RadSec の設定

この章では、RadSec over Transport Layer Security（TLS）および Datagram Transport Layer Security（DTLS）サーバーを設定する方法について説明します。

RadSec の設定に関する制限事項

RadSec 機能には、次のような制限事項が適用されます。

- RADIUS クライアントは、エフェメラルポートを送信元ポートとして使用します。この送信元ポートは、UDP、Datagram Transport Layer Security（DTLS）、および Transport Layer Security（TLS）に同時に使用できません。
- 設定の制限はありませんが、AAA サーバーグループ下のサーバーに同じタイプ（TLS のみまたは DTLS のみ）を使用することを推奨します。
- RadSec は、1 ～ 1024 の DTLS ポート範囲ではサポートされていません。

  DTLS ポートは、Radius サーバーと連携するように設定する必要があります。
- RadSec は、高可用性ではサポートされていません。
- 同じ認証チャネルを介した要求の RADIUS 許可変更（CoA）受信と応答の送信は、RadSec over TLS でのみサポートされます。DTLS またはプレーン RADIUS ではサポートされません。
- **tls watchdoginterval** 機能は、Packet of Disconnect（POD）の使用例には適用されません。
- CoA の FQDN 設定はサポートされていません。

RadSec に関する情報

RadSec は、安全なトンネルを介して転送される RADIUS サーバー上で暗号化サービスを提供します。RadSec over TLS および DTLS は、クライアントサーバとデバイスサーバーの両方に実装されています。クライアント側が RADIUS AAA を制御しているのに対し、デバイス側は CoA を制御します。

次のパラメータを設定できます:

- 個々のクライアント固有のアイドルタイムアウト、クライアントトラストポイント、およびサーバートラストポイント。
- グローバル CoA 固有の TLS または DTLS リスニングポートおよび対応するソースインターフェイスのリスト。

（注） 特定のサーバーに対して TLS または DTLS を無効にするには、RADIUS サーバーの設定モードで **no tls** または **no dtls** コマンドを使用します。

同じ認証チャネルを介した RadSec CoA 要求の受信と CoA 応答の送信を有効にするには、**tls watchdoginterval** コマンドを設定します。アイドルタイマーの期限が切れる前に RADIUS テスト認証パケットが確認された場合に、確立されたトンネルがアクティブなままになるように、TLS ウォッチドッグタイマーは TLS アイドルタイマーよりも小さくする必要があります。トンネルが切断され、**tls watchdoginterval** コマンドが有効になっている場合、トンネルはすぐに再確立されます。**tls watchdoginterval** コマンドが無効になっている場合、同じ認証チャネル上の CoA 要求は廃棄されます。

RadSec の設定方法

次のセクションでは、RadSec の設定を構成するさまざまな作業について説明します。

RadSec over TLS の設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** **例**： `Device> enable`	特権 EXEC モードを有効にします。 • プロンプトが表示されたらパスワードを入力します。

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device# configure terminal`	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	**radius server** *radius-server-name* 例： `Device(config)# radius server R1`	RADIUS サーバー設定の名前を Protected Access Credential（PAC）のプロビジョニング用に指定し、RADIUS サーバー設定モードを開始します。

