886F70D 01010105 ...`	リモート ピアの RSA 公開キーを指定します。 • これは、リモート ルータの RSA キーが生成されたときに、リモートピアの管理者によって以前に表示されていたキーです。 • 間違いを防ぐために、（データに手入力するのではなく）キーデータをカット アンド ペーストしてください。

	コマンドまたはアクション	目的
		• 各行にキーを入力します。 キーの前に **key-string** コマンドを入力する必要があります。 • リモート ピアの RSA キーの指定が完了したら、公開キー コンフィギュレーション プロンプトに **quit** と入力してグローバル コンフィギュレーション モードに戻ります。
ステップ 6	**quit** 例： `RP/0//CPU0:router(config-pubkey)# quit`	グローバル コンフィギュレーション モードに戻ります。

ステップ 7	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ステップ 8	**show crypto pubkey-chain rsa** [**name** *key-name* | **address** *key-address*] 例： `RP/0//CPU0:router# show crypto pubkey-chain rsa`	（任意）ルータに保存されているすべての RSA 公開キーのリストを表示します。 • ルータに保存されている特定の RSA 公開キーの詳細を表示するには、オプションの name または address キーワードを使用します。

RSA ベースのユーザ認証の公開キーのインポート

この作業では、RSA 公開キーをルータにインポートします。

手順の概要

1. **configure**
2. **crypto key import authentication rsa** {**address** *address* | **name** *fqdn*}
3. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
4. **show crypto key import authentication rsa** {**address** *address* | **name** *fqdn*}

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ 2	**crypto key import authentication rsa** {**address** *address*	**name** *fqdn*}
		
		例：
		`RP/0/RSP0RP00/CPU0:router(config)# crypto key import authentication rsa tftp://223.255.254.254/ssh/public.pub (in base64)`
		`RP/0/RSP0RP00/CPU0:router(config-keyring)#`
			公開キーをルータにインポートします。
			• **address** キーワードを使用して、リモート ピアの RSA 公開キーの IP アドレスを指定します。 address 引数は、手動で設定するリモート ピアのリモート RSA 公開キーの IP アドレスです。
			• **name** キーワードを使用して、ピアの完全修飾ドメイン名（FQDN）を指定します。

ステップ 3	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。

	コマンドまたはアクション	目的
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 ◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 4	**show crypto key import authentication rsa** {**address** *address* | **name** *fqdn*} 例： `RP/0/RSP0RP00/CPU0:router# show crypto key import authentication rsa`	（任意）ルータにインポートされたすべての RSA 公開キーのリストを表示します。 • ルータに保存されている特定の RSA 公開キーの詳細を表示するには、オプションの name または address キーワードを使用します。

RSA 公開キーのルータからの削除

この作業では、RSA 公開キーをルータから削除します。

手順の概要

1. **configure**
2. **zeroize crypto key import authentication rsa** {**address** *address* | **name** *fqdn*}
3. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
4. **show crypto pubkey-chain rsa** [**name key-name** | **address key-address**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ 2	**zeroize crypto key import authentication rsa** {**address** *address* | **name** *fqdn*}
	
	例：
	
	`RP/0/RSP0RP00/CPU0:router(config)# zeroize crypto key import authentication rsa tftp://223.255.254.254/ssh/public.pub (in base64)`
	`RP/0/RSP0RP00/CPU0:router(config-keyring)#`
		公開キーをルータから削除します。
		
		• **address** キーワードを使用して、リモート ピアの RSA 公開キーの IP アドレスを指定します。 address 引数は、手動で設定するリモート ピアのリモート RSA 公開キーの IP アドレスです。
		
		• **name** キーワードを使用して、ピアの完全修飾ドメイン名（FQDN）を指定します。

ステップ 3	次のいずれかのコマンドを使用します。
	
	• **end**
	
	• **commit**
	
	例：
	
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 4	**show crypto pubkey-chain rsa** [**name** *key-name* | **address** *key-address*] 例： `RP/0/RSP0RP00/CPU0:router# show crypto pubkey-chain rsa`	（任意）ルータに保存されているすべての RSA 公開キーのリストを表示します。 • ルータに保存されている特定の RSA 公開キーの詳細を表示するには、オプションの name または address キーワードを使用します。

ISAKMP 事前共有キーの ISAKMP キーリングでの設定

この作業では、ISAKMP 事前共有キーを ISAKMP キーリングに設定します。

はじめる前に

ISAKMP 事前共有キーを ISAKMP キーリングに設定するには、IKE ポリシーで事前共有キーを使用するピアそれぞれについて、以下の作業を実行します。

- 各ピアの ISAKMP ID を設定します。 各ピアの ID は、ピアのホスト名または IP アドレスに設定する必要があります。 デフォルトでは、ピアの ID はピアの IP アドレスに設定されています。 ISAKMP ID の設定については、ISAKMP ID の設定，（102 ページ）を参照してください。
- 各ピアに共有キーを指定します。 指定した事前共有キーは 2 つのピア間で共有されていることに注意してください。 任意のピアに同一のキーを指定することで、複数のリモート ピアとの共有が可能になります。ただし、ピアの各ペアに個別のキーを指定する方が安全性は高くなります。
- マスク事前共有キーのサポートを指定する必要があります。 IKE ポリシーで事前共有キーを使用する各ピアに対して、これらの手順を繰り返します。

手順の概要

1. **configure**
2. **crypto keyring** *keyring-name* [**vrf** *vrf-name*]
3. **pre-shared-key** {**address** *address* [*mask*] | **hostname** *hostname*} **key** *key*
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ 2	**crypto keyring** *keyring-name* [**vrf** *vrf-name*]
	
	例：
	`RP/0/RSP0RP00/CPU0:router(config)# crypto keyring vpnkeyring`
	`RP/0/RSP0RP00/CPU0:router(config-keyring)#`
		IKE 認証時の暗号キーリングを定義します。
		• *keyring-name* 引数を使用して、暗号化キーリングの名前を指定します。
		•（任意）**vrf** キーワードを使用して、前面扉仮想ルーティングおよび転送（FVRF）の名前が、参照されているキーリングであることを指定します。

ステップ 3	**pre-shared-key** {**address** *address* [*mask*] | **hostname** *hostname*} **key** *key*
	
	例：
	`RP/0/RSP0RP00/CPU0:router(config-keyring)# pre-shared-key address 10.72.23.11 key vpnkey`
	`RP/0/RSP0RP00/CPU0:router(config-keyring)# pre-shared-key hostname mycisco.com key vpnkey`
		IKE 認証の事前共有キーを定義します。
		• **address** キーワードを使用して、リモート ピアの IP アドレスまたはサブネットとマスクを指定します。
		•（任意）*mask* 引数を使用して、アドレスの範囲を照合します。
		• **hostname** キーワードを使用して、ピアの完全修飾ドメイン名（FQDN）を指定します。
		•（任意）**key** キーワードを使用して、キーを指定します。

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。

	コマンドまたはアクション	目的
		◦cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

コール アドミッション制御の設定

コール アドミッション制御（CAC）を設定するには、次の作業を実行します。

IKE セキュリティ アソシエーション制限の設定

この作業では、IKE セキュリティ アドミッション制限を設定します。

手順の概要

1. configure
2. **crypto isakmp call admission limit** {**in-negotiation-sa** *number* | **sa** *number*}
3. 次のいずれかを実行します。
   - end
   - commit
4. show cyrpto isakmp call admission statistics

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	configure 例： `RP/0/RSP0RP00/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**crypto isakmp call admission limit** {**in-negotiation-sa** *number* | **sa** *number*} 例： `RP/0/RSP0RP00/CPU0:router(config)# crypto isakmp call admission limit sa 25`	IKE が新しい SA 要求の拒否を開始するまでにルータが確立できる、IKE SA の最大数を指定します。 • **in-negotiation-sa** キーワードを使用して、IKE が新しい SA 要求の拒否を開始するまでにルータが確立できる、ネゴシエーション中（初期）IKE セキュリティ アソシエーション（SA）の最大数を指定します。 number 引数の範囲は 1 ～ 100000 です。 • **sa** キーワードを使用して、ルータが確立できるアクティブな IKE SA の最大数を指定します。 *number* 引数の範囲は 1 ～ 100000 です。

ステップ 3	次のいずれかを実行します。
	• end
	• commit
	
	例：
	
	`RP/0/RSP0RP00/CPU0:router(config)# end`
	
	または
	
	`RP/0/RSP0RP00/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)?[cancel]:`
		
		◦ **yes** と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ステップ 4	show cyrpto isakmp call admission statistics 例： `RP/0/RSP0RP00/CPU0:router# show crypto isakmp call admission statistics`	暗号化 CAC 統計情報をモニタします。

システム リソース制限の設定

この作業では、システム リソース制限を設定します。

手順の概要

1. configure
2. **crypto isakmp call admission limit** {**cpu** {**total** *percent* | **ike** *percent*}}
3. 次のいずれかを実行します。
   - end
   - commit
4. show cyrpto isakmp call admission statistics

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	configure 例： `RP/0/RSP0RP00/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**crypto isakmp call admission limit** {**cpu** {**total** *percent* | **ike** *percent*}} 例： `RP/0/RSP0RP00/CPU0:router(config)# crypto isakmp call admission limit cpu total 90`	IKE が新しい SA 要求の拒否を開始するまでにルータが確立できる、IKE SA の最大数を指定します。 • **cpu** キーワードを使用して、CPU 使用率の合計リソース制限を指定します。 • **total** キーワードを使用して、新しいコールを受け付ける最大合計 CPU 使用率を指定します。 *percent* 引数の範囲は 1 ～ 100 です。 • **ike** キーワードを使用して、新しいコールを受け付ける最大 IKE CPU 使用率を指定します。 percent 引数の範囲は 1 ～ 100 です。
ステップ 3	次のいずれかを実行します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• end
	• commit
	
	例：
	`RP/0/RSP0RP00/CPU0:router(config)# end`
	
	または
	
	`RP/0/RSP0RP00/CPU0:router(config)# commit`
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)?[cancel]:`
		
		◦ **yes** と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ステップ 4	show cyrpto isakmp call admission statistics 例： `RP/0/RSP0RP00/CPU0:router# show cyrpto isakmp call admission statistics`	暗号化 CAC 統計情報をモニタします。

暗号化キーリングの設定

暗号化キーリングは、事前共有キーおよび Rivest, Shamir, and Adelman（RSA）公開キーのリポジトリです。ルータは 0 個以上のキーリングを保持できます。オプションとして、各キーリングでは、キーリング内に定義されたキーが属する VRF を指定できます。

この作業では、暗号化キーリングを設定します。

はじめる前に

（注） 暗号化キーリングを設定する際は、次のガイドラインと制約事項に従ってください。

- 暗号化キーリングに関連付けられている VRF は変更できません。 新しい VRF 値には、別のキーリングを設定する必要があります。
- キーリング内のアドレスの重複は許可されておらず、設定時に適用する必要があります。
- 暗号化キーリングは 1 つ以上の ISAKMP プロファイルに付加され、使用中は削除できません。

手順の概要

1. **configure**
2. **crypto keyring** *keyring-name* [**vrf** *fvrf-name*]
3. **description** *string*
4. **local-address** *ip-address*
5. **pre-shared-key** {**address** *address* [*mask*] | **hostname** *hostname*} **key** *key*
6. **rsa-pubkey** {**address** *address* | **name** *fqdn*} [**encryption** | **signature**]
7. **key-string** *key-string*
8. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**crypto keyring** *keyring-name* [**vrf** *fvrf-name*] 例： `RP/0/RSP0/CPU0:router(config)# crypto keyring vpnkey`	IKE 認証時に使用する暗号化キーリングを定義します。 • *keyring-name* 引数は、暗号化キーリングの名前として使用されます。 • **vrf** キーワードを使用して、前面扉仮想ルーティングおよび転送（FVRF）の名前が、参照されているキーリングであることを指定します。 *fvrf-name* 引数は、VRF 設定時に定義された FVRF 名と一致する必要があります。

	コマンドまたはアクション	目的
ステップ 3	**description** *string* 例： `RP/0/RSP0/CPU0:router(config-keyring)# description this is a sample keyring`	キーリングの 1 行説明を作成します。 • *string* 引数を使用して、キーリングを説明する文字列を指定します。
ステップ 4	**local-address** *ip-address* 例： `RP/0/RSP0/CPU0:router(config-keyring)# local-address 10.40.1.1`	ISAKMP キーリング設定の範囲を、ローカル ターミネーション アドレスまたはインターフェイスに限定します。 • *ip-address* 引数を使用して、バインド先の IP アドレスを指定します。
ステップ 5	**pre-shared-key** {**address** *address* [*mask*] | **hostname** *hostname*} **key** *key* 例： `RP/0/RSP0/CPU0:router(config-keyring)# pre-shared-key address 10.72.23.11 key vpnkey`	IKE 認証に使用する事前共有キーを定義します。 • **address** キーワードを使用して、リモート ピアの IP アドレスまたはサブネットとマスクを指定します。 *mask* 引数はオプションです。 • **hostname** キーワードを使用して、ピアの完全修飾ドメイン名（FQDN）を指定します。 • **key** キーワードを使用して、秘密を指定します。
ステップ 6	**rsa-pubkey** {**address** *address* | **name** *fqdn*} [**encryption** | **signature**] 例： `RP/0/RSP0/CPU0:router(config-keyring)# rsa-pubkey name host.vpn.com`	アドレスまたはホスト名によって Rivest, Shamir, and Adelman（RSA）公開キーを定義します。 • **address** キーワードを使用して、リモート ピアの RSA 公開キーの IP アドレスを指定します。 *address* 引数は、手動で設定するリモート ピアのリモート RSA 公開キーの IP アドレスです。 • **name** キーワードを使用して、ピアの FQDN を指定します。 •（任意）**encryption** キーワードを使用して、キーが暗号化に使用されることを指定します。 •（任意）**signature** キーワードを使用して、キーがシグネチャに使用されることを指定します。 **signature** キーワードがデフォルトです。
ステップ 7	**key-string** *key-string* 例： `RP/0/RSP0/CPU0:router(config-pubkey)# key-string 005C300D 06092A86 4886F70D 01010105`	リモート ピアの RSA 公開キーを手動で指定します。
ステップ 8	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

IP セキュリティ VPN モニタリングの設定

次に、IP セキュリティ（IPSec）VPN モニタリングを設定する方法について説明します。

IKE ピアの説明の追加

この作業では、IKE ピアの説明を IPSec VPN セッションに追加できます。

手順の概要

1. **configure**
2. **crypto isakmp peer** {**address** *ip-address* | **hostname** *hostname*} [**description** *string* | **vrf** *fvrf-name*]
3. **description** *string*
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
5. **show crypto isakmp peers** [*ip-address* | **vrf** *vrf-name*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ 2	**crypto isakmp peer** {**address** *ip-address* | **hostname** *hostname*} [**description** *string* | **vrf** *fvrf-name*]
	例：
	`RP/0/RSP0RP00/CPU0:router(config)# crypto isakmp peer address 1040.40.40.2`
	`RP/0/RSP0RP00/CPU0:router(config-isakmp-peer)`
		アグレッシブ モードで、トンネル属性の認証、許可、アカウンティング（AAA）に関する IKE クエリー生成のための IPSec ピアをイネーブルにして、ISAKMP ピア コンフィギュレーション モードを開始します。
		セッションの IPSec ピアを指定します。

ステップ 3	**description** *string* 例： `RP/0/RSP0RP00/CPU0:router(config-isakmp-peer)# description citeA`	1 行のテキスト文字列による IKE ピアの説明を追加します。 • ピアの説明は最大 80 文字です。 •

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。

	コマンドまたはアクション	目的
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 ◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 5	**show crypto isakmp peers** [*ip-address* | **vrf** *vrf-name*] 例： `RP/0/RSP0RP00/CPU0:router# show crypto isakmp peers`	ピアの説明を表示します。

暗号化セッションのクリア

ユーザおよびグループの暗号化セッション（IP セキュリティ（IPSec）およびインターネット キー交換（IKE）セキュリティ アソシエーション（SA））を削除するには、EXEC モードで **clear crypto session** コマンドを使用します。

ISAKMP プロファイルの設定方法

この作業では、サービスインターフェイスまたはトンネルインターフェイスの ISAKMP プロファイル（ピアのセットに対するコマンドのリポジトリ）を設定します。

（注） Cisco ASR 9000 Series Router はサービス インターフェイスのみをサポートします。

手順の概要

1. **configure**
2. **crypto isakmp profile** {**local** [**local**] *profile-name*}
3. **description** *string*
4. **keepalive disable**
5. **self-identity** {**address** | **fqdn** | **user-fqdn** *user-fqdn*}
6. **keyring** *keyring-name*
7. **match identity** {**group** *group-name* | **address** *address* [*mask*] **vrf** [*fvrf*] | **host** *hostname* | **host domain** *domain-name* | **user** *username* | **user domain** *domain-name*}
8. 次のいずれかを実行します。
   - **set interface** { **tunnel-ipsec** *intf-index* |
   - **service-gre** *intf-index* | **service-ipsec** *intf-index* }
9. **set ipsec-profile** *profile-name*
10. 次のいずれかのコマンドを使用します。
    - **end**
    - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ 2	**crypto isakmp profile** {**local** [**local**] *profile-name*}
	
	例：
	
	`RP/0/RSP0RP00/CPU0:router(config)# crypto isakmp profile local vpnprofile`
	`RP/0/RSP0RP00/CPU0:router(config-isa-prof)#`
		ISAKMP プロファイルを定義し、IPSec ユーザ セッションを監査します。
		• （必須/任意）**local** キーワードを使用して、プロファイルがローカルを送信元または終端とするトラフィックに使用されることを指定します。
		
		（注） **local** キーワードは、この手順の後半で **set ipsec-profile** コマンドや **set interface tunnel-ipsec** コマンドを使用する場合には必須です。
		
		• （必須）*profile-name* 引数を使用して、ユーザ プロファイルの名前を指定します。

	コマンドまたはアクション	目的
ステップ 3	**description** *string* 例： `RP/0/RSP0RP00/CPU0:router(config-isa-prof)# description this is a sample profile`	キーリングの説明を作成します。 • *string* 引数を使用して、キーリングを説明する文字列を指定します。
ステップ 4	**keepalive disable** 例： `RP/0/RSP0RP00/CPU0:router(config-isa-prof)# keepalive disable`	ゲートウェイが DPD メッセージを Cisco IOS XR ピアに送信できるようにします。 • **disable** キーワードを使用して、キープアライブのグローバル宣言をディセーブルにします。
ステップ 5	**self-identity** {**address** | **fqdn** | **user-fqdn** *user-fqdn*} 例： `RP/0/RSP0RP00/CPU0:router(config-isa-prof)# self-identity user-fqdn user@vpn.com`	ローカル IKE がリモート ピアに対して IKE 自身を識別させるために使用する ID を定義します。 • **address** キーワードを使用して、ローカルエンドポイントの IP アドレスを指定します。 • **fqdn** キーワードを使用して、ホストの完全修飾ドメイン名（FQDN）を指定します。 • **user-fqdn** キーワードを使用して、ユーザの FQDN がリモート エンドポイントに送信されたことを指定します。
ステップ 6	**keyring** *keyring-name* 例： `RP/0/RSP0RP00/CPU0:router(config-isa-prof)# keyring vpnkeyring`	ISAKMP プロファイルとともにキーリングを定義します。 • *keyring-name* 引数を使用して、キーリング名を指定します。このキーリング名は、グローバルコンフィギュレーションで定義したキーリング名と一致している必要があります。

ステップ 7	**match identity** {**group** *group-name* | **address** *address* [*mask*] **vrf** [*fvrf*] | **host** *hostname* | **host domain** *domain-name* | **user** *username* | **user domain** *domain-name*}
	
	例：
	`RP/0/RSP0RP00/CPU0:router(config-isa-prof)# match identity group vpngroup`
	`RP/0/RSP0RP00/CPU0:router(config-isa-prof-match)#`
		ISAKMP プロファイルのピアの ID を照合します。
		• **group** キーワードを使用して、識別子（ID）タイプ ID_KEY_ID と一致する Unity グループを指定します。RSA シグニチャを使用する場合、*group-name* 引数は認定者名（DN）の組織ユニット（OU）フィールドと一致します。
		• **address** キーワードを使用して、*address* 引数を ID タイプ ID_IPV4_ADDR と照合します。
		• *mask* 引数を使用して、アドレスの範囲を指定します。

	コマンドまたはアクション	目的
		• **vrf** キーワードを使用して、ピアの前面扉 VPN ルーティングおよび転送（VRF）を指定します。 • *fvrf* 引数を使用して、前面扉仮想ルータ転送（FVRF）バーチャル プライベート ネットワーク（VPN）領域のアドレスを照合します。 • **host** キーワードを使用して、完全修飾ドメイン名（FQDN）がドメイン名で終わるタイプ ID_FQDN と一致する ID を指定します。 • **host domain** キーワードを使用して、タイプ ID_FQDN と一致する ID を指定します。 ドメイン名は *domain-name* 引数と同じです。 • **user** キーワードを使用して、FQDN と一致する ID を指定します。 • **user domain** キーワードを使用して、タイプ ID_USER_FQDN と一致する ID を指定します。 **user domain** キーワードが存在する場合、タイプ ID_USER_FQDN の識別情報を持ち、*domain-name* で終わるすべてのユーザが照合されます。
ステップ 8	次のいずれかを実行します。 • **set interface** { **tunnel-ipsec** *intf-index* | • **service-gre** *intf-index* | **service-ipsec** *intf-index* } 例： `RP/0/RP0/CPU0:router(config-isa-prof-match)# set interface tunnel-ipsec 50` または RP/0/ RSP0 0 /CPU0:router(config-isa-prof-match)# set interface tunnel service - ipsec 50 gre 34	• **set interface tunnel-ipsec** コマンドが使用できるのは、以前に **local** キーワードを選択した場合だけです。このコマンドは、ローカルを送信元または終端とするトラフィックの IPSec サービス アソシエーション（SA）について IKE がネゴシエートし、ローカル エンドポイントが IKE の応答側である際に、インターフェイス インスタンスを事前定義します。 • *intf-index* 引数を使用して、範囲を 0 ～ 4294967295 に設定します。 • **set interface service-gre** コマンドおよび **set interface service-ipsec** コマンドが使用できるのは、Cisco ASR 9000 Series Router の場合だけです。このコマンドは、リモートを送信元または終端とするトラフィックの IPSec SA について IKE がネゴシエートする際に、インターフェイス インスタンスを事前定義します。 • *intf-index* 引数の範囲は、設定対象がトンネルかサービスかによって異なります。

	コマンドまたはアクション	目的
		◦トンネル = 0 ～ 429496729 ◦サービス = 1 ～ 65535
ステップ 9	**set ipsec-profile** *profile-name* 例： `RP/0/RSP0RP00/CPU0:router(config-isa-prof-match)# set ipsec-profile myprofile`	（任意）ローカルを送信元または終端とするトラフィックの IPSec サービス アソシエーション（SA）について IKE がネゴシエートし、ローカルエンドポイントが IKE の応答側である際に、IPSec プロファイルインスタンスを事前定義します。 • *profile-name* 引数を使用して、IPSec プロファイルの名前を設定します。 • （注） この手順の前半で **local** キーワードを選択した場合のみ、使用できます。

ステップ 10	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

Dead Peer Detection 定期メッセージの設定方法

この作業では、定期的なキープアライブやオンデマンドの Dead Peer Detection（DPD）メッセージを設定します。

手順の概要

1. **configure**
2. **crypto isakmp keepalive***seconds retry-seconds*[**periodic** | **on-demand**]
3. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**crypto isakmp keepalive***seconds retry-seconds*[**periodic**	**on-demand**] 例： `RP/0/RSP0/CPU0:router(config)# crypto isakmp keepalive 20 20 on-demand`	IKE セキュリティ アソシエーション（SA）機能を使用すると、2 つの IP セキュリティ（IPSec）ピア間の接続切断を検出するメカニズムが提供されます。 • *seconds* 引数を使用して、キープアライブ メッセージ間の秒数を指定します。 範囲は 10 ～ 3600 です。 • *retry-seconds* 引数を使用して、キープアライブが失敗した場合のリトライ間の秒数を指定します。 範囲は 2 ～ 60 です。 •（任意）**periodic** キーワードを使用して、DPD メッセージに対してキープアライブ メッセージを定期的に送信することを指定します。 •（任意）**on-demand** キーワードを使用して、DPD のリトライをオンデマンドで送信することを指定します。
ステップ 3	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

IKE セキュリティ プロトコルの実装の設定例

ここでは、次の設定例について説明します。

IKE ポリシーの作成：例

この例では、2 つの IKE ポリシー（最大のプライオリティとして policy 15、次のプライオリティとして policy 20）を作成し、最小のプライオリティとして既存のデフォルト プライオリティを使用する方法について説明します。

```
crypto isakmp policy 15
encryption 3des
hash md5
authentication rsa-sig
group 2
lifetime 5000
crypto isakmp policy 20
authentication pre-share
lifetime 10000
```

この例では、暗号化アルゴリズム パラメータのデフォルト値のため、policy 20 の**暗号化 des** は記述した設定に表示されません。

この設定で **show crypto isakmp policy** コマンドを発行すると、出力は次のようになります。

```
Protection suite priority 15
encryption algorithm:3DES - Data Encryption Standard (168 bit keys)
hash algorithm:Message Digest 5
authentication method:Rivest-Shamir-Adelman Signature
Diffie-Hellman group:#2 (1024 bit)
lifetime:5000 seconds, no volume limit
Protection suite priority 20
encryption algorithm:DES - Data Encryption Standard (56 bit keys)
hash algorithm:Secure Hash Standard
authentication method:preshared Key
Diffie-Hellman group:#1 (768 bit)
lifetime:10000 seconds, no volume limit
Default protection suite
encryption algorithm:DES - Data Encryption Standard (56 bit keys)
hash algorithm:Secure Hash Standard
authentication method:Rivest-Shamir-Adelman Signature
Diffie-Hellman group:#1 (768 bit)
lifetime:86400 seconds, no volume limit
```

（注） ライフタイムに「no volume limit」と出力されていますが、time ライフタイム（86,400 秒など）だけは設定できます。volume-limit ライフタイムは設定できません。

ローカル IP アドレスに基づいた特定のポリシー セットへの IKE ピアの制限：例

最初の部分は、暗号化方式に関連した ISAKMP ポリシーの選択と SVI トンネル ソースの指定から構成されています。 次の例で、IP アドレス 1.1.1.1 に接続しているユーザには、ISAKMP ポリシーとして DES が採用されます。 ただし、IP アドレス 2.2.2.2 に接続しているユーザには、ISAKMP ポリシーとして AES のみが採用されます。

照合には、複数の ISAKMP ポリシー、または複数の IP アドレスを使用できます。 残りの設定、つまり SVI に設定されたグループ名に一致する ISAKMP プロファイルの設定も同様です。

この特定の例では、ポリシー セット内に 2 つのポリシー（policy 10 と policy 20）が設定されています。

例では、SVI1 トンネル ソースと SVI2 トンネル ソースは、それぞれ太字で **local-address 10 1 . 10 1 .1.1 および local-address 10 2 . 10 2 .2.2** として指定されていることに注意してください。 以下に例を示します。

```
RP/0/RSP0RP00/CPU0:router:: configure
RP/0/RSP0RP00/CPU0:router(config)# crypto isakmp policy 10
RP/0/RSP0RP00/CPU0:router(config-isakmp)# encryption des << restricts use to DES only
RP/0/RSP0RP00/CPU0:router(config-isakmp)# group 2
RP/0/RSP0RP00/CPU0:router(config-isakmp)## authentication pre-share
RP/0/RSP0RP00/CPU0:router(config)## crypto isakmp policy 20
RP/0/RSP0RP00/CPU0:router(config-isakmp)# encryption aes << restricts use to AES only
RP/0/RSP0RP00/CPU0:router(config-isakmp)# group 2
RP/0/RSP0RP00/CPU0:router(config-isakmp)# authentication pre-share
RP/0/RSP0RP00/CPU0:router(config)# crypto isakmp policy-set policy_1<< match ID
RP/0/RSP0RP00/CPU0:router(config-isakmp-pol-set)# policy 10 << routing priority
RP/0/RSP0RP00/CPU0:router(config-isakmp-pol-set)# match identity local-address 101.101.1.1
```

```
RP/0/RSP0RP00/CPU0:router(config)# crypto isakmp policy-set policy_2<< match ID
RP/0/RSP0RP00/CPU0:router(config-isakmp-pol-set)# crypto isakmp policy-set policy_2<< match
IDpolicy 20
RP/0/RSP0RP00/CPU0:router(config-isakmp-pol-set)# policy 20match identity local-address 2.2.2.2
RP/0/RSP0RP00/CPU0:router(config-isakmp-pol-set)# match identity local-address 10.10.2.2commit
RP/0/RSP0RP00/CPU0:router(config-isakmp-pol-set)# commitexit
RP/0/RSP0RP00/CPU0:router(config-isakmp-pol-set)# exit#
```

参考資料

ここでは、IKE セキュリティ プロトコルの実装に関連する参考資料を紹介します。

関連資料

関連項目	ドキュメント名
IKE セキュリティ プロトコル コマンド：コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』
IPSec-related オブジェクト トラッキング コマンド：コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco ASR 9000 Series Aggregation Services Router System Management Command Reference*』
オブジェクト トラッキング設定手順、および例	『*Cisco ASR 9000 Series Aggregation Services Router System Management Configuration Guide*』
IPSec ネットワーク セキュリティ コマンド：コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』 の *IPSec* ネットワーク セキュリティ コマンド

標準

標準	タイトル
この機能でサポートが追加または変更された標準はありません。また、この機能で変更された既存の標準のサポートはありません。	—

MIB

MIB	MIB リンク
—	Cisco IOS XR ソフトウェアを使用している MIB を特定してダウンロードするには、次の URL にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
RFC 2401	『Security Architecture for the Internet Protocol』
RFC 2402	『IP Authentication Header』
RFC 2403	『The Use of HMAC-MD5-96 within ESP and AH』
RFC 2404	『The Use of HMAC-SHA-1-96 within ESP and AH』
RFC 2405	『The ESP DES-CBC Cipher Algorithm With Explicit IV』
RFC 2406	『IP Encapsulating Security Payload (ESP)』
RFC 2407	『The Internet IP Security Domain of Interpretation for ISAKMP』
RFC 2408	『Internet Security Association and Key Management Protocol (ISAKMP)』
RFC 2409	『The Internet Key Exchange (IKE)』

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトには、数千ページに及ぶ検索可能な技術情報があります。製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクもあります。 Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	http://www.cisco.com/en/US/support/index.html

第 4 章

キーチェーン管理の実装

このモジュールでは、キーチェーン管理の実装方法について説明します。キーチェーン管理は、相互に信頼を確立する前に、キーなどの秘密を交換するすべてのエンティティに共有秘密を設定する、認証の一般的な方法です。 Cisco IOS XR ソフトウェアのルーティング プロトコルおよびネットワーク管理アプリケーションでは、ピアとの通信中におけるセキュリティ向上のために、認証が頻繁に使用されます。

キーチェーン管理の実装の機能履歴

リリース	変更点
リリース 3.7.2	この機能を追加しました。

キーチェーン管理の設定に関する前提条件

適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。

キーチェーン管理の実装に関する制約事項

システム クロックを変更すると、既存の設定におけるキーの有効性に影響を与えることに注意してください。

キーチェーン管理の実装について

キーチェーン自体は関連性を持っていません。このため、キー（認証用）を使用してピアと通信する必要があるアプリケーションで使用される必要があります。キーチェーンは、ライフタイムに基づいてキーとロールオーバーを処理する、セキュリティの高いメカニズムを提供します。ボーダー ゲートウェイ プロトコル（BGP）、Open Shortest Path First（OSPF）、および Intermediate System-to-Intermediate System（IS-IS）では、キーチェーンを使用して認証用のヒットレス キー ロールオーバーを実装します。BGP は TCP 認証を使用します。この認証では、認証オプションを有効にし、キーチェーン用に設定された暗号化アルゴリズムに基づいたメッセージ認証コード（MAC）を送信します。BGP、OSPF、および IS-IS のキーチェーン設定の詳細については、を参照してください。

- リソース予約プロトコル（RSVP）は、認証にキーチェーンを使用します。 RSVP の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router MPLS Configuration Guide*』を参照してください。
- IP サービス レベル契約（IP SLA）は、IP SLA 制御メッセージの MD5 認証にキーチェーンを使用します。 IP SLA の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Monitoring Configuration Guide*』を参照してください。また、**key-chain** コマンドの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Monitoring Comand Reference*』を参照してください。

キーチェーン管理を実装するには、次の項で説明されているキーのライフタイムの概念を理解しておく必要があります。

キーのライフタイム

セキュリティ方式としてキーを使用する場合は、キーのライフタイムを指定して、期限が切れた際には定期的にキーを変更する必要があります。 安定性を維持するには、各パーティがアプリケーションのキーを複数保存して同時に使用できるようにする必要があります。 キーチェーンは、同じピア、ピアのグループ、またはその両方を認証するために一括管理されている一連のキーです。

キーチェーン管理では、一連のキーをキーチェーンの下にまとめてグループ化し、キーチェーン内の各キーをライフタイムに関連付けます。

（注） ライフタイムが設定されていないキーはすべて無効と見なされるため、キーは設定中に拒否されます。

キーのライフタイムは、次のオプションによって定義されます。

- Start-time：絶対時間を指定します。
- End-time：開始時間に対応する絶対時間を指定するか、無期限を指定します。

キーチェーン内のそれぞれのキーの定義では、キーが有効な期間（ライフタイムなど）を指定する必要があります。 指定したキーのライフタイム期間中は、この有効なキーとともにルーティング更新パケットが送信されます。 キーが有効ではない期間はキーを使用できません。 このため、指定したキーチェーンでは、キーの有効期間を重複させて、有効なキーの不在期間をなくすことを推奨します。 有効なキーの不在期間が発生した場合、ネイバー認証は行われず、ルーティング更新は失敗します。

複数のキーチェーンを指定できます。

キーチェーン管理の実装方法

この項では、次の手順について説明します。

キーチェーンの設定

この作業では、キーチェーンの名前を設定します。

キーチェーンの名前を作成または変更できます。

手順の概要

1. **configure**
2. **key chain** *key-chain-name*
3. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
4. **show key chain** *key-chain-name*

手順の詳細

	コマンドまたはアクション	目的
ステップ1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ2	**key chain** *key-chain-name*
	
	例：
	`RP/0/RSP0/CPU0:router(config)# key chain isis-keys`
	`RP/0/RSP0/CPU0:router(config-isis-keys)#`
		キーチェーンの名前を作成します。
		（注） キーのIDを設定せずにキーチェーン名のみを設定しても、操作は無効と見なされます。 設定を終了しても、キーのIDと1つ以上のグローバルコンフィギュレーションモードの属性またはkeychain-keyコンフィギュレーション モードの属性（ライフタイムやキー文字列など）を設定するまでは、変更のコミットは要求されません。

ステップ3	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end**コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yesと入力すると、実行コンフィギュレーションファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータがEXECモードに戻ります。
		◦ noと入力すると、コンフィギュレーションセッションが終了して、ルータがEXECモードに戻ります。変更はコミットされません。
		◦ cancelと入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 4	**show key chain** *key-chain-name* 例： `RP/0/RSP0/CPU0:router# show key chain isis-keys`	（任意）キーチェーン名を表示します。 （注） *key-chain-name* 引数はオプションです。 *key-chain-name* 引数の名前を指定しない場合、すべてのキーチェーンが表示されます。

次の作業

キーチェーン設定が完了したら、キーを受け付ける許容値の設定，（135 ページ）の項を参照してください。

キーを受け付ける許容値の設定

この作業では、キーを受け付ける許容値を設定し、キーチェーンによるアプリケーション（ルーティングプロトコルや管理プロトコルなど）のヒットレスキーロールオーバーを容易にします。

手順の概要

1. **configure**
2. **key chain** *key-chain-name*
3. **accept-tolerance** *value* [**infinite**]
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**key chain** *key-chain-name* 例： `RP/0/RSP0/CPU0:router(config)# key chain isis-keys`	キーチェーンの名前を作成します。

	コマンドまたはアクション	目的
ステップ 3	**accept-tolerance** *value* [**infinite**] 例： `RP/0/RSP0/CPU0:router(config-isis-keys)# accept-tolerance infinite`	キーチェーンのキーを受け入れる際の許容値を設定します。 • *value* 引数を使用して、許容値の範囲を秒数で設定します。 範囲は、1 ～ 8640000 です。 • **infinite** キーワードを使用して、許容値が無期限であることを指定します。

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

キーチェーンのキー ID の設定

この作業では、キーチェーンのキー ID を設定します。

キーチェーンのキーを作成または変更できます。

手順の概要

1. **configure**
2. **key chain** *key-chain-name*
3. **key** *key-id*
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**key chain** *key-chain-name* 例： `RP/0/RSP0/CPU0:router(config)# key chain isis-keys`	キーチェーンの名前を作成します。
ステップ 3	**key** *key-id* 例： `RP/0/RSP0/CPU0:router(config-isis-keys)# key 8`	キーチェーンのキーを作成します。 キー ID 番号は 10 進数から 16 進数に変換され、コマンド モード サブプロンプトが作成されます。 • *key-id* 引数は 48 ビット整数型として使用します。

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。

	コマンドまたはアクション	目的
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

次の作業

キーチェーンのキー ID を設定したら、キー文字列のテキストの設定, （138 ページ）の項を参照してください。

キー文字列のテキストの設定

この作業では、キー文字列のテキストを設定します。

手順の概要

1. **configure**
2. **key chain** *key-chain-name*
3. **key** *key-id*
4. **key-string** [**clear** | **password**] *key-string-text*
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**key chain** *key-chain-name* 例： `RP/0/RSP0/CPU0:router(config)# key chain isis-keys`	キーチェーンの名前を作成します。

ステップ 3	**key** *key-id*
	
	例：
	`RP/0/RSP0/CPU0:router(config-isis-keys)# key 8`
	`RP/0/RSP0/CPU0:router(config-isis-keys-0x8)#`
		キーチェーンのキーを作成します。

ステップ 4	**key-string** [**clear** | **password**] *key-string-text* 例： `RP/0/RSP0/CPU0:router(config-isis-keys-0x8)# key-string password 8`	キーのテキスト文字列を指定します。 • クリア テキスト形式でキー文字列を指定するには **clear** キーワードを使用します。暗号化形式でキーを指定するには **password** キーワードを使用します。 • 文字列を有効なパスワードにするには、次の規則に従う必要があります。 ◦ 偶数個の文字が含まれている。 ◦ 最小文字数は 4 文字である。 ◦ 最初の 2 桁は 10 進数、残りの桁は 16 進数である。 ◦ 最初の 2 桁は 53 以下である。 有効なパスワードの例は、次のとおりです。 ◦ 12abcd ◦ 32986510 •
ステップ 5	次のいずれかのコマンドを使用します。 • **end** • **commit** 例： `RP/0/RSP0/CPU0:router(config)# end`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` ◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレー

	コマンドまたはアクション	目的
	または `RP/0/RSP0/CPU0:router(config)# commit`	ション セッションが終了して、ルータが EXEC モードに戻ります。 ◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 ◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

次の作業

キー文字列のテキストを設定したら、アウトバウンド アプリケーションのトラフィックの認証ダイジェストを生成するキーの設定, （142 ページ）の項を参照してください。