ステップ 4	**tls** [**connectiontimeout** *connection-timeout-value*] [**idletimeout** *idle-timeout-value*] [[**ip | ipv6**] {**radius source-interface** *interface-name* |**vrf forwarding** *forwarding-table-name*}] [**match-server-identity** {**email-address** *email-address* | **hostname** *host-name* | **ip-address** *ip-address*}] [**port** *port-number*] [**retries** *number-of-connection-retries*] [**trustpoint** {**client** *trustpoint name* | **server** *trustpoint name*}] [**watchdoginterval** *interval*]
	例：
	`Device(config-radius-server)# tls connectiontimeout 10`
	`Device(config-radius-server)# tls idletimeout 75`
	`Device(config-radius-server)# tls retries 15`
	`Device(config-radius-server)# tls ip radius source-interface GigabitEthernet 1/0/1`
	`Device(config-radius-server)# tls ipv6 vrf forwarding table-1`
	`Device(config-radius-server)# tls match-server-identity ip-address 10.1.1.10`
	`Device(config-radius-server)# tls port 10`
	`Device(config-radius-server)# tls trustpoint client TP-self-signed-721943660`
	`Device(config-radius-server)# tls trustpoint server isetp`
	`Device(config-radius-server)# tls watchdoginterval 10`
		TLS パラメータを設定します。次のパラメータを設定できます:
		• **connectiontimeout**：TLS 接続タイムアウト値を設定します。デフォルトは 5 秒です。
		• **idletimeout**：TLS アイドルタイムアウト値を設定します。デフォルトは 60 秒です。
		• **ip**：IP 送信元パラメータを設定します。
		• **ipv6**：IPv6 送信元パラメータを設定します。
		• **match-server-identity**：RadSec 認定検証パラメータを設定します。
		（注） この設定は必須です。
		• **port**：TLS ポート番号を設定します。デフォルトは 2083 です。
		• **retries**：TLS 接続再試行の回数を設定します。デフォルトは 5 分です。
		• **trustpoint**：ライアントとサーバーに TLS トラストポイントを設定します。クライアントとサーバーの TLS トラストポイントが同じ場合、トラストポイント名も両方で同じである必要があります。
		• **watchdoginterval**：ウォッチドッグ間隔を設定します。これにより、同じ認証チャネルで CoA 要求を受信できるようになります。また、TLS

	コマンドまたはアクション	目的
		トンネルを維持するキープアライブとして機能し、トンネルが切断された場合にトンネルを再確立します。 （注） **watchdoginterval** 値は、確立されたトンネルがアップ状態を維持するために、**idletimeout** よりも小さい値である必要があります。
ステップ 5	**end** 例： `Device(config-radius-server)# end`	RADIUS サーバ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

TLS CoA の動的認可の設定

（注） **tls watchdoginterval** コマンドを有効にすると、**aaa server radius dynamic-author** のクライアント IP 設定は使用されません。代わりに、**radius server** で設定されたキーが CoA トランザクションに使用されます。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	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	**aaa server radius dynamic-author** 例： `Device(config)# aaa server radius dynamic-author`	ダイナミック認証ローカル サーバー コンフィギュレーションモードを入力し、デバイスが認可変更（CoA）を受け入れ、要求を取り外す RADIUS クライアントを指定します。デバイスを AAA サーバーとして設定し、外部ポリシー サーバーとの連携を促進します。

	コマンドまたはアクション	目的
ステップ 4	**client {ip-addr | hostname}** [**tls** [**client-tp** *client-tp-name*] [**idletimeout** *idletimeout-interval*] [**server-key** *server-key*] [**server-tp** *server-tp-name*]] 例： `Device(config-locsvr-da-radius)# client 10.104.49.14 tls idletimeout 100 client-tp tls_ise server-tp tls_client server-key key1`	AAA サーバー クライアントの IP アドレスまたはホスト名を設定します。次のオプションのパラメータを設定できます。 • **tls**：クライアントの TLS を有効にします。 • **client-tp**：クライアント トラストポイントを設定します。 • **idletimeout**：TLS アイドルタイムアウト値を設定します。 • **server-key**：RADIUS クライアントサーバーキーを設定します。 • **server-tp**：サーバートラストポイントを設定します。
ステップ 5	**end** 例： `Device(config-locsvr-da-radius)# end`	ダイナミック認証ローカル サーバー コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