有効なキーの確認

この作業では、ローカル アプリケーションがリモート ピアを認証するための有効なキーを決定します。

手順の概要

1. **configure**
2. **key chain** *key-chain-name*
3. **key** *key-id*
4. **accept-lifetime** *start-time* [**duration** *duration-value* | **infinite** | *end-time*]
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**key chain** *key-chain-name* 例： `RP/0/RSP0/CPU0:router(config)# key chain isis-keys`	キーチェーンの名前を作成します。

ステップ 3	**key** *key-id*
	
	例：
	`RP/0/RSP0/CPU0:router(config-isis-keys)# key 8`
	`RP/0/RSP0/CPU0:router(config-isis-keys-0x8)#`
		キーチェーンのキーを作成します。

ステップ 4	**accept-lifetime** *start-time* [**duration** *duration-value* | **infinite** | *end-time*] 例： `RP/0/RSP0/CPU0:router(config-isis-keys-0x8)# accept-lifetime 1:00:00 october 24 2005 infinite`	（任意）クロック タイムの観点から、キーのライフタイムの有効性を指定します。

ステップ 5	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュ

	コマンドまたはアクション	目的
		レーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

アウトバウンドアプリケーションのトラフィックの認証ダイジェストを生成するキーの設定

この作業では、アウトバウンド アプリケーションのトラフィックの認証ダイジェストを生成するキーを設定します。

手順の概要

1. **configure**
2. **key chain** *key-chain-name*
3. **key** *key-id*
4. **send-lifetime** *start-time* [**duration** *duration-value* | **infinite** | *end-time*]
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**key chain** *key-chain-name* 例： `RP/0/RSP0/CPU0:router(config)# key chain isis-keys`	キーチェーンの名前を作成します。

	コマンドまたはアクション	目的

ステップ 3	**key** *key-id*
	
	例：
	`RP/0/RSP0/CPU0:router(config-isis-keys)# key 8`
	`RP/0/RSP0/CPU0:router(config-isis-keys-0x8)#`
		キーチェーンのキーを作成します。

ステップ 4	**send-lifetime** *start-time* [**duration** *duration-value* | **infinite** | *end-time*] 例： `RP/0/RSP0/CPU0:router(config-isis-keys-0x8)# send-lifetime 1:00:00 october 24 2005 infinite`	（任意）キーチェーンの認証キーが有効に送信される設定期間を指定します。 クロック タイムの観点から、キーのライフタイムの有効性を指定できます。 さらに、start-time値と次のいずれかの値を指定できます。 • **duration** キーワード（秒） • **infinite** キーワード • *end-time* 引数 キーのライフタイムを設定する場合は、ネットワーク タイム プロトコル（NTP）または他の時刻同期方式を推奨します。

ステップ 5	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータがEXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

	コマンドまたはアクション	目的

暗号化アルゴリズムの設定

この作業では、暗号化アルゴリズムを選択してキーチェーン設定に反映できます。

手順の概要

1. **configure**
2. **key chain** *key-chain-name*
3. **key** *key-id*
4. **cryptographic-algorithm** [**HMAC-MD5** | **HMAC-SHA1-12** | **HMAC-SHA1-20** | **MD5** | **SHA-1**]
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ 2	**key chain** *key-chain-name*
	
	例：
	`RP/0/RSP0/CPU0:router(config)# key chain isis-keys`
	`RP/0/RSP0/CPU0:router(config-isis-keys)#`
		キーチェーンの名前を作成します。

ステップ 3	**key** *key-id*
	
	例：
	`RP/0/RSP0/CPU0:router(config-isis-keys)# key 8`
	`RP/0/RSP0/CPU0:router(config-isis-keys-0x8)#`
		キーチェーンのキーを作成します。

	コマンドまたはアクション	目的
ステップ 4	**cryptographic-algorithm** [**HMAC-MD5** | **HMAC-SHA1-12** | **HMAC-SHA1-20** | **MD5** | **SHA-1**] 例： `RP/0/RSP0/CPU0:router(config-isis-keys-0x8)# cryptographic-algorithm MD5`	暗号化アルゴリズムを選択します。 次のアルゴリズムのリストから選択できます。 • HMAC-MD5 • HMAC-SHA1-12 • HMAC-SHA1-20 • MD5 • SHA-1 ルーティング プロトコルは、それぞれ異なる暗号化アルゴリズムのセットをサポートしています。 • ボーダー ゲートウェイ プロトコル（BGP）は HMAC-MD5 と HMAC-SHA1-12 だけをサポート • Intermediate System-to-Intermediate System（IS-IS）は HMAC-MD5 だけをサポート • Open Shortest Path First（OSPF）は MD5 と HMAC-MD5 だけをサポート

ステップ 5	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。

	コマンドまたはアクション	目的
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

キーチェーン管理の実装の設定例

この項では、次の設定例について説明します。

キーチェーン管理の設定：例

次に、キーチェーン管理を設定する例を示します。

```
configure
key chain isis-keys
accept-tolerance infinite
key 8
key-string mykey91abcd
cryptographic-algorithm MD5
send-lifetime 1:00:00 june 29 2006 infinite
accept-lifetime 1:00:00 june 29 2006 infinite
end

Uncommitted changes found, commit them? [yes]: yes

show key chain isis-keys

Key-chain: isis-keys/ -

accept-tolerance -- infinite
Key 8 -- text "1104000E120B520005282820"
  cryptographic-algorithm -- MD5
  Send lifetime:   01:00:00, 29 Jun 2006 - Always valid  [Valid now]
  Accept lifetime: 01:00:00, 29 Jun 2006 - Always valid [Valid now]
```

参考資料

ここでは、キーチェーン管理の実装に関連する参考資料について説明します。

関連資料

関連項目	ドキュメント名
キーチェーン管理のコマンド：コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』のキーチェーン管理コマンド

標準

標準	タイトル
この機能でサポートが追加または変更された標準はありません。また、この機能で変更された既存の標準のサポートはありません。	—

MIB

MIB	MIB リンク
—	Cisco IOS XR ソフトウェアを使用して MIB を検出およびダウンロードするには、URL（http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml）にある Cisco MIB Locator を使用して、[Cisco Access Products] メニューでプラットフォームを選択します。

RFC

RFC	タイトル
この機能でサポートが追加または変更された RFC はありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトには、数千ページに及ぶ検索可能な技術情報があります。製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクもあります。 Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	http://www.cisco.com/en/US/support/index.html

第 5 章

合法的傍受の実装

合法的傍受は司法命令や行政命令によって認可され、司法当局が回線通信およびパケットモード通信に対して電子機器を用いた情報収集を実施するプロセスです。 世界中のサービス プロバイダーは、司法当局の回線交換およびパケットモード ネットワークにおける電子機器を用いた情報収集の実施をサポートすることが法的に求められます。

認可されたサービス プロバイダーの担当者のみが、法的に認可された傍受命令を処理および設定することを許可されています。 ネットワーク管理者および技術者は、法的に認可された傍受命令、または進行中の傍受に関する知識を得ることを禁止されています。 ルータにインストールされている傍受に関するエラー メッセージまたはプログラム メッセージは、コンソールには表示されません。

合法的傍受の実装の機能履歴

リリース	変更点
リリース 4.1.0	この機能を追加しました。
リリース 4.2.0	合法的傍受のハイ アベイラビリティ サポートが追加されました。 IPv6 の合法的傍受のサポートが追加されました。

• 参考資料，164 ページ

合法的傍受の実装に関する前提条件

適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。 ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。

合法的傍受の実装には、次の前提条件も満たす必要があります。

- Cisco ASR 9000 シリーズ アグリゲーション サービス ルータは、合法的傍受の運用においてコンテンツの傍受アクセス ポイント（IAP）ルータとして使用されます。
- **プロビジョニングされたルータ**：ルータはプロビジョニング済みである必要があります。 詳細については、『*Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide*』を参照してください。

ヒント 合法的傍受のタップには、ループバック インターフェイスをプロビジョニングすると、他のインターフェイス タイプに比べて利点があります。

- **Cisco IOS XR ソフトウェアの SNMP Server コマンドの理解**：合法的傍受を実現する基盤となる簡易ネットワーク管理プロトコル バージョン 3（SNMP v3）は、『*Cisco ASR 9000 Series Aggregation Services Router System Management Command Reference*』の「*SNMP Server Commands*」モジュールに説明されているコマンドを使用して設定されます。 合法的傍受を実装するには、SNMP サーバの機能を理解する必要があります。 このため、『*Cisco ASR 9000 Series Aggregation Services Router System Management Configuration Guide*』の「*Implementing SNMP*」モジュールに説明されている情報をよく確認してください。
- **合法的傍受が明示的にディセーブルになっていること**：プロビジョニングされたルータでは、合法的傍受は自動的にイネーブルになっています。 ただし、進行中のアクティブなタップがある場合、タップは削除されるため、LI をディセーブルにしないでください。
- **管理プレーンで SNMPv3 がイネーブルに設定されていること**：コマンドがルータのインターフェイス（できればループバック）に送信されるよう、管理プレーンが SNMP コマンドを受け付けられるようにします。 これにより、メディエーション デバイス（MD）が物理インターフェイスと通信できるようになります。
- **VACM ビューが SNMP サーバ向けにイネーブルになっていること**：ビューベース アクセス制御モデル（VACM）ビューは、ルータでイネーブルになっている必要があります。
- **プロビジョニングされた MD**：詳細については、ご使用の MD に関するベンダーのマニュアルを参照してください。 シスコが推奨する MD 機器サプライヤのリストについては、http://www.cisco.com/en/US/tech/tk583/tk799/tsd_technology_support_protocol_home.html を参照してください。
- **VoIP 監視固有の要件**

- **合法的傍受がイネーブルになっているコール エージェント**：合法的傍受がイネーブルになっているコール エージェントでは、監視ターゲットが MD にシグナリング情報を提供できるように、MD との通信用インターフェイスをサポートしている必要があります。 MD は、監視ターゲットのセッション記述プロトコル（SDP）のシグナリング情報から、送信元 IP アドレス、宛先 IP アドレス、Real-Time Protocol（RTP）のポート番号を抽出します。 これらの情報を使用して SNMPv3 SET を作成します。SNMPv3 SET はコンテンツ IAP として動作しているルータに送信され、監視ターゲットの傍受を実現します。

  MD は CISCO-TAP2-MIB を使用して、コンテンツ IAP として動作しているルータと MD との間の通信をセットアップします。

  MD は CISCO-IP-TAP-MIB を使用して、SDP から傍受および取得する IP アドレスとポート番号のフィルタをセットアップします。

- ターゲット番号によるコールで使用されるルータは、この目的のために MD を通じてプロビジョニングされる必要があります。

- 傍受するターゲット番号がプロビジョニングされている MD。

• データ セッション監視固有の要件

- データ ターゲットによって使用される、この目的のために MD を通じてプロビジョニングされているルータ。

- **ユーザ ログイン ID、ユーザの CPE デバイスの MAC アドレス、または DSLAM の物理位置 ID がプロビジョニングされている MD**：IP アドレスは、ネットワーク内のターゲットの特定に非常に頻繁に使用されるバインディングになります。 ただし、一部のネットワーク アーキテクチャでは、ネットワーク内のターゲットを独自に特定する別の情報形式が使用されている場合があります。 このような情報形式には、MAC アドレスと acct-session-id が含まれています。

• MD はネットワーク内の任意の場所に配置できますが、ターゲットの傍受に使用されているコンテンツ IAP ルータから到達可能である必要があります。 MD はグローバル ルーティング テーブルからのみ到達可能で、VRF ルーティング テーブルからは到達不可である必要があります。

合法的傍受の実装に関する制約事項

合法的傍受は、Cisco ASR 9000 Series Router では次の機能をサポートしていません。

• IPv6 マルチキャスト タッピング

• IPv4 マルチキャスト タッピング

• タップ別ドロップ カウンタ

• ギガビット イーサネット LC における IPv6 の傍受

• IPv6 MD カプセル化

- インターフェイス別タッピング
- 1 つのタップの複数 MD への複製
- タグ パケットのタッピング
- L2 フローのタッピング
- RTP のカプセル化
- 複製デバイスの暗号化および整合性チェック

（注） タップ別ドロップ カウンタのサポートは、ASR9000-SIP-700 ラインカードのみで利用できます。イーサネット ラインカードでは利用できません。

合法的傍受の実装について

シスコの合法的傍受は、サービス非依存傍受（SII）アーキテクチャと、SNMPv3 プロビジョニング アーキテクチャに基づいています。 SNMPv3 は、データの送信元を認証し、ルータから MD への接続がセキュアであることを保証する要件に対応します。 これにより、認可されていないパーティが傍受のターゲットを偽造できないようにします。

合法的傍受は、次の機能を提供します。

- SNMPv3 を使用した、MD からの Voice-over IP（VoIP）およびデータ セッション傍受のプロビジョニング
- 傍受された VoIP およびデータ セッション データの MD への配信
- SNMPv3 合法的傍受プロビジョニング インターフェイス
- 合法的傍受 MIB：CISCO-TAP2-MIB バージョン 2
- CISCO-IP-TAP-MIB は、IP 用のシスコの傍受機能を管理し、CISCO-TAP2-MIB とともに IP トラフィックの傍受に使用されます。
- ユーザ データグラム プロトコル（UDP）の MD へのカプセル化
- 傍受されたパケットの MD への複製および転送
- 受信パケットに設定された任意の規則に基づいた Voice-over IP（VoIP）コール傍受。
- LI がイネーブルになっているコール エージェントによる Voice-over IP（VoIP）の傍受
- IP アドレスに基づいたデータ セッションのコール傍受

VoIP コールのプロビジョニング

VoIP の合法的傍受のプロビジョニングは、次の方法で行われます。

- ユーザが SNMPv3 を通じて定義しているセキュリティと認証が実行されます。
- MD は SNMPv3 を使用して、合法的傍受情報のプロビジョニングを行います。
- ネットワーク管理は標準 MIB を通じて行われます。

コールの傍受

VoIP コールは、次の方法で傍受されます。

- MD はコンフィギュレーション コマンドを使用して、コール制御エンティティに傍受を設定します。
- コール制御エンティティは、ターゲットの傍受に関する情報を MD に送信します。
- MD は SNMPv3 を通じて、コンテンツ IAP ルータまたはトランク ゲートウェイにコール内容の傍受要求を開始します。
- コンテンツ IAP ルータまたはトランク ゲートウェイはコール内容を傍受し複製して、Packet Cable Electronic Surveillance UDP 形式で MD に送信します。 特に、IP ヘッダーの最初のバイトから始まる元のパケットには、TAP2-MIB において MD から提供される 4 バイトの CCCID がパケットの前に付与されます。 次に、このパケットは宛先アドレスおよびMDのポートとともに UDP フレームに入れられます。
- 複製された VoIP パケットが MD に送信されると、MD は一般的な規格でコピーを司法当局が所有する収集機能に転送します。

データ セッションのプロビジョニング

データ セッション用のプロビジョニングは、VoIP コールの合法的傍受の際と同様の方法で行われます。 （VoIP コールのプロビジョニング，（152 ページ）を参照してください）。

データの傍受

データは、次の方法で傍受されます。

- 合法的傍受がイネーブルになっている認証サーバまたはアカウンティング サーバが利用できない場合は、ネットワーク内でターゲットの存在を検出するためにスニファ デバイスを使用できます。
  - MD はコンフィギュレーション コマンドを使用して、スニファに傍受を設定します。
  - スニファ デバイスは、ターゲットの傍受に関する情報を MD に送信します。
- MD は SNMPv3 を使用して、コンテンツ IAP ルータに通信内容の傍受要求を開始します。
- コンテンツ IAP ルータは通信内容を傍受し複製して、UDP 形式で MD に送信します。

- 傍受されたデータ セッションは、サポートされている合法的傍受の提供規格を使用して、MD から司法当局の収集機能へ送信されます。

MD について

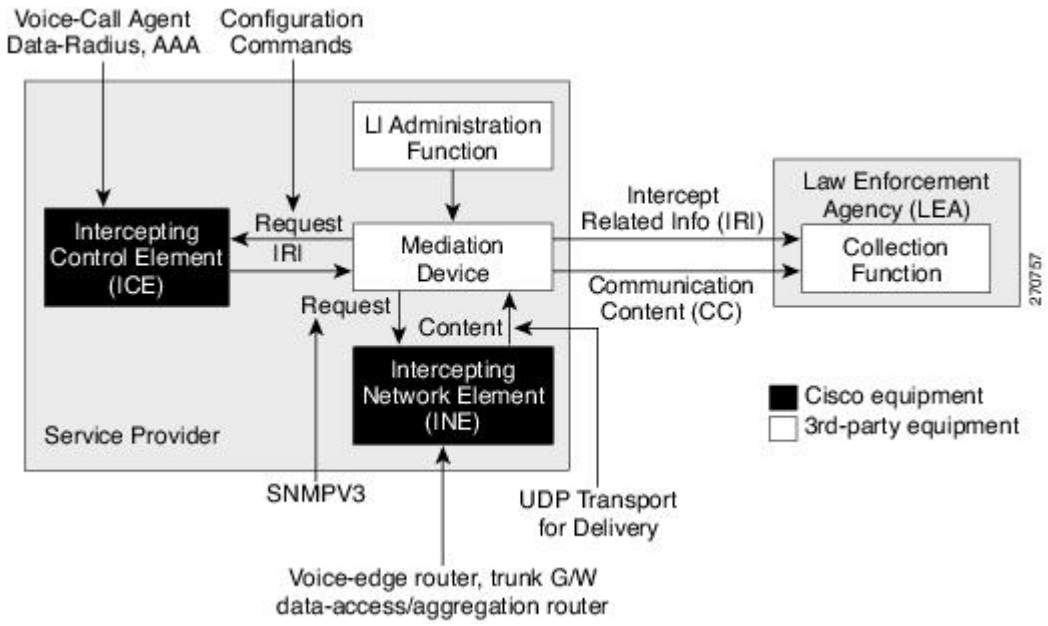
MD は次の作業を実行します。

- 認可された時間に傍受をアクティブにし、認可された期間が経過したときには傍受を削除する。
- 以下を確認するために、定期的にネットワーク内の要素を監査する。
  - 認可された傍受のみが存在していること。
  - 認可された傍受がすべて存在していること。

合法的傍受トポロジ

次の図は、音声傍受およびデータ傍受の合法的傍受トポロジにおける、傍受アクセス ポイントおよびインターフェイスを示しています。

図 *1*：音声傍受およびデータ傍受の合法的傍受トポロジ



スケールまたはパフォーマンスの改善

合法的傍受の拡張性およびパフォーマンスに関して、Cisco ASR 9000 Series Router に新たに導入された拡張機能は次のとおりです。

- IPv4の合法的傍受タップの上限は IPv4 ごとに 1000 タップ。
- IPv6の合法的傍受タップの上限は IPv6 ごとに 1000 タップ。
- 傍受レートは次のとおり。
  - ASR9000-SIP-700 ラインカードの場合、ネットワーク プロセッサ（NP）ごとに 50 Mbps。
  - ギガビット イーサネット ラインカードの場合、100 Mbps。
  - モジュラ Weapon-X ラインカードの場合、500 Mbps。
  - 100GE ラインカードの場合、1000 Mbps。
- 最大 512 個の MD をサポート。

IPv6 パケットの傍受

ここでは、Cisco ASR 9000 Series Router でサポートされている IPv6 パケットの傍受の詳細について説明します。

合法的傍受フィルタ

タップの分類に使用されるフィルタは次のとおりです。

- IP アドレス タイプ
- 宛先アドレス
- 宛先マスク
- 送信元アドレス
- 送信元マスク
- ToS（タイプ オブ サービス）および ToS マスク
- プロトコル
- 範囲指定の宛先ポート
- 範囲指定の送信元ポート
- VRF（VPN ルーティングおよび転送）
- フロー ID

フロー ID に基づいた IPv6 パケットの傍受

IPv6 パケットのフィルタ条件をさらに拡張するために、フロー ID に基づく IPv6 パケット傍受のサポートが Cisco ASR 9000 Series Router に追加されました。 すべての IPv6 パケットは、次の「IPv6

ヘッダーフィールドの詳細」表で定義されている数値フィールドを構成するIPv6ヘッダーのフィールドに基づいて傍受されます。

（注） フィールド長またはペイロード長はパケットの傍受には使用されません。

表 *5*：*IPv6* ヘッダー フィールドの詳細

IPv6 フィールド名	フィールドの説明	フィールド長
バージョン	IPv6 バージョン番号。	4 ビット
トラフィック クラス	インターネット トラフィックにおける配信の優先度を示す値。	8 ビット
フロー ID（フロー ラベル）	一連のパケットに対して、送信元から宛先までの特別なルータ処理を指定するために使用されます。	20 ビット
ペイロード長	パケット内のデータ長を指定します。ゼロにクリアすると、オプションはホップバイホップのジャンボペイロードになります。	16 ビット（未割り当て）
次ヘッダー	次のカプセル化されたプロトコルを指定します。値は、IPv4 プロトコル フィールドで指定されている値と互換性があります。	8 ビット
ホップ リミット	各ルータがパケットを転送するたびに、ホップ リミットは1ずつ減少します。ホップリミットフィールドがゼロに達すると、パケットは廃棄されます。このフィールドは、本来時間ベースのホップリミットとして使用されることを目的としていたIPv4ヘッダーの TTL フィールドに代わるものです。	8 ビット（符号なし）
送信元アドレス	送信ノードの IPv6 アドレス。	16 バイト
宛先アドレス	宛先ノードの IPv6 アドレス。	16 バイト

フロー ID またはフロー ラベルは、トラフィック フローの区別に使用される、IPv6 パケット ヘッダー内の 20 ビットのフィールドです。 各フローには、一意のフロー ID が含まれています。 特定のフロー ID に一致するパケットを傍受するフィルタ条件は、タップ設定ファイルに定義されます。 傍受されたマップ済みのフロー ID は、ラインカードから MD 設定ファイル内で指定されている次のホップに送信されます。 傍受されたパケットは複製され、ラインカードから MD に送信されます。

VRF（6VPE）および 6PE パケットの傍受

ここでは、VRF 対応パケットおよび 6PE パケットの傍受について説明します。 この傍受の仕組みを説明する前に、6VPE ネットワークの基本的な知識について説明します。

MPLS VPN モデルは真のピア VPN モデルです。 このモデルは、プロバイダーのコンテンツ IAP ルータで一意の VPN ルート転送（VRF）テーブルを各カスタマーの VPN に割り当てることで、トラフィックの分離を実行します。 そのため、VPN 内のユーザは外部のトラフィックを見ることができません。

Cisco ASR 9000 Series Router は、6VPE において、指定した VRF ID の IPv6 パケットの傍受をサポートしています。 VPN 上のトラフィックを区別するために、特定の VRF ID を含む VRF が定義されています。 特定の VRF ID をタップするフィルタ条件は、タップ内で指定されます。 IPv6 パケットは、インポジション（ip2mpls）およびディスポジション（mpls2ip）の両方のシナリオで、VRF コンテキストを使用して傍受されます。

6PE パケットは VPN 上で IPv6 パケットを伝送します。 パケットには VRF ID は含まれていません。 IP トラフィックのみが傍受されます。MPLS ベースの傍受はサポートされていません。 IPv6 トラフィックは、インポジション（ip2mpls）およびディスポジション（mpls2ip）の MPLS クラウドのコンテンツ IAP で傍受されます。

ip2tag パケットおよび tag2ip パケットに対しても、IPv6 パケットの傍受が実行されます。 ip2tag パケットは、プロバイダーのコンテンツ IAP ルータで IPv6 からタギングに変換されたパケット（IPv6 to MPLS）を指し、tag2ip パケットは、プロバイダーのコンテンツ IAP ルータでタギングから IPv6 に変換されたパケット（MPLS to IPv6）を指します。

傍受パケットでサポートされるカプセル化タイプ

タップをマッピングする傍受パケットは複製およびカプセル化され、MD に送信されます。 IPv4 パケットおよび IPv6 パケットは、UDP（ユーザ データグラム プロトコル）カプセル化を使用してカプセル化されます。 複製されたパケットは、コンテンツ配信プロトコルに UDP を使用して、MD に転送されます。 IPv4 MD カプセル化のみサポートされています。

傍受パケットには、新しい UDP ヘッダーと IPv4 ヘッダーが付与されます。 IPv4 ヘッダーの情報は MD 設定から取得されます。 IP ヘッダーおよび UDP ヘッダーとは別に、4 バイトのチャネル ID（CCCID）もパケットの UDP ヘッダーの後に挿入されます。 MD カプセル化を追加した後、パケット サイズが MTU を超過する場合、出力 LC CPU はパケットをフラグメント化します。 また、タップされたパケットがすべてにフラグメントである場合もあります。 各タップには、MD が 1 つだけ関連付けられています。 Cisco ASR 9000 Series Router は、複数 MD への複製パケットの転送をサポートしていません。

（注） RTP や RTP-NOR などのカプセル化タイプはサポートされていません。

タップ別ドロップ カウンタのサポート

Cisco ASR 9000 Series Router ラインカードでは、インターフェイスとして SNMP サーバを提供し、MD パケットに転送された各タップとドロップ数をエクスポートします。 ポリサー処理により MD に転送される前にドロップされた傍受パケットは、すべてカウントおよびレポートされます。 ポリサー処理によりドロップされるパケットは、ドロップされるパケットの中で唯一タップ別ドロップ カウンタでカウントされます。 合法的傍受フィルタが変更された場合、パケット カウントは 0 にリセットされます。

（注） タップ別ドロップ カウンタのサポートは、ASR9000-SIP-700 ラインカードのみで利用できます。イーサネット ラインカードでは利用できません。

合法的傍受のハイ アベイラビリティ

合法的傍受のハイ アベイラビリティでは、タップ フローおよびプロビジョニングされた MD テーブルの継続的な運用を実現し、ルート プロセッサ フェールオーバー（RPFO）による情報の喪失を低減します。

ストリームの継続的な傍受を実現するには、RP フェールオーバーが検出された際に、MD が CISCO-TAP2-MIB、CISCO-IP-TAP-MIB、および CISCO-USER-CONNECTION-TAP-MIB に関連するすべての行を再プロビジョニングし、RP および MD にまたがるデータベース ビューを同期する必要があります。

（注） 合法的傍受のハイ アベイラビリティは、リリース 4.2.0 以降ではデフォルトでイネーブルになっています。

RP フェールオーバー中のタップおよび MD テーブルの維持

MD はあらゆるタイミングで SNMP 設定プロセスを通じて、タップの喪失を検出する役割を果たします。

RPFO が完了すると、MD はストリーム テーブルのすべてのエントリ、MD テーブル、および IP タップにフェールオーバー前と同じ値を再プロビジョニングする必要があります。 エントリが時間どおりに再プロビジョニングされる限り、既存のタップは喪失されずにフローを継続します。

citapStreamEntry、cTap2StreamEntry、cTap2MediationEntry MIB オブジェクトでの SNMP 操作の動作に関連して、MD テーブルおよびタップ テーブルの再プロビジョニングには次の制約事項があります。

- RPFO 後に、再プロビジョニングされていないテーブルの行は SNMP GET 操作の結果として NO_SUCH_INSTANCE 値を返します。

- テーブルの行全体が RPFO 前と完全に同じ値で、かつ rowStatus を CreateAndGo にして、1 回の設定ステップで作成される必要があります。 cTap2MediationTimeout オブジェクトのみは例外で、有効な未来時刻を反映する必要があります。

リプレイ タイマー

リプレイ タイマーは、MD が既存のタップ フローを維持しながらタップ エントリを再プロビジョニングするための十分な時間を確保する内部タイムアウトです。 RPFO が実行されると、このタイマーはアクティブな RP でリセットされ、開始されます。 リプレイ タイマーは、ルータ内の LI エントリ数の係数で、最小値は 10 分です。

リプレイ タイムアウト後、再プロビジョニングされていないタップでは傍受が停止します。

（注） ハイ アベイラビリティが必須でない場合、MD はフェールオーバー後にエントリがエージング アウトするまで待機します。 MD はリプレイ タイマーが満了するまでエントリを変更できません。 MD でタップをそのまま再インストールしてその後に変更を加えるか、エントリがエージング アウトするまで MD を待機させることができます。

ルータでの合法的傍受の SNMP v3 アクセスの設定方法

合法的傍受をイネーブルにする目的で管理プレーン保護（MPP）および SNMP を設定するには、次の手順を示されている順番で実行します。

合法的傍受のディセーブル化

合法的傍受は、この機能がサポートされているルータでは、デフォルトでイネーブルになっています。

- LI をディセーブルにするには、グローバル コンフィギュレーション モードで **lawful-intercept disable** コマンドを入力します。
- この機能を再度イネーブルにするには、このコマンドの **no** 形式を使用します。

（注） プロビジョニングされているアクティブなタップや MD が存在する場合は、LI をディセーブルにしないでください。 ディセーブルにした場合、ルータからすべてのタップと MD が削除されます。

インバンド管理プレーン保護機能の設定

以前にMPPを別のプロトコルと連携して動作するように設定していない場合は、合法的傍受の目的でMDと通信できるように、MPP機能を設定してSNMPサーバをイネーブルにする必要はありません。 このような場合だけ、明示的にMPPをインバンドインターフェイスとして設定し、指定したインターフェイスまたはすべてのインターフェイスを使用してSNMPコマンドをルータで受け付けられるようにする必要があります。

（注） 最近Cisco IOSからCisco IOS XRソフトウェアに移行し、任意のプロトコルに対してMPPを設定済みである場合は、この作業を実行する必要があります。

合法的傍受の目的で、ループバックインターフェイスをSNMPメッセージの宛先にする場合が多くあります。 このインターフェイス タイプを選択した場合は、インバンド管理設定にこのインターフェイス タイプを含める必要があります。

設定手順については、インバンドインターフェイスの管理プレーン保護のデバイスの設定, （171ページ）の項を参照してください。 この手順のLIに関する例については、インバンド管理プレーン保護機能の設定：例, （163ページ）を参照してください。

インバンド管理インターフェイスの詳細な説明については、インバンド管理インターフェイス, （168ページ）を参照してください。

VoIPおよびデータ セッションを傍受するためのメディエーション デバイスのイネーブル化

次のSNMPサーバ設定作業では、MDによるVoIPまたはデータ セッションの傍受を許可することで、Cisco IOS XRソフトウェアを実行しているルータ上でCisco SII機能をイネーブルにします。

手順の概要

1. **configure**
2. **snmp-server view** *view-name* **ciscoTap2MIB included**
3. **snmp-server view** *view-name* **ciscoIpTapMIB included**
4. **snmp-server group** *group-name* **v3 auth read** *view-name* **write** *view-name* **notify** *view-name*
5. **snmp-server host** *ip-address* **traps version 3 priv** *username* **udp-port** *port-number*
6. **snmp-server user** *mduser-id groupname* **v3 auth md5** md-*password*
7. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
8. show snmp users
9. show snmp group
10. show snmp view

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**snmp-server view** *view-name* **ciscoTap2MIB included** 例： `RP/0//CPU0:router(config)# snmp-server view TapName ciscoTap2MIB included`	ビュー レコードを作成または変更し、CISCO-TAP2-MIB ファミリを含めます。
ステップ 3	**snmp-server view** *view-name* **ciscoIpTapMIB included** 例： `RP/0//CPU0:router(config)# snmp-server view TapName ciscoIpTapMIB included`	ビュー レコードを作成または変更し、CISCO-IP-TAP-MIB ファミリを含めます。
ステップ 4	**snmp-server group** *group-name* **v3 auth read** *view-name* **write** *view-name* **notify** *view-name*	新しい SNMP グループの設定、または SNMP ユーザを SNMP ビューにマップするテーブルの設定を行います。このグループは SNMP ビューの読み取り、書き込み、および通知権限を持っています。

	コマンドまたはアクション	目的
	例： `RP/0//CPU0:router(config)# snmp-server group TapGroup v3 auth read TapView write TapView notify TapView`	
ステップ 5	**snmp-server host** *ip-address* **traps version 3 priv** *username* **udp-port** *port-number* 例： `RP/0//CPU0:router(config)# snmp-server host 223.255.254.224 traps version 3 priv bgreen udp-port 2555`	SNMP トラップ通知、使用する SNMP のバージョン、通知のセキュリティ レベル、および通知の受信者（ホスト）を指定します。
ステップ 6	**snmp-server user** *mduser-id groupname* **v3 auth md5** md-*password* 例： `RP/0//CPU0:router(config)# snmp-server mduser-id TapGroup v3 auth md5 mdpassword`	MD パスワードと関連付ける v3 セキュリティ モデルと HMAC MD5 アルゴリズムを使用して、MD ユーザが SNMP グループに属するように設定します。 • *mduser-id* および *mdpassword* は MD に設定されている値と一致している必要があります。 あるいは、これらの値はルータで使用されている値と一致している必要があります。 • SNMPv3 セキュリティの最低基準を満たすには、パスワードの長さは 8 文字以上である必要があります。 • LI を利用する最低限のセキュリティ レベルは *auth* です。*noauth* では動作しません。 LI のセキュリティ レベルは MD のセキュリティ レベルとも一致している必要があります。 • ルータでは MD5 以外を選ぶこともできますが、MD 値は一致している必要があります。 ほとんどの MD では、MD5 がデフォルトになっているか、MD 5 のみをサポートしています。
ステップ 7	次のいずれかのコマンドを使用します。 • **end** • **commit** 例： `RP/0/RSP0/CPU0:router(config)# end`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` ◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッ

	コマンドまたはアクション	目的
	または `RP/0/RSP0/CPU0:router(config)# commit`	ションが終了して、ルータが EXEC モードに戻ります。 ◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 ◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。
ステップ 8	show snmp users 例： `RP/0//CPU0:router# show snmp users`	SNMP ユーザ テーブルの各 SNMP ユーザ名に関する情報を表示します。
ステップ 9	show snmp group 例： `RP/0//CPU0:router# show snmp group`	ネットワークの各 SNMP グループの情報を表示します。
ステップ 10	show snmp view 例： `RP/0//CPU0:router# show snmp view`	関連付けられた MIB ビュー ファミリ名、ストレージ タイプ、ステータスなど、設定されたビューに関する情報を表示します。

インバンド管理プレーン機能のイネーブル化の設定例

次に、デフォルトでディセーブルになっているMPP機能を合法的傍受の目的でイネーブルにする方法の例を説明します。

インバンド管理プレーン保護機能の設定：例

次の手順を使用して、管理アクティビティをグローバルまたはインバンド ポート単位で明示的にイネーブルにする必要があります。 インバンド MPP をグローバルにイネーブルにするには、

interface コマンドで特定のインターフェイス タイプとインスタンス ID を使用するのではなく、**all** キーワードを使用します。

```
RP/0//CPU0:router# configure
RP/0//CPU0:router(config)# control-plane
RP/0//CPU0:router(config-ctrl)# management-plane
RP/0//CPU0:router(config-mpp)# inband
RP/0//CPU0:router(config-mpp-inband)# interface loopback0
RP/0//CPU0:router(config-mpp-inband-Loopback0)# allow snmp
RP/0//CPU0:router(config-mpp-inband-Loopback0)# commit
RP/0//CPU0:router(config-mpp-inband-Loopback0)# exit
RP/0//CPU0:router(config-mpp-inband)# exit
RP/0//CPU0:router(config-mpp)# exit
RP/0//CPU0:router(config-ctr)# exit
RP/0//CPU0:router(config)# exit
RP/0//CPU0:router# show mgmt-plane inband interface loopback0

Management Plane Protection - inband interface


interface - Loopback0
        snmp configured -
                All peers allowed
RP/0//CPU0:router(config)# commit
```

参考資料

ここでは、合法的傍受の実装に関連する参考資料について説明します。

関連資料

関連項目	ドキュメント名
合法的傍受コマンド	『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』
SNMP の実装	『*Cisco ASR 9000 Series Aggregation Services Router System Management Configuration Guide*』
SNMP サーバ コマンド	『*Cisco ASR 9000 Series Aggregation Services Router System Management Command Reference*』

標準

標準	タイトル
サードパーティ機器と容易に通信してサービス プロバイダーの合法的傍受の要件を満たすシンプルな実装を目的に設計されたモジュール式のオープン アーキテクチャ。	RFC, （165 ページ）の RFC-3924 を参照してください。

標準	タイトル
ネットワーク デバイス間での管理情報の交換を容易にするアプリケーション層プロトコル。伝送制御プロトコル/インターネット プロトコル（TCP/IP）プロトコル スイートの一部。	『Simple Network Management Protocol Version 3 (SNMPv3)』

MIB

MIB	MIB リンク
• CISCO-TAP2-MIB バージョン 2 • CISCO-IP-TAP-MIB	Cisco IOS XR ソフトウェアを使用して MIB を検出およびダウンロードするには、URL（http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml）にある Cisco MIB Locator を使用して、[Cisco Access Products] メニューでプラットフォームを選択します。

RFC

RFC	タイトル
RFC-3924	『Cisco Architecture for Lawful Intercept in IP Networks』

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトには、数千ページに及ぶ検索可能な技術情報があります。製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクもあります。 Cisco.com に登録済みのユーザは、このページからログインして詳細情報にアクセスできます。	http://www.cisco.com/en/US/support/index.html

第 6 章

管理プレーン保護の実装

Cisco IOS XR ソフトウェア の管理プレーン保護（MPP）機能では、ネットワーク管理パケットのデバイスへの着信を許可するインターフェイスを制限できます。 ネットワーク オペレータは MPP 機能を使用して、1 つ以上のルータ インターフェイスを管理インターフェイスとして指定できます。

デバイス管理トラフィックは、これらの管理インターフェイスを通じてのみ着信が許可されます。 MPP をイネーブルにすると、指定された管理インターフェイス以外のインターフェイスでは、そのデバイス宛のネットワーク管理トラフィックは許可されません。 指定されたインターフェイスに管理パケットを制限することで、デバイスの管理方法をより詳細に制御できるため、デバイスのセキュリティが向上します。

このモジュールでは、Cisco ASR 9000 Series Routers での管理プレーン保護の実装方法について説明します。

MPP コマンドの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』の「*Management Plane Protection Commands on* Cisco ASR 9000 シリーズ ルータ」モジュールを参照してください。

管理プレーン保護の実装の機能履歴

リリース	変更点
リリース 3.7.2	この機能を追加しました。

管理プレーン保護の実装に関する前提条件

適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。

管理プレーン保護の実装に関する制約事項

管理プレーン保護（MPP）の実装には次の制約事項があります。

- 現在、MPP は拒否またはドロップされたプロトコル要求を追跡していません。
- MPP 設定では、プロトコル サービスをイネーブルにはできません。 MPP はさまざまなインターフェイスでサービスを利用可能にする役割のみを果たします。プロトコルは明示的にイネーブル化されます。
- インバンドインターフェイスで受信する管理要求は、その場で必ずしも認知されるわけではありません。
- ルータ プロセッサ（RP）と分散ルート プロセッサ（DRP）のイーサネット インターフェイスは、デフォルトでアウトオブバンド インターフェイスとなり、MPP で設定できます。
- MPP設定に加えた変更は、その変更よりも前に確立されているアクティブなセッションには影響を与えません。
- 現在、MPP は、TFTP、Telnet、簡易ネットワーク管理プロトコル（SNMP）、セキュア シェル（SSH）、HTTP などのプロトコルに対して着信する管理要求のみを制御します。
- MIB はサポートされていません。