RadSec over DTLS の設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	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	**radius server** *radius-server-name* 例： `Device(config)# radius server R1`	RADIUS サーバー設定の名前を Protected Access Credential（PAC）のプロビジョニング用に指定し、RADIUS サーバー設定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 4	**dtls** [**connectiontimeout** *connection-timeout-value*] [**idletimeout** *idle-timeout-value*] [[**ip** | **ipv6**] {**radius source-interface** *interface-name* |**vrf forwarding** *forwarding-table-name*}] [**match-server-identity** {**email-address** *email-address* | **hostname** *host-name* | **ip-address** *ip-address*}] [**port** *port-number*] [**retries** *number-of-connection-retries*] [**trustpoint** {**client** *trustpoint name* | **server** *trustpoint name*}]
	例：
	`Device(config-radius-server)# dtls connectiontimeout 10`
	`Device(config-radius-server)# dtls idletimeout 75`
	`Device(config-radius-server)# dtls retries 15`
	`Device(config-radius-server)# dtls ip radius source-interface GigabitEthernet 1/0/1`
	`Device(config-radius-server)# dtls ipv6 vrf forwarding table-1`
	`Device(config-radius-server)# tls match-server-identity ip-address 10.1.1.10`
	`Device(config-radius-server)# dtls port 10`
	`Device(config-radius-server)# dtls trustpoint client TP-self-signed-721943660`
	`Device(config-radius-server)# dtls trustpoint server isetp`
		DTLS パラメータを設定します。次のパラメータを設定できます。
		• **connectiontimeout**：DTLS 接続タイムアウト値を設定します。デフォルトは 5 秒です。
		• **idletimeout**：DTLS アイドルタイムアウト値を設定します。デフォルトは 60 秒です。
		（注） アイドルタイムアウトの期限が切れ、最後のアイドルタイムアウトの後にトランザクションがない場合、DTLS セッションは終了します。セッションが再確立されたら、アイドルタイマーを再起動して機能させます。
		設定されたアイドルタイムアウトが 30 秒である場合、タイムアウトが期限切れになると、RADIUS DTLS トランザクションの数がチェックされます。RADIUS DTLS パケットが 0 より大きい場合、トランザクションカウンタがリセットされ、タイマーが再開されます。
		• **ip**：IP 送信元パラメータを設定します。
		• **ipv6**：IPv6 送信元パラメータを設定します。
		• **match-server-identity**：RadSec 認定検証パラメータを設定します。
		（注） この設定は必須です。

	コマンドまたはアクション	目的
		• **port**：DTLS ポート番号を設定します。デフォルトは 2083 です。 • **retries**：DTLS 接続再試行の回数を設定します。デフォルトは 5 分です。 • **trustpoint**：クライアントとサーバーに DTLS トラストポイントを設定します。クライアントとサーバーの DTLS トラストポイントが同じ場合、トラストポイント名も両方で同じである必要があります。
ステップ 5	**end** 例： `Device(config-radius-server)# end`	RADIUS サーバ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

DTLS CoA の動的認可の設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	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	**aaa server radius dynamic-author** 例： `Device(config)# aaa server radius dynamic-author`	ダイナミック認可ローカル サーバー コンフィギュレーションモードを開始し、デバイスが認可変更（CoA）を受け入れ、要求を取り外す RADIUS クライアントを指定します。デバイスを AAA サーバーとして設定し、外部ポリシー サーバーとの連携を促進します。
ステップ 4	**client {ip-addr | hostname}** [**dtls** [**client-tp** *client-tp-name*] [**idletimeout** *idletimeout-interval*] [**server-key** *server-key*] [**server-tp** *server-tp-name*]]	AAA サーバー クライアントの IP アドレスまたはホスト名を設定します。次の

	コマンドまたはアクション	目的
	例： `Device(config-locsvr-da-radius)# client 10.104.49.14 dtls idletimeout 100 client-tp tls_ise server-tp tls_client server-key key1`	オプションのパラメータを設定できます。 • **tls**：クライアントの TLS を有効にします。 • **client-tp**：クライアントトラストポイントを設定します。 • **idletimeout**：TLS アイドルタイムアウト値を設定します。 • **server-key**：RADIUS クライアントサーバーキーを設定します。 • **server-tp**：サーバートラストポイントを設定します。