管理プレーン保護の実装について

管理プレーン保護機能をイネーブルにする前に、次の概念について理解しておく必要があります。

インバンド管理インターフェイス

インバンド管理インターフェイスは、データ転送パケットだけでなく管理パケットも処理する、Cisco IOS XR ソフトウェア の物理インターフェイスまたは論理インターフェイスです。 インバンド管理インターフェイスは、共有管理インターフェイスとも呼ばれています。

アウトオブバンド管理インターフェイス

アウトオブバンドは、管理プロトコル トラフィックの転送または処理だけを許可するインターフェイスを意味します。アウトオブバンド管理インターフェイスは、ネットワーク管理トラフィックだけを受信するようネットワーク オペレータによって定義されます。 これには、転送（またはカスタマー）トラフィックによってルータの管理が妨害されないという利点があります。これにより、サービス拒否攻撃を受ける可能性は大幅に低下します。

アウトオブバンドインターフェイスは、アウトオブバンドインターフェイス間のトラフィックのみを転送するか、ルータ宛の管理パケットを終端します。また、アウトオブバンドインターフェイスをダイナミック ルーティング プロトコルに加えることができます。 サービス プロバイダーはルータのアウトオブバンド インターフェイスに接続し、ルータが提供可能なすべてのルーティング ツールおよびポリシー ツールを使用して、独立したオーバーレイ管理ネットワークを構築します。

インターフェイス上のピア フィルタリング

ピア フィルタリング オプションでは、特定のピアまたはピア範囲からの管理トラフィックの設定を許可します。

コントロール プレーン保護の概要

コントロール プレーンは、ルート プロセッサ上でプロセス レベルで動作し、Cisco IOS XR ソフトウェアのほとんどの機能に対して高レベルの制御を一括提供するプロセスの集合です。直接または間接的にルータが宛先となるすべてのトラフィックは、コントロール プレーンによって処理されます。 管理プレーン保護はコントロール プレーン インフラストラクチャ内で動作します。

管理プレーン

管理プレーンは、ルーティング プラットフォームの管理に関連するすべてのトラフィックの論理パスです。 レイヤおよびプレーン内で構造化されている通信アーキテクチャの 3 つのプレーンのうちの 1 つである管理プレーンは、ネットワークの管理機能を実行し、すべてのプレーン（管理プレーン、コントロール プレーン、データ プレーン）の機能を調整します。 また、管理プレーンはネットワークとの接続を通じてデバイスの管理に使用されます。

管理プレーンで処理されるプロトコルには、簡易ネットワーク管理プロトコル（SNMP）、Telnet、HTTP、セキュア HTTP（HTTPS）、SSH などがあります。 これらの管理プロトコルは、モニタリングやコマンドライン インターフェイス（CLI）のアクセスに使用されます。 デバイスへのアクセスを内部ソース（信頼ネットワーク）に制限することが重要です。

管理プレーン保護機能

MPP 保護機能は、MPP 配下のすべての管理プロトコルと同様、デフォルトではディセーブルになっています。 インターフェイスをアウトオブバンドまたはインバンドとして設定すると、インターフェイスは自動的に MPP をイネーブルにします。 これにより、MPP 配下のすべてのプロトコルもイネーブルになります。

MPP がディセーブルでプロトコルがアクティブな場合、トラフィックはすべてのインターフェイスを通過できます。

アクティブなプロトコルが存在する状態でMPPがイネーブルになると、管理トラフィックを許可するデフォルトの管理インターフェイスはルート プロセッサ（RP）およびスタンバイ ルート プロセッサ（SRP）のイーサネットインターフェイスのみになります。 MPP をイネーブルにする他のすべてのインターフェイスについては、次に説明する MPP CLI を使用して、手動で管理インターフェイスとして設定する必要があります。 以後は、デフォルト管理インターフェイスと事前にMPPインターフェイスとして設定したインターフェイスのみがデバイス宛のネットワーク管理パケットを受け付けます。 他のすべてのインターフェイスは、デバイス宛のネットワーク管理パケットをドロップします。

（注） 論理インターフェイス（またはデータ プレーンに存在しない他のすべてのインターフェイス）は、入力物理インターフェイスに基づいてパケットをフィルタリングします。

設定後に、管理インターフェイスを変更または削除できます。

MPP機能がサポートしている管理プロトコルは、次のとおりです。 これらの管理プロトコルは、MPP がイネーブルになった際に影響を受ける唯一のプロトコルでもあります。

- SSH v1 と v2
- SNMP のすべてのバージョン
- Telnet
- TFTP
- HTTP
- HTTPS

管理プレーン保護機能のメリット

MPP 機能の実装には次の利点があります。

- デバイスの管理における、すべてのインターフェイスで管理プロトコルを許可するよりも優れたアクセス制御の実現。
- 管理インターフェイスでないインターフェイスでのデータ パケットのパフォーマンスの向上。

- ネットワークの拡張性のサポート。
- デバイスへの管理アクセスを制限する、インターフェイス別のアクセス コントロール リスト（ACL）を使用する作業のシンプル化。
- デバイスへのアクセス制限に必要な ACL 数の低減。
- スイッチングインターフェイスおよびルーティングインターフェイスのパケット フラッディングによる CPU への影響を防止。

管理プレーン保護のデバイスの設定方法

ここでは、次の作業について説明します。

インバンド インターフェイスの管理プレーン保護のデバイスの設定

ネットワークに追加した直後のデバイスや、ネットワークですでに動作しているデバイスを設定するには、この作業を実行します。 この作業では、特定のインターフェイスを通じてのみ Telnet のルータへのアクセスが許可されるインバンドインターフェイスとして、MPP を設定する方法について説明します。

デフォルトでない VRF でインバンド MPP インターフェイスを設定するには、次の作業を追加で実行します。

- デフォルトでないインバンド VRF のインターフェイスを設定します。
- グローバル インバンド VRF を設定します。
- Telnet の場合は、インバンド VRF に対して Telnet VRF サーバを設定します。

手順の概要

1. **configure**
2. control-plane
3. management-plane
4. inband
5. **interface** {*type instance* | **all**}
6. **allow** {*protocol* | **all**} [**peer**]
7. **address ipv4** {*peer-ip-address* | *peer ip-address/length*}
8. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
9. **show mgmt-plane** [**inband** | **out-of-band**] [**interface** {*type instance*}]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ 2	control-plane
	例：
	`RP/0/RSP0/CPU0:router(config)# control-plane`
	`RP/0/RSP0/CPU0:router(config-ctrl)#`
		コントロールプレーン コンフィギュレーション モードを開始します。

ステップ 3	management-plane
	例：
	`RP/0/RSP0/CPU0:router(config-ctrl)# management-plane`
	`RP/0/RSP0/CPU0:router(config-mpp)#`
		管理プレーン保護を設定してプロトコルを許可および拒否し、管理プレーン保護コンフィギュレーション モードを開始します。

ステップ 4	inband
	例：
	`RP/0/RSP0/CPU0:router(config-mpp)# inband`
	`RP/0/RSP0/CPU0:router(config-mpp-inband)#`
		インバンドインターフェイスを設定し、管理プレーン保護インバンドコンフィギュレーション モードを開始します。

ステップ 5	**interface** {*type instance* | **all**}
	例：
	`RP/0/RSP0/CPU0:router(config-mpp-inband)# interface GigabitEthernet 0/6/0/1`
	`RP/0/RSP0/CPU0:router(config-mpp-inband-Gi0_6_0_1)#`
		特定のインバンドインターフェイスを設定するか、すべてのインバンド インターフェイスを設定します。管理プレーン保護インバンドインターフェイス コンフィギュレーション モードを開始するには、**interface** コマンドを使用します。
		• **all** キーワードを使用して、すべてのインターフェイスを設定します。

ステップ 6	**allow** {*protocol* | **all**} [**peer**]
	例：
	`RP/0/RSP0/CPU0:router(config-mpp-inband-Gi0_6_0_1)# allow Telnet peer`
	`RP/0/RSP0/CPU0:router(config-telnet-peer)#`
		指定されたプロトコルまたはすべてのプロトコルに対するインバンド インターフェイスとして、インターフェイスを設定します。
		• *protocol* 引数を使用して、指定管理インターフェイスで管理プロトコルを許可します。
		◦ HTTP または HTTPS
		◦ SNMP（バージョンも）

	コマンドまたはアクション	目的
		◦セキュア シェル（v1 および v2） ◦TFTP ◦Telnet • **all** キーワードを使用して、プロトコルのリストで指定されるすべての管理トラフィックを許可するようにインターフェイスを設定します。 •（任意）**peer** キーワードを使用して、インターフェイスでピア アドレスを設定します。
ステップ **7**	**address ipv4** {*peer-ip-address* | *peer ip-address/length*} 例： `RP/0/RSP0/CPU0:router(config-telnet-peer)# address ipv4 10.1.0.0/16`	このインターフェイス上で管理トラフィックが許可されるピア IPv4 アドレスを設定します。 • *peer-ip-address* 引数を使用して、このインターフェイス上で管理トラフィックが許可されるピア IPv4 アドレスを設定します。 • *peer ip-address/length* 引数を使用して、ピア IPv4 アドレスのプレフィックスを設定します。

ステップ **8**	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。

	コマンドまたはアクション	目的
		• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。
ステップ 9	**show mgmt-plane** [**inband** | **out-of-band**] [**interface** {*type instance*}] 例： `RP/0/RSP0/CPU0:router# show mgmt-plane inband interface GigabitEthernet 0/6/0/1`	インターフェイスのタイプやインターフェイスでイネーブルにされるプロトコルなど、管理プレーンに関する情報を表示します。 • （任意）**inband** キーワードを使用して、管理パケットおよびデータ転送パケットを処理するインターフェイスであるインバンド管理インターフェイスの設定を表示します。 • （任意）**out-of-band** キーワードを使用して、アウトオブバンドインターフェイス設定を表示します。 • （任意）**interface** キーワードを使用して、特定のインターフェイスの詳細を表示します。

アウトオブバンドインターフェイスの管理プレーン保護のデバイスの設定

アウトオブバンド MPP インターフェイスを設定するには、次の作業を実行します。

- アウトオブバンド VRF のインターフェイスを設定します。
- グローバル アウトオブバンド VRF を設定します。
- Telnet の場合は、アウトオブバンド VRF に対して Telnet VRF サーバを設定します。

手順の概要

1. **configure**
2. control-plane
3. management-plane
4. out-of-band
5. **vrf** *vrf-name*
6. **interface** {*type instance* | **all**}
7. **allow** {*protocol* | **all**} [**peer**]
8. **address ipv6** {*peer-ip-address* | *peer ip-address/length*}
9. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
10. **show mgmt-plane** [**inband** | **out-of-band**] [**interface** {*type instance*} | **vrf**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ 2	control-plane
	例：
	`RP/0/RSP0/CPU0:router(config)# control-plane`
	`RP/0/RSP0/CPU0:router(config-ctrl)#`
		コントロール プレーン コンフィギュレーション モードを開始します。

ステップ 3	management-plane
	例：
	`RP/0/RSP0/CPU0:router(config-ctrl)# management-plane`
	`RP/0/RSP0/CPU0:router(config-mpp)#`
		管理プレーン保護を設定してプロトコルを許可および拒否し、管理プレーン保護コンフィギュレーション モードを開始します。

ステップ 4	out-of-band 例： `RP/0/RSP0/CPU0:router(config-mpp)# out-of-band`	帯域外インターフェイスまたはプロトコルを設定し、管理プレーン保護帯域外コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	`RP/0/RSP0/CPU0:router(config-mpp-outband)#`	
ステップ 5	**vrf** *vrf-name* 例： `RP/0/RSP0/CPU0:router(config-mpp-outband)# vrf target`	帯域外インターフェイスのバーチャル プライベート ネットワーク（VPN）Routing and Forwarding（VRF; VPN ルーティングおよび転送）リファレンスを設定します。 • *vrf-name* 引数を使用して、VRF に名前を割り当てます。

ステップ 6	**interface** {*type instance* | **all**}
	
	例：
	
	`RP/0/RSP0/CPU0:router(config-mpp-outband)# interface GigabitEthernet 0/6/0/2`
	`RP/0/RSP0/CPU0:router(config-mpp-outband-Gi0_6_0_2)#`
		特定のアウトオブバンド インターフェイス、またはすべてのアウトオブバンド インターフェイスをアウトオブバンド インターフェイスとして設定します。管理プレーン保護アウトオブバンドコンフィギュレーション モードを開始するには、**interface** コマンドを使用します。
		• **all** キーワードを使用して、すべてのインターフェイスを設定します。

ステップ 7	**allow** {*protocol* | **all**} [**peer**]
	
	例：
	
	`RP/0/RSP0/CPU0:router(config-mpp-outband-Gi0_6_0_2)# allow TFTP peer`
	`RP/0/RSP0/CPU0:router(config-tftp-peer)#`
		指定されたプロトコルまたはすべてのプロトコルに対するアウトオブバンドインターフェイスとして、インターフェイスを設定します。
		• *protocol* 引数を使用して、指定管理インターフェイスで管理プロトコルを許可します。
		◦ HTTP または HTTPS
		◦ SNMP（バージョンも）
		◦ セキュア シェル（v1 および v2）
		◦ TFTP
		◦ Telnet
		• **all** キーワードを使用して、プロトコルのリストで指定されるすべての管理トラフィックを許可するようにインターフェイスを設定します。
		• （任意）**peer** キーワードを使用して、インターフェイスでピア アドレスを設定します。

	コマンドまたはアクション	目的
ステップ 8	**address ipv6** {*peer-ip-address* | *peer ip-address/length*} 例： `RP/0/RSP0/CPU0:router(config-tftp-peer)# address ipv6 33::33`	このインターフェイス上で管理トラフィックが許可されるピア IPv6 アドレスを設定します。 • *peer-ip-address* 引数を使用して、このインターフェイス上で管理トラフィックが許可されるピア IPv6 アドレスを設定します。 • *peer ip-address/length* 引数を使用して、ピア IPv6アドレスのプレフィックスを設定します。

ステップ 9	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータがEXECモードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ステップ 10	**show mgmt-plane** [**inband** | **out-of-band**] [**interface** {*type instance*} | **vrf**] 例： `RP/0/RSP0/CPU0:router# show mgmt-plane out-of-band interface GigabitEthernet 0/6/0/2`	インターフェイスのタイプやインターフェイスでイネーブルにされるプロトコルなど、管理プレーンに関する情報を表示します。 • （任意）**inband** キーワードを使用して、管理パケットおよびデータ転送パケットを処理するインターフェイスであるインバンド管理インターフェイスの設定を表示します。

	コマンドまたはアクション	目的
		• （任意）**out-of-band** キーワードを使用して、アウトオブバンド インターフェイス設定を表示します。 • （任意）**interface** キーワードを使用して、特定のインターフェイスの詳細を表示します。 • （任意）**vrf** キーワードを使用して、アウトオブバンド インターフェイスのバーチャル プライベート ネットワーク（VPN）ルーティングおよび転送リファレンスを表示します。

管理プレーン保護の実装の設定例

この項では、次の設定例について説明します。

管理プレーン保護の設定：例

次に、MPP 配下の特定の IP アドレスにインバンドおよびアウトオブバンド インターフェイスを設定する例を示します。

```
configure
 control-plane
  management-plane
   inband
    interface all
     allow SSH
     !
    interface GigabitEthernet 0/6/0/0
     allow all
     allow SSH
     allow Telnet peer
      address ipv4 10.1.0.0/16
     !
    !
    interface GigabitEthernet 0/6/0/1
     allow Telnet peer
      address ipv4 10.1.0.0/16
     !
    !
   !
   out-of-band
    vrf my_out_of_band
    interface GigabitEthernet 0/6/0/2
     allow TFTP peer
      address ipv6 33::33
     !
    !
   !
```

```
  !
 !

show mgmt-plane

Management Plane Protection

inband interfaces
----------------------

interface - GigabitEthernet0_6_0_0
        ssh configured -
                All peers allowed
        telnet configured -
                peer v4 allowed - 10.1.0.0/16
        all configured -
                All peers allowed
interface - GigabitEthernet0_6_0_1
        telnet configured -
                peer v4 allowed - 10.1.0.0/16

interface - all
        all configured -
                All peers allowed

outband interfaces
----------------------
interface - POS0_6_0_2
        tftp configured -
                peer v6 allowed - 33::33

show mgmt-plane out-of-band vrf

Management Plane Protection -
        out-of-band VRF - my_out_of_band
```

参考資料

ここでは、管理プレーン保護の実装に関する関連資料について説明します。

関連資料

関連項目	ドキュメント名
MPP コマンド：コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco ASR 9000 Series Aggregation Services Router System Monitoring Command Reference*』の管理プレーン保護コマンド

標準

標準	タイトル
この機能でサポートが追加または変更された標準はありません。また、この機能で変更された既存の標準のサポートはありません。	—

MIB

MIB	MIB リンク
—	Cisco IOS XR ソフトウェアを使用して MIB を検出およびダウンロードするには、URL（http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml）にある Cisco MIB Locator を使用して、[Cisco Access Products] メニューでプラットフォームを選択します。

RFC

RFC	タイトル
この機能でサポートが追加または変更された RFC はありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカルサポート Web サイトには、数千ページに及ぶ検索可能な技術情報があります。製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクもあります。 Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	http://www.cisco.com/en/US/support/index.html

第 7 章

Software Authentication Manager の設定

Software Authentication Manager（SAM）は、ルータにインストールされているソフトウェアは安全で、整合性が改ざんされていた場合にはソフトウェアは動作しないことを保証する Cisco ASR 9000 シリーズ ルータ オペレーティング システムのコンポーネントです。

SAM コマンドの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』の「*Software Authentication Manager Commands on* Cisco IOS XR ソフトウェア」モジュールを参照してください。

システム クロックの設定の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Management Command Reference*』の「*Clock Commands on* Cisco IOS XR ソフトウェア」モジュールにある **clock set** コマンドを参照してください。

Software Authentication Manager の設定の機能履歴

リリース	変更点
リリース 3.7.2	この機能を追加しました。

Software Authentication Manager の設定に関する前提条件

適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。

Software Authentication Manager について

SAM がインストール中にソフトウェアを検証するには、インストールされるソフトウェアは IOS/ENA（PIE）形式のパッケージャ内に存在する必要があります。 PIE はデジタル署名されており、SAM は PIE のデータのルータへの配置を許可する前にデジタル署名を検証します。 インストールされたソフトウェアの一部が実行されるたびに、SAM はソフトウェアの整合性がインストール時から改ざんされていないことを確認します。 SAM は、フラッシュ カードにプリインストールされているソフトウェアが送信中に改ざんされていないことも検証します。

初期イメージまたはソフトウェア パッケージの更新版がルータにロードされると、SAM はイメージの署名に使用された証明書の有効期限をチェックして、イメージの有効性を検証します。 証明書の有効期限が切れていることを示すエラー メッセージが表示された場合は、システム クロックをチェックし、正確であることを確認してください。 システム クロックが正しく設定されていない場合、システムは正常に機能しません。

Software Authentication Manager のプロンプト インターバルの設定方法

SAM は起動時に異常な状態を検出すると、ユーザにアクションを実行するように求め、一定時間待機します。 ユーザがこの時間の間に応答しないと、SAM は事前定義されたアクションを実行します。このアクションも設定可能です。

プロンプト インターバルを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **sam promptinterval** *time-interval* {**proceed** | **terminate**}
3. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**sam promptinterval** *time-interval* {**proceed** | **terminate**} 例： `RP/0/RSP0/CPU0:router(config)# sam prompt-interval 25 {proceed | terminate}`	SAM が次のアクションを実行するか、インターバルを終了するまでのプロンプトインターバルを秒単位で設定します。プロンプトインターバルの範囲は 0 ～ 300 秒です。 ユーザが応答した場合、SAM はユーザが同意したと見なし、次のアクションを実行します。ユーザが応答しなかった場合、SAM はユーザが拒否したと見なし、アクションを終了します。SAM が待機するデフォルトの時間は 10 秒です。

ステップ 3	次のいずれかのコマンドを使用します。
	
	• **end**
	
	• **commit**
	
	例：
	
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。
		
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。
		
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

第 8 章

セキュア シェルの実装

セキュア シェル（SSH）は、Berkeley の r ツールへのセキュアな置換を提供するアプリケーションおよびプロトコルです。 プロトコルは標準の暗号メカニズムを使用してセッションの安全を確保します。アプリケーションは Berkeley の **rexec** および **rsh** ツールと同様に使用できます。

SSH サーバには、SSH バージョン 1（SSHv1）と SSH バージョン 2（SSHv2）の 2 種類のバージョンがあります。 SSHv1 は Rivest, Shamir, and Adelman（RSA）キーを使用し、SSHv2 はデジタル署名アルゴリズム（DSA）キーを使用します。 Cisco IOS XR ソフトウェア は SSHv1 と SSHv2 の両方をサポートしています。

このモジュールでは、Cisco ASR 9000 シリーズ ルータ でのセキュア シェルの実装方法について説明します。

（注） このモジュールで使用されているセキュア シェル コマンドの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』マニュアルの「*Secure Shell Commands*」モジュールを参照してください。 このモジュールの他のコマンドに関するマニュアルについては、コマンド リファレンス マスター インデックスを使用するか、オンラインで検索します。

セキュア シェルの実装の機能履歴

リリース	変更点
リリース 3.7.2	この機能を追加しました。
リリース 3.9.0	次の拡張サポートが追加されました。 • SSH サーバでの RSA ベース認証 • インタラクティブ モードでの SFTP クライアント • SFTP サーバ実装

セキュア シェルの実装に関する前提条件

セキュア シェルを実装するには、次の前提条件を満たす必要があります。

- 適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。 ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。
- 必要なイメージをルータにダウンロードします。 SSH サーバと SSH クライアントでは、暗号化パッケージ（データ暗号規格（DES）、トリプル DES、および AES）をシスコからご使用のルータにダウンロードする必要があります。
- SSHv2 サーバを実行するには、VRF が必要です。 これはデフォルトの VRF でも固有の VRF でも構いません。 VRF に関する変更は SSH v2 サーバのみに適用されます。
- ローカル アクセスまたはリモート アクセス用にユーザ認証を設定します。 認証、許可、アカウンティング（AAA）の有無に関係なく、認証を設定できます。 詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』マニュアルの「*Authentication, Authorization, and Accounting Commands on* Cisco IOS XR ソフトウェア」モジュールおよび『*Cisco ASR 9000 Series Aggregation Services Router System Security Configuration Guide*』マニュアルの「*Configuring AAA Services on*Cisco IOS XR ソフトウェア」モジュールを参照してください。
- セキュア シェル ファイル転送プロトコル（SFTP）が動作するには、AAA 認証および認可を正しく設定する必要があります。

セキュア シェルの実装に関する制約事項

SSH の基本的な制約事項と SFTP 機能の制限は、次のとおりです。

- VRF がすでにアウトオブバンド VRF として設定されている場合は、VRF はインバンドとしては受け付けられません。 SSH v1 は継続してデフォルトの VRF のみにバインドします。
- 外部クライアントがルータに接続するには、ルータに RSA（SSHv1）または DSA（SSHv2）キー ペアが設定されている必要があります。 ルータから外部ルーティング デバイスに SSH クライアント接続を開始する場合、DSA および RSA キーは必要ありません。 これは SFTP

も同様です。SFTP はクライアント モードでのみ動作するため、DSA および RSA キーは必要ありません。

- SFTP が正常に動作するには、リモート SSH サーバは SFTP サーバ機能をイネーブルにする必要があります。 たとえば、**/etc/ssh2/sshd2_config** などの行を使用して、SFTP サブシステムを処理するように SSHv2 サーバを設定します。
- **subsystem-sftp /usr/local/sbin/sftp-server**
- SFTP サーバは通常パブリック ドメインの SSH パッケージの一部として含まれており、デフォルトの構成では有効になっています。
- SFTP は、SFTP サーバ バージョン OpenSSH_2.9.9p2 以上と互換性があります。
- SSH サーバおよび SFTP サーバでは、RSA ベースのユーザ認証がサポートされています。 ただし、SSH クライアントではこの認証はサポートされていません。
- サポートされるアプリケーションは、実行シェルおよび SFTP のみです。
- SSHv2 サーバおよびクライアントでは AES 暗号化アルゴリズムがサポートされていますが、SSHv1 サーバおよびクライアントではサポートされていません。 SSHv2 クライアントから SSHv1 サーバに送信された AES 暗号の要求はすべて無視されます。代わりにサーバではトリプル DES を使用します。
- SFTP クライアントは、ワイルドカード（*、?、[]）を含むリモート ファイル名をサポートしません。 ソース ファイルをルータにダウンロードするには、ユーザは **sftp** コマンドを複数回発行するか、リモート ホストからすべてのソース ファイルを表示する必要があります。アップロードについては、この項の 1 番目から 3 番目までの箇条書きで示した問題が解決されている場合、ルータ SFTP クライアントはワイルドカードを使用した複数ファイルの指定をサポートできます。
- SSH サーバの暗号化設定は、AES128、AES192、AES256、トリプル DES の順です。 サポートされていない暗号の場合、サーバはクライアントの要求をすべて拒否し、SSH セッションは続行されません。
- vt100 以外の端末タイプの使用はサポートされていません。この場合、ソフトウェアは警告メッセージを生成します。
- SSH クライアントでは、パスワード メッセージとして「none」を使用することはサポートされていません。
- ルータ インフラストラクチャは UNIX 同様のファイル権限をサポートしていないため、ローカル デバイスに作成されたファイルは元の権限情報を失います。 リモート ファイル システム上に作成されたファイルの場合、ファイル権限は宛先ホストの umask に従い、変更時間および最終アクセス時間はコピーの時間になります。

セキュア シェルの実装について

SSH を実装するには、次の概念について理解しておく必要があります。

SSH サーバ

SSH サーバの機能によって、SSH クライアントは Cisco ルータに対してセキュアで暗号化された接続を実行できます。 この接続は、インバウンド Telnet 接続の機能と同様です。 SSH 以前は、セキュリティは Telnet のセキュリティに限定されていました。 SSH を Cisco IOS XR ソフトウェア認証と併用することで、強力な暗号化が可能になりました。 Cisco IOS XR ソフトウェア の SSH サーバは、無償あるいは市販の SSH クライアントとの相互運用が可能です。

SSH クライアント

SSH クライアント機能は、SSH プロトコルを介して実行されるアプリケーションで、認証と暗号化を行います。 SSH クライアントによって、Cisco ルータは他の Cisco ルータ、または SSH サーバを実行する他のデバイスに対して、セキュアで暗号化された接続を実行できます。 この接続は、接続が暗号化されている点を除き、アウトバウンド Telnet 接続の機能と同様です。 認証と暗号化により、SSH クライアントは、セキュリティ保護されていないネットワーク上でもセキュアな通信を実現できます。

Cisco IOS XR ソフトウェア の SSH クライアントは、無償あるいは市販の SSH サーバとの相互運用が可能でした。 SSH クライアントは、AES、トリプル DES、メッセージ ダイジェスト アルゴリズム 5（MD5）、SHA1、およびパスワード認証による暗号化をサポートしていました。 ユーザ認証はルータへの Telnet セッション内で実行されました。 SSH がサポートするユーザ認証メカニズムには、RADIUS、TACACS+、およびローカルに格納されたユーザ名とパスワードを使用した認証がありました。

SFTP 機能の概要

SSH には、SSHv2 で導入された新たな標準ファイル転送プロトコルである Standard File Transfer Protocol（SFTP）のサポートが含まれています。 この機能は、ルータ設定またはルータ イメージ ファイルをコピーするセキュアで認証された方法を提供します。

SFTP クライアント機能は SSH コンポーネントの一部として提供され、ルータで常にイネーブルになっています。 このため、適切なレベルのユーザは、ルータへのファイルのコピーおよびルータからのファイルのコピーが可能です。 **copy** コマンドと同様に、**sftp** コマンドは EXEC モードでしか使用できません。

SFTP クライアントは VRF 対応であるため、接続の試行時に特定の送信元インターフェイスに関連付けられた VRF を使用するようにセキュア FTP クライアントを設定することもできます。 SFTP クライアントはインタラクティブ モードもサポートしています。このモードでは、ユーザはサーバにログインして特定の作業を UNIX サーバ経由で実行できます。

SFTP サーバは SSH サーバのサブシステムです。 つまり、SSH サーバが SFTP サーバ要求を受信すると、SFTP API は SSH サーバに対して子プロセスとして SFTP サーバを作成します。 新たな要求のたびに、新しい SFTP サーバ インスタンスが作成されます。

SFTP は、次の手順で新たな SFTP サーバを要求します。

- ユーザは必要な引数を付与して **sftp** コマンドを実行します
- SFTP API は SSH サーバと通信する子プロセスを内部に作成します
- SSH サーバは SFTP サーバ子プロセスを作成します
- SFTP サーバおよびクライアントは暗号化形式で相互に通信します

SSH サーバが SSH クライアントと新たな接続を確立すると、サーバ デーモンは新たな SSH サーバ子プロセスを作成します。 子サーバ プロセスは、キー交換とユーザ認証プロセスによって、SSH クライアントとサーバとの間にセキュアな通信チャネルを構築します。 SSH サーバがサブシステムを SFTP サーバにする要求を受信した場合、SSH サーバ デーモンは SFTP サーバ子プロセスを作成します。 SFTP サーバ サブシステム要求を受信するたびに、新たな SSH サーバ子インスタンスおよび SFTP サーバ インスタンスが作成されます。 SFTP サーバはユーザ セッションを認証し、接続を開始します。ユーザのデフォルトディレクトリおよびクライアントの環境を設定します。

初期化が実行されると、SFTP サーバはクライアントからの SSH_FXP_INIT メッセージを待機します。このメッセージは、ファイル通信セッションを開始するためには不可欠です。 このメッセージの後に、クライアントの要求に基づいたメッセージが続く場合があります。 ここでは、プロトコルは「要求応答」モデルを採用しています。クライアントがサーバに要求を送信すると、サーバはこの要求を処理し応答を送信します。

SFTP サーバは次の応答を表示します。

- ステータス応答
- 処理応答
- データ応答
- 名前応答

（注） サーバは、着信する SFTP 接続を受け付けるために稼働している必要があります。

RSA ベースのホスト認証

サーバの正当性を検証することは、セキュアな SSH 接続を実現する最初の手順です。 このプロセスはホスト認証と呼ばれ、クライアントが有効なサーバに接続していることを確認するために実施されます。

ホスト認証はサーバの公開キーを使用して実行されます。 サーバは、キー交換フェーズの間に公開キーをクライアントに提供します。 クライアントはこのサーバの既知ホストのデータベースと、対応する公開キーをチェックします。クライアントでサーバの IP アドレスが見つからなかった場合は、ユーザに警告メッセージを表示し、ユーザは公開キーを保存するか廃棄するかを選択できます。サーバの IP アドレスは見つかったものの公開キーが一致しない場合、クライアントは

接続を終了します。 公開キーが有効な場合、サーバは検証され、セキュアなSSH接続が確立されます。

IOS XR SSH サーバおよびクライアントは、DSA ベースのホスト認証をサポートしていました。ただし、IOS などの他の製品との互換性のため、RSA ベースのホスト認証のサポートも追加されました。

RSA ベースのユーザ認証

SSH プロトコルにおいてユーザを認証する方法の 1 つに、RSA 公開キー ベースのユーザ認証があります。 秘密キーの保持がユーザ認証の役割を果たします。 この方法は、ユーザの秘密キーで作成した署名を送信することで機能します。 各ユーザは RSA キーペアをクライアント マシンに保持しています。 RSA キーペアの秘密キーはクライアント マシンに残ったままです。

ユーザは、ssh-keygen などの標準的なキー生成メカニズムを使用して、RSA 公開キーと秘密キーのキーペアをUNIXクライアント上に生成します。 サポートされているキーの最大の長さは2048ビットで、最小の長さは 512 ビットです。 次に、一般的なキー生成アクティビティの例を示します。

```
bash-2.05b$ ssh-keygen -b 1024 -t rsa
Generating RSA private key, 1024 bit long modulus
```

公開キーを正常にボックスにインポートするには、公開キーはBase64エンコード（バイナリ）形式である必要があります。 インターネットで入手できるサード パーティのツールを使用して、キーをバイナリ形式に変換できます。

公開キーがルータにインポートされると、SSH クライアントは内部で「-o」オプションを使用して要求を指定することで、公開キー認証方式を使用できます。 例：

```
client$ ssh -o PreferredAuthentications=publickey 1.2.3.4
```

公開キーがRSA方式を使用してルータにインポートされていない場合、SSHサーバはパスワードベースの認証を開始します。 公開キーがインポートされている場合、サーバは両方の方式の使用を提案します。 SSHクライアントはいずれかの方式を使用して、接続を確立します。 システムでは、発信する SSH クライアント接続の数を 10 まで許可しています。

現在、SSH バージョン 2 および SFTP サーバのみが RSA ベースの認証をサポートしています。 公開キーのルータへのインポート方法の詳細については、このガイドの「*Implementing Certification Authority Interoperability on Cisco ASR 9000* シリーズ ルータ（認証局相互運用性の実装）」の章を参照してください。

（注） 推奨される認証方法は SSH RFC に記載されています。 RSA ベース認証のサポートはローカル認証のみです。TACACS/RADIUS サーバに対してはサポートされていません。

認証、許可、およびアカウンティング（AAA）は、Cisco ルータまたはアクセス サーバにアクセス コントロールを設定できる主要なフレームワークを提供する一連のネットワーク セキュリティ サービスです。 AAA の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』マニュアルの「*Authentication, Authorization, and Accounting Commands on* Cisco ASR 9000 シリーズ ルータ *Software*」モジュールおよび『*Cisco ASR 9000 Series Aggregation*

Services Router System Security Configuration Guide』マニュアルの「*Configuring AAA Services on* Cisco ASR 9000 シリーズ ルータ 」モジュールを参照してください。

セキュア シェルの実装方法

SSH を設定するには、次の項で説明する作業を実行します。

SSH の設定

SSH を設定するには、次の作業を実行します。

（注） SSHv1 の設定では、ステップ 1 ～ 4 は必須です。 SSHv2 の設定では、ステップ 1 ～ 4 は任意です。

手順の概要

1. **configure**
2. **hostname** *hostname*
3. **domain name** *domain-name*
4. exit
5. **crypto key generate rsa** [**usage keys** | **general-keys**] [*keypair-label*]
6. crypto key generate dsa
7. configure
8. **ssh timeout** *seconds*
9. 次のいずれかを実行します。
   - **ssh server** [**vrf** *vrf-name*]
   - ssh server v2
10. 次のいずれかのコマンドを使用します。
    - **end**
    - **commit**
11. show ssh
12. show ssh session details

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**hostname** *hostname* 例： `RP/0/RSP0/CPU0:router(config)# hostname router1`	ルータのホスト名を設定します。
ステップ 3	**domain name** *domain-name* 例： `RP/0/RSP0/CPU0:router(config)# domain name cisco.com`	ソフトウェアで使用するデフォルトのドメイン名を定義し、不完全なホスト名のドメインを補完します。
ステップ 4	exit 例： `RP/0/RSP0/CPU0:router(config)# exit`	グローバル コンフィギュレーション モードを終了して、ルータを EXEC モードに戻します。
ステップ 5	**crypto key generate rsa** [**usage keys** | **general-keys**] [*keypair-label*] 例： `RP/0/RSP0/CPU0:router# crypto key generate rsa general-keys`	RSA キー ペアを生成します。 • RSA キー ペアを削除するには、**crypto key zeroize rsa** コマンドを使用します。 このコマンドは SSHv1 だけに使用されます。
ステップ 6	crypto key generate dsa 例： `RP/0/RSP0/CPU0:router# crypto key generate dsa`	ルータでローカルおよびリモート認証を行う SSH サーバをイネーブルにします。 • 推奨する最小絶対サイズは 1024 ビットです。 • DSA キー ペアを生成します。 DSA キー ペアを削除するには、**crypto key zeroize dsa** コマンドを使用します。 このコマンドは SSHv2 だけに使用されます。
ステップ 7	configure 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 8	**ssh timeout** *seconds* 例： `RP/0/RSP0/CPU0:router(config)# ssh timeout 60`	（任意）AAA へのユーザ認証に対するタイムアウト値を設定します。 • 設定された時間内にユーザ自身の AAA への認証に失敗すると、接続は中断されます。 • 値を設定しなければ、30秒のデフォルト値が使用されます。範囲は 5 ～ 120 です。

ステップ 9	次のいずれかを実行します。
	• **ssh server** [**vrf** *vrf-name*]
	• ssh server v2
	
	例：
	`RP/0/RSP0/CPU0:router(config)# ssh server`
	または
	`RP/0/RSP0/CPU0:router(config)# ssh server v2`
		• （任意）最大 32 文字の指定された VRF を使用して、SSH サーバを起動します。 VRFが指定されていない場合、デフォルトの VRF が使用されます。
		SSH サーバが指定された VRF の接続をこれ以上受信しないようにするには、このコマンドの **no** 形式を使用します。VRFが指定されていない場合、デフォルトが使用されます。
		（注） SSH サーバは複数の VRF で使用できるように設定できます。
		• （任意）**ssh server v2** コマンドを使用して SSHv2 オプションを設定した場合に、SSH サーバが SSHv2 クライアントのみを受け付けるようにします。 **ssh server v2** コマンドを選択した場合は、SSH v2 クライアント接続のみを受け付けます。

ステップ 10	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。

	コマンドまたはアクション	目的
		• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。
ステップ 11	show ssh 例： `RP/0/RSP0/CPU0:router# show ssh`	（任意）ルータで発着信するすべての SSHv1 および SSHv2 接続を表示します。
ステップ 12	show ssh session details 例： `RP/0/RSP0/CPU0:router# show ssh session details`	（任意）ルータで発着信する SSHv2 接続の詳細レポートを表示します。

SSH クライアントの設定

SSH クライアントを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **ssh client knownhost** *device* **:** */filename*
3. exit
4. **ssh** {*ipv4-address* | *hostname*} [**username** *user- id* | **cipher des** | **source-interface** *type instance*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**ssh client knownhost** *device* **:** */filename* 例： `RP/0/RSP0/CPU0:router(config)# ssh client knownhost slot0:/server_pubkey`	（任意）クライアント側でサーバの公開キー（pubkey）を認証およびチェックする機能をイネーブルにします。 （注） ファイル名の完全なパスが必要です。 コロン（:）とスラッシュ マーク（/）も必要です。
ステップ 3	exit 例： `RP/0/RSP0/CPU0:router(config)# exit`	グローバル コンフィギュレーション モードを終了して、ルータを EXEC モードに戻します。
ステップ 4	**ssh** {*ipv4-address* | *hostname*} [**username** *user- id* | **cipher des** | **source-interface** *type instance*] 例： `RP/0/RSP0/CPU0:router# ssh remotehost username user1234`	アウトバウント SSH 接続をイネーブルにします。 • SSHv2 サーバを実行するには、VRF が必要です。 これはデフォルトの VRF でも固有の VRF でも構いません。 VRF に関する変更は SSH v2 サーバのみに適用されます。 • SSH クライアントにより、リモート ピアへの SSHv2 接続が試みられます。 リモート ピアで SSHv1 サーバしかサポートされていない場合、リモート サーバへの SSHv1 接続が内部生成されます。 • **cipher des** オプションは、SSHv1 クライアントでしか使用できません。 • SSHv1 クライアントは、これらの SSH クライアントのみデフォルトで利用できるトリプル DES 暗号化アルゴリズム オプションのみをサポートしています。 • *hostname* 引数が使用されており、ホストが IPv4 アドレスと IPv6 アドレスの両方を持っている場合、IPv6 アドレスが使用されます。