ステップ 5	**dtls** {{**ip** | **ipv6**} **radius source-interface** *interface-name* | **port** *radius-dtls-server-port-number*}
	例：
	`Device(config-locsvr-da-radius)# dtls ip radius source-interface GigabitEthernet 1/0/24`
	`Device(config-locsvr-da-radius)# dtls port 100`
		RADIUS CoA サーバーを設定します。次のパラメータを設定できます:
		• {**ip** | **ipv6**} **radius source-interface** *interface-name*：RADIUS CoA サーバーの送信元アドレスのインターフェイスを指定します。
		• **port** *radius-dtls-server-port-numbe*：ローカル DTLS RADIUS サーバーがリッスンするポートを指定します。

ステップ 6	**end** 例： `Device(config-locsvr-da-radius)# end`	ダイナミック認証ローカル サーバー コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

RadSec のモニタリング

次のコマンドを使用して、TLS および DTLS サーバーの統計を監視します。

表 *1: TLS* および *DTLS* サーバー統計コマンドの監視

コマンド	目的
show aaa servers	TLS および DTLS サーバーに関連する情報を表示します。

コマンド	目的
clear aaa counters servers radius {*server id* | **all**}	RADIUS TLS 固有または DTLS 固有の統計情報をクリアします。
debug radius radsec	RADIUS RadSec デバッグを有効にします。

RadSec の設定例

次の例は、RadSec の設定を理解するのに役立ちます。

例：RadSec over TLS の設定

次に、RadSec over TLS を設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# radius server R1
Device(config-radius-server)# tls connectiontimeout 10
Device(config-radius-server)# tls idletimeout 75
Device(config-radius-server)# tls retries 15
Device(config-radius-server)# tls ip radius source-interface GigabitEthernet 1/0/1
Device(config-radius-server)# tls ip vrf forwarding table-1
Device(config-radius-server)# tls port 10
Device(config-radius-server)# tls trustpoint client TP-self-signed-721943660
Device(config-radius-server)# tls trustpoint server isetp
Device(config-radius-server)# tls watchdoginterval 10
Device(config-radius-server)# end
```

例：TLS CoA の動的認可の設定

次の例は、TLS CoA の動的許可を設定する方法を示しています。

```
Device> enable
Device# configure terminal
Device(config)# aaa server radius dynamic-author
Device(config-locsvr-da-radius)# client 10.104.49.14 tls idletimeout 100
client-tp tls_ise server-tp tls_client
Device(config-locsvr-da-radius)# end
```

例：RadSec over DTLS の設定

次に、RadSec over DTLS を設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# radius server R1
Device(config-radius-server)# dtls connectiontimeout 10
Device(config-radius-server)# dtls idletimeout 75
Device(config-radius-server)# dtls retries 15
Device(config-radius-server)# dtls ip radius source-interface GigabitEthernet 1/0/1
```

```
Device(config-radius-server)# dtls ip vrf forwarding table-1
Device(config-radius-server)# dtls port 10
Device(config-radius-server)# dtls trustpoint client TP-self-signed-721943660
Device(config-radius-server)# dtls trustpoint server isetp
Device(config-radius-server)# end
```

例：DTLS CoA の動的認可の設定

次の例は、DTLS CoA の動的認証を設定する方法を示しています。

```
Device> enable
Device# configure terminal
Device(config)# aaa server radius dynamic-author
Device(config-locsvr-da-radius)# client 10.104.49.14 dtls idletimeout 100
client-tp dtls_ise server-tp dtls_client
Device(config-locsvr-da-radius)# dtls ip radius source-interface GigabitEthernet 1/0/24
Device(config-locsvr-da-radius)# dtls port 100
Device(config-locsvr-da-radius)# end
```

RadSec 設定の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Bengaluru 17.4.1	RadSec の設定	RadSec は、安全なトンネルを介して転送される RADIUS サーバー上で暗号化サービスを提供します。
Cisco IOS XE Bengaluru 17.6.1	同じトンネル上の RadSec CoA	RadSec CoA 要求の受信と CoA 応答の送信は、同じ認証チャネルを介して実行できます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。

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