- SSHv1 を使用しており、SSH 接続が拒否されている場合は、ルータの RSA キー ペアが適切に生成されていません。 ホスト名およびドメインを指定していることを確認します。 次に、**crypto key generate rsa** コマンドを使用して RSA キー ペアを生成し、SSH サーバをイネーブルにします。
- SSHv2 を使用しており、SSH 接続が拒否されている場合は、ルータの DSA キー ペアが適切に生成されていません。 ホスト名およびドメインを指定していることを確認します。 次に、**crypto key generate dsa** コマンドを使用して DSA キー ペアを生成し、SSH サーバをイネーブルにします。

- RSA または DSA キー ペアを設定すると、次のエラー メッセージが表示されることがあります。
  - No hostname specified

  hostname グローバル コンフィギュレーション コマンドを使用して、ルータのホスト名を設定する必要があります。
  - No domain specified

  domain-name グローバル コンフィギュレーション コマンドを使用して、ルータのホスト ドメインを設定する必要があります。
- 使用できる SSH 接続数は、ルータに設定されている仮想端末回線の最大数に制限されます。各 SSH 接続は vty リソースを使用します。
- SSH では、ルータで AAA によって設定されるローカル セキュリティまたはセキュリティ プロトコルが、ユーザ認証に使用されます。 AAA を設定する場合、コンソール上で AAA を無効にするためにグローバル コンフィギュレーション モードでキーワードを適用することにより、コンソールが AAA の下で実行されていないことを確認する必要があります。

セキュア シェルの実装の設定例

この項では、次の設定例について説明します。

セキュア シェルの設定：例

次に、SSHv2 を設定する例を示します。この手順には、ホスト名の作成、ドメイン名の定義、DSA キーペアの生成によるルータでのローカルおよびリモート認証用の SSH サーバのイネーブル化、SSH サーバの起動、および実行コンフィギュレーション ファイルへのコンフィギュレーション コマンドの保存が含まれています。

SSH の設定が完了すると、ルータで SFTP 機能が使用できます。

```
configure
hostname router1
domain name cisco.com
exit
crypto key generate dsa
configure
ssh server
end
```

参考資料

ここでは、セキュア シェルの実装に関する関連資料について説明します。

関連資料

関連項目	ドキュメント名
AAA コマンド：コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』の「*Authentication, Authorization, and Accounting Commands on* Cisco ASR 9000 シリーズ ルータ *Software*」モジュール。
AAA 設定作業	『*Cisco ASR 9000 Series Aggregation Services Router System Security Configuration Guide*』の「*Configuring AAA Services on* Cisco ASR 9000 シリーズ ルータ *Software*」モジュール。
ホスト サービスおよびアプリケーション コマンド：詳細なコマンド構文、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco ASR 9000 Series Aggregation Services Router IP Addresses and Services Command Reference*』の「*Host Services and Applications Commands on* Cisco ASR 9000 シリーズ ルータ」モジュール。
IPSec コマンド：コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』の「*IPSec Network Security Commands on* Cisco ASR 9000 シリーズ ルータ *Software*」モジュール
SSH コマンド：コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』の「*Secure Shell Commands on* Cisco ASR 9000 シリーズ ルータ *Software*」モジュール

標準

標準	タイトル
Draft-ietf-secsh-userauth-17.txt	『*SSH Authentication Protocol*』（2003 年 7 月）
Draft-ietf-secsh-connect-17.txt	『*SSH Connection Protocol*』（2003 年 7 月）
Draft-ietf-secsh-architecture-14.txt	『*SSH Protocol Architecture*』（2003 年 7 月）
Draft-ietf-secsh-transport-16.txt	『*SSH Transport Layer Protocol*』（2003 年 7 月）

MIB

MIB	MIB リンク
—	Cisco IOS XR ソフトウェアを使用して MIB を検出およびダウンロードするには、URL（http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml）にある Cisco MIB Locator を使用して、[Cisco Access Products] メニューでプラットフォームを選択します。

RFC

RFC	タイトル
この機能でサポートが追加または変更された RFC はありません。また、この機能で変更された既存の RFC のサポートはありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカルサポート Web サイトには、数千ページに及ぶ検索可能な技術情報があります。製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクもあります。 Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	http://www.cisco.com/en/US/support/index.html

第 9 章

Secure Socket Layer の実装

このモジュールでは、SSL の実装方法について説明します。

Secure Socket Layer（SSL）プロトコルと Transport Layer Security（TLS）は、相互認証、整合性を目的としたハッシュの使用、プライバシーを目的とした暗号化を許可することで、クライアントとサーバとの間のセキュアな通信を提供するアプリケーションレベルのプロトコルです。 SSL および TLS は証明書、公開キー、および秘密キーを使用します。

証明書はデジタル ID カードに似ています。 この証明書は、クライアントに対してサーバの ID を証明します。 VeriSign や Thawte などの認証局（CA）が証明書を発行します。 各証明書には、発行した機関の名前、証明書の発行先エンティティの名前、エンティティの公開キー、および証明書の有効期限を示すタイムスタンプが含まれます。

公開キーおよび秘密キーは、情報の暗号化および復号化に使用される暗号キーです。 公開キーは非常に簡単に共有されますが、秘密キーは公開されることはありません。 公開キーと秘密キーの各キー ペアは連携して動作します。公開キーで暗号化されたデータは秘密キーでのみ復号化できます。

（注） このモジュールで使用されている公開キーインフラストラクチャ（PKI）の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』の「*Public Key Infrastructure Commands on* Cisco ASR 9000 Series Router」モジュールを参照してください。このモジュールの他のコマンドに関するマニュアルについては、コマンド リファレンス マスター インデックスを使用するか、オンラインで検索します。

Secure Socket Layer の実装の機能履歴

リリース	変更点
リリース 3.7.2	この機能を追加しました。

- Secure Socket Layer の実装に関する前提条件，200 ページ

Secure Socket Layer の実装に関する前提条件

SSL を実装するには、次の前提条件を満たす必要があります。

- 適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。 ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。
- セキュリティ ソフトウェアのパッケージインストレーション エンベロープ（PIE）をインストールしてアクティブにする必要があります。

  オプションの PIE インストールの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide*』を参照してください。
- SSL の使用を開始する前に、Rivest, Shamir, and Adelman（RSA）またはデジタル署名アルゴリズム（DSA）キー ペアを生成し、CA に登録して、ルータ キーの CA 証明書を取得する必要があります。
- SSL サーバは Advanced Encryption Standard（AES）をサポートしています。AES のキー サイズには 128 ビット、192 ビット、および 256 ビットがあります。

  これらの作業の実行に必要なコマンドの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』の「*Public Key Infrastructure Commands on* Cisco ASR 9000 シリーズ ルータ」モジュールの **crypto key generate rsa** コマンド、**crypto key generate dsa** コマンド、**crypto ca enroll** コマンド、および **crypto ca authenticate** コマンドを参照してください。

Secure Socket Layer の実装について

SSL を実装するには、次の概念を理解しておく必要があります。

認証局の目的

認証局（CA）は、証明書要求を管理し、関係する IPSec ネットワーク デバイスへの証明書を発行します。 これらのサービスは、参加デバイスのキー管理を一元化して行います。

CA は、IPSec ネットワーク デバイスの管理を簡素化します。 CA は、ルータなど、複数の IPSec 対応デバイスを含むネットワークで使用できます。

Public Key Cryptography によりイネーブルにされたデジタル署名は、デバイスおよび個人ユーザをデジタル認証します。 RSA 暗号化システムなどの Public Key Cryptography では、各ユーザは、公開キーと秘密キーの両方を含むキーペアを使用します。 これらのキーは、補足として機能し、一方で暗号化されたものは、もう一方で復号化できます。 つまり、シグニチャは、データがユーザの秘密キーで暗号化されるときに形成されます。 受信側は、送信側の公開キーを使用してメッセージを復号化することで、シグニチャを検証します。 メッセージは、送信側の公開キーを使用して復号化できるため、秘密キーの所有者、つまり送信者がメッセージを作成することになります。 このプロセスは、受信者が送信者の公開キーのコピーを持っていて、これが本当に送信者のものであり、送信者を騙る他人のものではないことを高い確実性を持って知っていることを基盤としています。

デジタル証明書はリンクを提供します。 デジタル証明書には、名前、シリアル番号、企業、部署またはIPアドレスなど、ユーザまたはデバイスを特定する情報を含んでいます。 また、エンティティの公開キーのコピーも含んでいます。 証明書自体は、受信者が身元を証明し、デジタル証明書を作成するうえで確実に信頼できるサード パーティである、CA により署名されます。

CA のシグニチャを検証するには、受信者は、CA の公開キーを認識している必要があります。 通常、このプロセスは、アウトオブバンドで、またはインストールで行われる操作により処理されます。 たとえば、通常の Web ブラウザでは、デフォルトで、複数の CA の公開キーが設定されています。 IPSec の基本コンポーネントであるインターネット キー交換（IKE）は、デジタル シグニチャを使用して、セキュリティ アソシエーション（SA）を設定する前にピア デバイスをスケーラブルに認証できます。

デジタル シグニチャがない場合、ユーザは、IPSec を使用するデバイスの各ペア間で公開キーまたはシークレットを手動で交換して、通信を保護する必要があります。 証明書がない場合、ネットワークに新しいデバイスが追加されるたびに、安全に通信を行う他のすべてのデバイスで設定を変更する必要があります。 デジタル証明書がある場合、各デバイスは、CA に登録されます。 2 台のデバイスが通信する場合、証明書を交換し、データをデジタル署名して、お互いを認証します。 新しいデバイスがネットワークに追加されると、ユーザは、そのデバイスを CA に登録します。他のデバイスでは変更の必要はありません。 新しいデバイスが IPSec 接続を試行すると、証明書が自動的に交換され、デバイスを認証できます。

Secure Socket Layer の実装方法

HTTP サーバやオブジェクト リクエスト ブローカ（ORB）サーバなど、任意のアプリケーションで SSL を使用できるように設定するには、次の項で説明されている作業を実行します。

Secure Socket Layer の設定

ここでは、SSL の設定方法について説明します。

手順の概要

1. **crypto key generate rsa** [**usage-keys** | **general-keys**] [*keypair-label*]
2. configure
3. **domain ipv4 hos t** *host-name v4address1* [*v4address2...v4address8*] [**unicast** | **multicast**]
4. **crypto ca trustpoint** *ca-name*
5. **enrollment url CA-URL**
6. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
7. RP/0/RSP0/CPU0:router**crypto ca authenticate** *ca-name*
8. **crypto ca enroll** *ca-name*
9. show crypto ca certificates

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**crypto key generate rsa** [**usage-keys** | **general-keys**] [*keypair-label*]
	
	例：
	`RP/0/RSP0/CPU0:router# crypto key generate rsa general-keys`
	`The name for the keys will be: the_default`
	`% You already have keys defined for the_default`
	`Do you really want to replace them? [yes/no]:`
		RSA キー ペアを生成します。
		• RSA キーペアはインターネット キー交換（IKE）キー管理メッセージの署名および暗号化に使用されます。また、ルータの証明書を取得する際に必要です。
		• **usage-keys** キーワードを使用して、特定目的のキーを指定します。**general-keys** キーワードを使用して、汎用 RSA キーを指定します。
		• *keypair-label* 引数は、RSA キー ペアを指定する RSA キー ペア ラベルです。
		• DSA キーペアを生成するには、EXEC モードで **crypto key generate dsa** コマンドを使用します。

ステップ 2	configure 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**domain ipv4 hos t** *host-name v4address1* [*v4address2...v4address8*] [**unicast** | **multicast**] 例： `RP/0/RSP0/CPU0:router(config)# domain ipv4 host ultra5 192.168.7.18`	ホスト名とアドレスのスタティック マッピングを IPv4 を使用してホスト キャッシュに定義します。

	コマンドまたはアクション	目的
ステップ 4	**crypto ca trustpoint** *ca-name* 例： `RP/0/RSP0/CPU0:router(config)# crypto ca trustpoint myca`	ルータがピアに対して発行された証明書を確認できるように、選択した名前で信頼できるポイントを設定します。 • トラストポイント コンフィギュレーション モードを開始します。
ステップ 5	**enrollment url CA-URL** 例： `RP/0/RSP0/CPU0:router(config-trustp)# enrollment url http://ca.domain.com/certsrv/mscep/mscep.dll`	CA の URL を指定します。 • URL には、非標準 cgi-bin スクリプトの場所が含まれている必要があります。

ステップ 6	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ステップ 7	RP/0/RSP0/CPU0:router**crypto ca authenticate** *ca-name* 例： `RP/0/RSP0/CPU0:router# crypto ca authenticate myca`	このコマンドは、CA の公開キーを含む CA 証明書を取得することで、ルータに対して CA を認証します。 • 確認の画面が表示されたら、「**y**」を入力して証明書を承認します。

	コマンドまたはアクション	目的
ステップ 8	**crypto ca enroll** *ca-name* 例： `RP/0/RSP0/CPU0:router# crypto ca enroll myca`	すべての RSA キー ペアの証明書を要求します。 • このコマンドでは、ルータは存在する RSA キー ペアと同数の証明書を要求するため、特定目的の RSA キー ペアがある場合にも、このコマンドは 1 回しか実行する必要はありません。 • このコマンドでは、設定に保存されないチャレンジ パスワードを作成する必要があります。 証明書を失効させる必要が生じた場合、このパスワードが要求されるので、このパスワードを覚えておく必要があります。 • 証明書はすぐに発行できます。または、登録リトライ時間に達し、タイムアウトが発生するまで、ルータが証明書要求を毎分送信します。 タイムアウトが発生した場合、システム管理者に要求承認を依頼して、このコマンドを再入力します。 • **show crypto ca certificates** コマンドを使用して、証明書が許可されていることを確認します。
ステップ 9	show crypto ca certificates 例： `RP/0/RSP0/CPU0:router# show crypto ca certificates`	証明書と CA 証明書に関する情報を表示します。

Secure Socket Layer の実装の設定例

この項では、次の設定例について説明します。

Secure Socket Layer の設定：例

次に、ルータの RSA キーの生成、トラストポイントの設定、CA サーバの認証、キーに対する CA からの証明書の取得、および証明書に関する情報の表示の例を示します。

```
crypto key generate rsa general-keys commit configure domain ipv4 host
           xyz-ultra5 10.0.0.5 crypto ca trustpoint myca enrollment url http://xyz-ultra5
end
           crypto ca authenticate myca crypto ca enroll myca show crypto ca certificates
```

参考資料

ここでは、SSL の実装に関する関連資料について説明します。

関連資料

関連項目	ドキュメント名
PKI コマンド：コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』の「*Public Key Infrastructure Commands on Cisco ASR 9000* シリーズ ルータ」モジュール
SSL コマンド：コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference*』の「*Secure Socket Layer Protocol Commands on* Cisco ASR 9000 シリーズ ルータ」モジュール

標準

標準	タイトル
この機能でサポートが追加または変更された標準はありません。また、この機能で変更された既存の標準のサポートはありません。	—

MIB

MIB	MIB リンク
—	Cisco IOS XR ソフトウェアを使用して MIB を検出およびダウンロードするには、URL（http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml）にある Cisco MIB Locator を使用して、[Cisco Access Products] メニューでプラットフォームを選択します。

RFC

RFC	タイトル
RFC 2246	『The TLS Protocol, Version 1』、T. Dierks, C. Allen. 1999 年 1 月。

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトには、数千ページに及ぶ検索可能な技術情報があります。製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクもあります。 Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	http://www.cisco.com/en/US/support/index.html

第 10 章

レイヤ 2 セキュリティ機能

このモジュールでは、レイヤ 2 サービスのセキュリティ機能の概要について説明します。 すべてのレイヤ 2 セキュリティ機能は、VPLS ブリッジ ドメイン レベルで設定する必要があります。

レイヤ 2 VPLS ブリッジ ドメインのセキュリティ機能

次の表に、レイヤ 2 VPLS ブリッジ ドメインのセキュリティ機能の一覧と、各機能の詳細な設定マニュアルを示します。

表 *6*：レイヤ *2 VPN* のセキュリティ機能

機能	ドキュメント名
VPLS ブリッジ ドメインでの MAC アドレスベースのトラフィックのブロッキング、フィルタリング、および制限	『*Cisco ASR 9000* シリーズ アグリゲーション サービス ルータ *MPLS* 設定ガイド』の「Implementing Virtual Private LAN Services on Cisco ASR 9000 Series Routers」モジュールの「Configuring the MAC Address-related Parameters」の項を参照してください。
VPLS ブリッジ ドメインでのトラフィック ストーム制御	『*Cisco ASR 9000 Series Aggregation Services Router System Security Configuration Guide*』（このマニュアル）の「Cisco ASR 9000 Series Router の VPLS ブリッジでのトラフィック ストーム制御の実装」モジュールを参照してください。

機能	ドキュメント名
VPLS ブリッジ ドメインでの DHCP スヌーピング	『*Cisco ASR 9000 Series Aggregation Services Router IP Addresses and Services Configuration Guide*』の「Implementing DHCP on Cisco ASR 9000 Series Routers」モジュールを参照してください。このモジュールでは、レイヤ 2 での DHCP リレー サービスと DHCP スヌーピングの両方が説明されています。
VPLS ブリッジ ドメインでの IGMP スヌーピング	『*Cisco ASR 9000 Series Aggregation Services Router Multicast Configuration Guide*』の「Implementing Layer 2 Multicast with IGMP Snooping」モジュールを参照してください。

第 11 章

VPLSブリッジでのトラフィックストーム制御の実装

トラフィックストーム制御では、過剰なトラフィックによるブリッジの遮断を防止することで、バーチャルプライベートLANサービス（VPLS）ブリッジにおけるレイヤ2ポートセキュリティを提供します。 このモジュールでは、トラフィック ストーム制御の実装方法について説明します。

トラフィック ストーム制御の機能履歴

リリース	変更点
リリース 3.7.2	VPLS ブリッジにおける接続回線（AC）とアクセス疑似回線（PW）のトラフィック ストーム制御が追加されました。

トラフィック ストーム制御の実装に関する前提条件

トラフィック ストーム制御を実装する前に、次の前提条件を満たす必要があります。

- MPLS レイヤ 2 VPN の VPLS ブリッジ ドメインによって、ネットワークが設定されている必要があります。

- 適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。

トラフィック ストーム制御の実装に関する制約事項

Cisco IOS XR ソフトウェア リリース 3.7.0 FCI では、次の制約事項が適用されます。

- ブリッジ ドメインへの直接のトラフィック ストーム制御はサポートされていません。ブリッジ ドメイン サブモードを使用して、ブリッジ ドメイン配下の Ethernet Flow Points（EFP）に機能を設定する必要があります。 AC およびアクセス PW の設定に使用されるサブモードがサポートされています。
- トラフィック ストーム制御は、集約 EFP（バンドル）ではサポートされていません。
- トラフィック ストーム制御は、転送疑似回線（VFI PW）ではサポートされていません。
- ルート スイッチ プロセッサ（RSP）のフェールオーバー直後は、トラフィック ストーム制御のドロップ カウンタが正確でない場合があります。 このフェールオーバー後のカウンタ情報の損失は、Cisco IOS XR ソフトウェア カウンタでは正常な動作です。
- パケットがドロップされた際、アラームは生成されません。

トラフィック ストーム制御の実装について

トラフィック ストーム制御を実装するには、次の概念について理解しておく必要があります。

トラフィック ストーム制御について

トラフィック ストームは、パケットが VPLS ブリッジでフラッディングする場合に発生するもので、過剰なトラフィックを生成し、ネットワークのパフォーマンスを低下させます。 トラフィック ストーム制御では、パケット数が設定されたしきい値レベルに達したときに、トラフィックを抑制することで VPLS ブリッジの遮断を防止します。 VPLS ブリッジ配下の各ポートでは、さまざまなタイプのトラフィックに対して個別のしきい値レベルを設定できます。

トラフィック ストーム制御では、ポート上の着信トラフィック レベルが監視され、1 秒のインターバルのうちにパケット数が設定したしきい値レベルに到達すると、トラフィックがドロップされます。 この 1 秒のインターバルは、ハードウェアに設定されるため、変更できません。 1 秒のインターバルの間に通過を許可するパケット数は、ポート別、トラフィック タイプ別に設定可能です。

しきい値は、1 秒あたりのパケット数のレートを使用して設定されます。 指定されたトラフィック タイプのパケット数がポートのしきい値レベルに到達すると、そのポートは 1 秒のインターバルの残り時間がなくなるまで、そのトラフィック タイプの新たなパケットをすべてドロップしま

す。 新しい 1 秒のインターバルが開始されると、その指定されたタイプのトラフィックはポートを通過できるようになります。

トラフィック ストーム制御はルータのパフォーマンスにほとんど影響を与えません。ポートを通過するパケットは、この機能がイネーブルになっているかどうかに関係なくカウントされます。新たなカウントは、ドロップされたパケットをモニタするドロップ カウンタのみで発生します。

パケットがドロップされた際、アラームは生成されません。

トラフィック ストーム制御のデフォルト

- トラフィック ストーム制御機能は、デフォルトではディセーブルに設定されています。 各トラフィック タイプに対して、各ポートでこの機能を明示的にイネーブルにする必要があります。
- トラフィック ストーム制御のモニタリング インターバルは、ハードウェアに設定されるため、変更できません。 Cisco ASR 9000 Series Router では、モニタリング インターバルは常に 1 秒です。

トラフィック ストーム制御でサポートされるトラフィック タイプ

各 VPLS ブリッジ ポートで、サポートされているトラフィック タイプにそれぞれ対応する、最大 3 つのストーム制御しきい値を設定できます。 トラフィック タイプに対してしきい値を設定しない場合、そのポートまたはインターフェイスでは、そのトラフィック タイプに対してトラフィック ストーム制御がイネーブルになりません。

サポートされているトラフィック タイプは次のとおりです。

- ブロードキャスト トラフィック：宛先 MAC アドレスが FFFF.FFFF.FFFF になっているパケット。
- マルチキャスト トラフィック：宛先 MAC アドレスがブロードキャスト アドレスではなく、マルチキャスト ビットが 1 に設定されているパケット。 マルチキャスト ビットは MAC アドレスの最も重要なバイトのビット 0 です。
- 不明なユニキャスト トラフィック：宛先 MAC アドレスがまだ学習されていないパケット。

トラフィック ストーム制御は、Bridge Protocol Data Unit（BPDU; ブリッジ プロトコル データ ユニット）パケットには適用されません。すべての BPDU パケットは、トラフィック ストーム制御が設定されていないものとして処理されます。

トラフィック ストーム制御でサポートされるポート

Cisco IOS XR ソフトウェア リリース 3.7.0 FCI では、VPLS ブリッジ ドメイン配下の次のコンポーネントで、トラフィック ストーム制御を設定できます。

- VPLS ブリッジ ドメインの AC

- VPLS ブリッジ ドメインのアクセス PW

トラフィック ストーム制御のしきい値

トラフィック ストーム制御のしきい値は、1 秒あたりのパケット数のレートで設定されます。 しきい値は、ポートで 1 秒のインターバルの間に通過できる、指定されたトラフィック タイプのパケット数です。 トラフィック ストーム制御のしきい値の有効値は、1 ～ 160000 の整数です。 最大値では、10 Gbps リンクで 1 秒あたりに帯域幅の約 19 % の通過を許可します（パケット サイズは 1500 バイトを想定）。

トラフィック ストーム制御ドロップ カウンタ

トラフィック ストーム制御では、ポートおよびトラフィック タイプ別にドロップされたパケット数をカウントします。 ドロップ カウンタは、明示的にクリアしない限り、累積されます。 ドロップ カウントを表示するには、**show l2vpn bridge-domain detail** コマンドおよび **show l2vpn forwarding detail** コマンドを使用します。 ドロップ カウンタをクリアするには、**clear l2vpn forwarding counters** コマンドを使用します。

トラフィック ストーム制御の設定方法

ここでは、トラフィック ストーム制御を設定する方法について説明します。

ブリッジの AC でのトラフィック ストーム制御のイネーブル化

VPLS ブリッジ配下の AC でトラフィック ストーム制御をイネーブルにするには、次の作業を実行します。 次の作業では、イーサネット インターフェイス上の VLAN である AC でトラフィック ストーム制御をイネーブルにする方法について説明します。

（注） トラフィック ストーム制御をディセーブルにするには、機能をイネーブルにしたサブモードにナビゲートして、このコマンドの **no** 形式を使用します。

手順の概要

1. **configure**
2. l2vpn
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **interface** *interface-name*
6. **storm-control** {**broadcast** | **multicast** | **unknown-unicast**} **pps** *packet-threshold*
7. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
8. **show l2vpn bridge-domain bd-name** *bridge-name* **detail**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

ステップ 2	l2vpn
	
	例：
	
	`RP/0/0/CPU0:router(config)# l2vpn`
	`RP/0/0/CPU0:router(config-l2vpn)#`
		L2 VPN コンフィギュレーション モードを開始します。

ステップ 3	**bridge group** *bridge-group-name*
	
	例：
	
	`RP/0/0/CPU0:router(config-l2vpn)# bridge group csco`
	`RP/0/0/CPU0:router(config-l2vpn-bg)#`
		L2 VPN ブリッジ グループ コンフィギュレーション モードを開始します。

ステップ 4	**bridge-domain** *bridge-domain-name*
	
	例：
	
	`RP/0/0/CPU0:router(config-l2vpn-bg)# bridge-domain abc`
	`RP/0/0/CPU0:router(config-l2vpn-bg-bd)#`
		L2 VPN ブリッジ ドメイン コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的

ステップ 5	**interface** *interface-name*
	
	例：
	`RP/0/0/CPU0:router(config-l2vpn-bg-bd)# interface GigabitEthernet0/1/0/0.100`
	`RP/0/0/CPU0:router(config-l2vpn-bg-bd-ac)#`
		ブリッジ ドメイン配下の AC を指定します。 この場合、AC はイーサネット インターフェイス上の VLAN です。

ステップ 6	**storm-control** {**broadcast** | **multicast** | **unknown-unicast**} **pps** *packet-threshold*
	
	例：
	`RP/0/0/CPU0:router(config-l2vpn-bg-bd-ac)# storm-control broadcast pps 4500`
	`RP/0/0/CPU0:router(config-l2vpn-bg-bd-ac)# storm-control multicast pps 500`
	`RP/0/0/CPU0:router(config-l2vpn-bg-bd-ac)#`
		指定されたトラフィック タイプに対して、このインターフェイスでトラフィック ストーム制御をイネーブルにします。 このコマンドを各トラフィック タイプに対して繰り返します。
		
		packet-threshold は、1 秒あたりのパケット数のレートで、値は 1 ～ 160000 の整数です。 1 秒のインターバルの間に、指定されたトラフィック タイプのインターフェイスで通過を許可するパケット数を指定します。

ステップ 7	次のいずれかのコマンドを使用します。
	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。
		◦ no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。
		◦ cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。
		• 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ステップ 8	**show l2vpn bridge-domain bd-name** *bridge-name* **detail**	ストーム制御設定を表示します。

	コマンドまたはアクション	目的
	例： `RP/0/0/CPU0:router# show l2vpn bridge-domain bd-name abc detail`	

ブリッジの PW でのトラフィック ストーム制御のイネーブル化

VPLS ブリッジ配下の疑似回線でトラフィック ストーム制御をイネーブルにするには、次の作業を実行します。

（注） トラフィック ストーム制御をディセーブルにするには、機能をイネーブルにしたサブモードにナビゲートして、このコマンドの **no** 形式を使用します。

手順の概要

1. **configure**
2. l2vpn
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **neighbor** *address* **pw-id** *id*
6. **storm-control** {**broadcast** | **multicast** | **unknown-unicast**} **pps** *packet-threshold*
7. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
8. **show l2vpn bridge-domain bd-name** *bridge-name* **detail**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的

ステップ 2	l2vpn
	
	例：
	
	`RP/0/0/CPU0:router(config)# l2vpn`
	`RP/0/0/CPU0:router(config-l2vpn)#`
		L2 VPN コンフィギュレーション モードを開始します。

ステップ 3	**bridge group** *bridge-group-name*
	
	例：
	
	`RP/0/0/CPU0:router(config-l2vpn)# bridge group csco`
	`RP/0/0/CPU0:router(config-l2vpn-bg)#`
		L2 VPN ブリッジ グループ コンフィギュレーション モードを開始します。

ステップ 4	**bridge-domain** *bridge-domain-name*
	
	例：
	
	`RP/0/0/CPU0:router(config-l2vpn-bg)# bridge-domain abc`
	`RP/0/0/CPU0:router(config-l2vpn-bg-bd)#`
		L2 VPN ブリッジ ドメイン コンフィギュレーション モードを開始します。

ステップ 5	**neighbor** *address* **pw-id** *id*
	
	例：
	
	`RP/0/0/CPU0:router(config-l2vpn-bg-bd)# neighbor 1.1.1.1 pw-id 100`
	`RP/0/0/CPU0:router(config-l2vpn-bg-bd-pw)#`
		ブリッジ ドメイン配下のアクセス疑似回線を指定します。
		
		（注） ストーム制御を転送 PW（VFI 配下の PW）に適用することはできません。

ステップ 6	**storm-control** {**broadcast** | **multicast** | **unknown-unicast**} **pps** *packet-threshold*
	
	例：
	
	`RP/0/0/CPU0:router(config-l2vpn-bg-bd-pw)# storm-control broadcast pps 4500`
	`RP/0/0/CPU0:router(config-l2vpn-bg-bd-pw)# storm-control multicast pps 500`
	`RP/0/0/CPU0:router(config-l2vpn-bg-bd-pw)#`
		指定されたトラフィック タイプに対して、この疑似回線でトラフィック ストーム制御をイネーブルにします。 このコマンドを各トラフィック タイプに対して繰り返します。
		
		packet-threshold は、1 秒あたりのパケット数のレートで、値は 1 ～ 160000 の整数です。 1 秒のインターバルの間に、指定されたトラフィック タイプのインターフェイスで通過を許可するパケット数を指定します。

ステップ 7	次のいずれかのコマンドを使用します。
	
	• **end**
	• **commit**
	
	例：
	
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		
		◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。

	コマンドまたはアクション	目的
		◦ no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 ◦ cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 8	**show l2vpn bridge-domain bd-name** *bridge-name* **detail** 例： `RP/0/0/CPU0:router# show l2vpn bridge-domain bd-name csco detail`	指定されたブリッジドメインのストーム制御設定を表示します。このコマンドでは、設定された各ストーム制御インスタンスのドロップ カウンタ値も表示されます。

トラフィック ストーム制御ドロップ カウンタのクリア

トラフィック ストーム制御ドロップ カウンタをゼロにリセットするには、次の作業を実行します。

手順の概要

1. clear l2vpn forwarding counters

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	clear l2vpn forwarding counters 例： `RP/0/0/CPU0:router# clear l2vpn forwarding counters`	ストーム制御ドロップ カウンタを含む、L2VPN 転送カウンタをクリアします。

トラフィック ストーム制御の設定例

ここでは、次の設定例を示します。

AC でのトラフィック ストーム制御の設定：例

次に、VPLS ブリッジの AC でのブロードキャストおよびマルチキャスト ストーム制御設定の例を示します。

```
RP/0/RSP0/CPU0:router# show run

[lines deleted]

bridge group 215
  bridge-domain 215
   mtu 9000
   interface GigabitEthernet0/1/0/3.215
    storm-control multicast pps 500
    storm-control broadcast pps 4500
   !
[lines deleted]


RP/0/RSP0/CPU0:router# show l2vpn bridge-domain bd-name 215 detail
Bridge group: 215, bridge-domain: 215, id: 3, state: up, ShgId: 0, MSTi: 0
  MAC learning: enabled
  MAC withdraw: disabled
  Flooding:
    Broadcast & Multicast: enabled
    Unknown unicast: enabled
  MAC aging time: 300 s, Type: inactivity
  MAC limit: 4000, Action: none, Notification: syslog
  MAC limit reached: no
  Security: disabled
  Split Horizon Group: none
  DHCPv4 snooping: disabled
  IGMP Snooping profile: none
  Bridge MTU: 9000
  Filter MAC addresses:
  ACs: 2 (2 up), VFIs: 1, PWs: 1 (1 up)
  List of ACs:
    AC: GigabitEthernet0/1/0/3.215, state is up
      Type VLAN; Num Ranges: 1
     vlan ranges: [100, 100]
      MTU 9008; XC ID 0x440005; interworking none; MSTi 0 (unprotected)
      MAC learning: enabled
      Flooding:
        Broadcast & Multicast: enabled
        Unknown unicast: enabled
      MAC aging time: 300 s, Type: inactivity
      MAC limit: 4000, Action: none, Notification: syslog
      MAC limit reached: no
      Security: disabled
      Split Horizon Group: none
      DHCPv4 snooping: disabled
      IGMP Snooping profile: none

      Storm Control:
       Broadcast: enabled(4500)
       Multicast: enabled(500)
       Unknown unicast: disabled
      Static MAC addresses:
      Statistics:
        packet totals: receive 36728, send 31
```

```
        byte totals: receive 2791284, send 2318
        Storm control drop counters:
          packet totals: broadcast 0, multicast 0, unknown unicast 0
          byte totals: broadcast 0, multicast 0, unknown unicast 0
[lines deleted]
```

アクセス PW でのトラフィック ストーム制御の設定：例

次に、VPLS ブリッジのアクセス PW でのブロードキャストおよびマルチキャスト ストーム制御設定の例を示します。

```
RP/0/RSP0/CPU0:router# show run
l2vpn
 bridge group bg_storm_pw
  bridge-domain bd_storm_pw
   interface Bundle-Ether101
   !
   neighbor 10.10.30.30 pw-id 1
    storm-control unknown-unicast pps 120
    storm-control multicast pps 110
    storm-control broadcast pps 100
   !
  !
 !
!

RP/0/RSP0/CPU0:router# show l2vpn bridge-domain group bg_storm_pw detail
Bridge group: bg_storm_pw, bridge-domain: bd_storm_pw, id: 2, state: up, ShgId: 0, MSTi: 0
  MAC learning: enabled
  MAC withdraw: disabled
  Flooding:
    Broadcast & Multicast: enabled
    Unknown unicast: enabled
  MAC aging time: 300 s, Type: inactivity
  MAC limit: 4000, Action: none, Notification: syslog
  MAC limit reached: no
  Security: disabled
  Split Horizon Group: none
  DHCPv4 snooping: disabled
  IGMP Snooping profile: none
  Bridge MTU: 1500
  Filter MAC addresses:
  ACs: 1 (1 up), VFIs: 0, PWs: 1 (1 up)
  List of ACs:
    AC: Bundle-Ether101, state is up
      Type Ethernet
      MTU 1500; XC ID 0xfffc0003; interworking none
      MAC learning: enabled
      Flooding:
        Broadcast & Multicast: enabled
        Unknown unicast: enabled
      MAC aging time: 300 s, Type: inactivity
      MAC limit: 4000, Action: none, Notification: syslog
      MAC limit reached: no
      Security: disabled
      Split Horizon Group: none
      DHCPv4 snooping: disabled
      IGMP Snooping profile: none
      Storm Control: disabled
      Static MAC addresses:
      Statistics:
        packets: received 0, sent 5205
        bytes: received 0, sent 645420
      Storm control drop counters:
        packets: broadcast 0, multicast 0, unknown unicast 0
        bytes: broadcast 0, multicast 0, unknown unicast 0
  List of Access PWs:
    PW: neighbor 10.10.30.30, PW ID 1, state is up ( established )
```

```
  PW class not set, XC ID 0xfffc0006
  Encapsulation MPLS, protocol LDP
  PW type Ethernet, control word disabled, interworking none
  PW backup disable delay 0 sec
  Sequencing not set
PW Status TLV in use
    MPLS         Local                          Remote
    ------------ ------------------------------ ------------------------------
    Label        16001                          16001
    Group ID     0x2                            0x2
    Interface    Access PW                      Access PW
    MTU          1500                           1500
    Control word disabled                       disabled
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x2                            0x2
                 (LSP ping verification)        (LSP ping verification)
    VCCV CC type 0x6                            0x6
                 (router alert label)           (router alert label)
                 (TTL expiry)                   (TTL expiry)
    ------------ ------------------------------ ------------------------------
  Incoming Status (PW Status TLV):
    Status code: 0x0 (Up) in Notification message
  Outgoing Status (PW Status TLV):
    Status code: 0x0 (Up) in Notification message
  Create time: 16/12/2008 00:06:08 (01:00:22 ago)
  Last time status changed: 16/12/2008 00:35:02 (00:31:28 ago)
    MAC withdraw message: send 0 receive 0
  Static MAC addresses:
  Statistics:
    packets: received 0, sent 0
    bytes: received 0, sent 0
  Storm control drop counters:
    packets: broadcast 0, multicast 0, unknown unicast 0
    bytes: broadcast 0, multicast 0, unknown unicast 0
  MAC learning: enabled
  Flooding:
    Broadcast & Multicast: enabled
    Unknown unicast: enabled
  MAC aging time: 300 s, Type: inactivity
  MAC limit: 4000, Action: none, Notification: syslog
  MAC limit reached: no
  Security: disabled
  Split Horizon Group: none
  DHCPv4 snooping: disabled
  IGMP Snooping profile: none
  Storm Control:
   Broadcast: enabled(100)
   Multicast: enabled(110)
   Unknown unicast: enabled(120)
```

参考資料

トラフィック ストーム制御の実装に関する詳細情報については、次の参考資料を参照してください。

関連資料

関連項目	ドキュメント名
MPLS レイヤ 2 VPN	『*Cisco ASR 9000* シリーズ アグリゲーション サービス ルータ *MPLS* 設定ガイド』の「*Implementing MPLS Layer 2 VPNs on* Cisco ASR 9000 Series Router」モジュール

関連項目	ドキュメント名
MPLS VPLS ブリッジ	『*Cisco ASR 9000* シリーズ アグリゲーション サービス ルータ *MPLS* 設定ガイド』の「*Implementing Virtual Private LAN Services on* Cisco ASR 9000 Series Router」モジュール
スタートアップ資料	『*Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide*』

標準

標準[1]	タイトル
この機能でサポートが追加または変更された標準はありません。また、この機能で変更された既存の標準のサポートはありません。	—

1 サポートされている規格がすべて記載されているわけではありません。

MIB

MIB	MIB リンク
—	Cisco IOS XR ソフトウェアを使用して MIB を検出およびダウンロードするには、URL（http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml）にある Cisco MIB Locator を使用して、[Cisco Access Products] メニューでプラットフォームを選択します。

RFC

RFC	タイトル
新しい RFC または変更された RFC はサポートされていません。また、既存の RFC に対するサポートに変更はありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトには、数千ページに及ぶ検索可能な技術情報があります。製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクもあります。 Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	http://www.cisco.com/en/US/support/index.html

索 引

記号

A

C

D

I

K

M

O

P

R

S

T

U

V

X

Z

あ

い

お

か

き

く

け

こ

さ

し

せ

た

ち

つ

て

と

な

に

ね

は

ひ

ふ

ほ

ま

め

も

ゆ

よ

ら

り

ろ