RADIUS DTLS

RADIUS DTLS について

Remote Authentication Dial-In User Service（RADIUS）は、ネットワークへの管理アクセス権を取得しようとするユーザーに対して中央管理されたセキュリティ機能を提供する、クライアントまたはサーバープロトコルです。RADIUS プロトコルは広く導入されている認証および認可プロトコルであり、完全な認証、認可、およびアカウンティング（AAA）ソリューションを実現します。

RADIUS DTLS のポート

RADIUS のポート（DTLS サーバー）は認証とアカウンティングに使用されます。デフォルトの DTLS サーバー ポートは 2083 です。

RADIUS DTLS ポート番号は **dtls port** *port_number* を使用して変更できます。詳細については、「RADIUS DTLS ポート番号の設定」を参照してください。

共有秘密

すでに特定のサーバーに対して DTLS を有効にしている場合は、共有秘密として **radius/dtls** を使用できます。

CTS 通信のための PAC の処理

CTS 通信のために ISE から PAC をダウンロードできます。PAC をダウンロードしたら、共有秘密の代わりに PAC キーを使用してすべての CTS 属性を暗号化する必要があります。

その後、ISE は PAC を使用してそれらの属性を復号化します。

セッション管理

RADIUS クライアントは、DTLS サーバーからの応答にのみ依存します。セッションが理想的なタイムアウトに最も適している場合は、セッションを閉じる必要があります。

応答が無効の場合は、セッションを削除する必要があります。

DTLS 経由で RADIUS パケットを送信する必要がある場合は、特定のサーバーで DTLS セッションを再確立する必要があります。

ロード バランシング

複数の DTLS サーバーとロード バランシング方式が設定されています。

要求を必要とする送信先の AAA サーバーを選択する必要があります。その後、特定のサーバーの DTLS コンテキストを使用し、RADIUS パケットを暗号化して送り返します。

接続タイムアウト

暗号化された RADIUS パケットを送信した後、再送信タイマーを開始する必要があります。再送信タイマーが期限切れになる前に応答がなかった場合は、パケットが再暗号化され再送信されます。

この試行回数は、**dtls retries** の設定に従って、またはデフォルト値まで継続できます。試行回数が制限を超えると、サーバーは使用不可となり、応答は AAA クライアントに戻されます。

（注） デフォルトの接続タイムアウトは 5 秒です。

接続の再試行回数

RADIUS DTLS は UDP ベースであるため、特定の再試行回数において特定のタイムアウト間隔後に接続を再試行する必要があります。

すべての再試行を終えると、DTLS 接続では次のことが実行されます。

- 失敗としてマークされます。
- RADIUS 要求を処理するために次に使用可能なサーバーを検索します。

（注） デフォルトの接続再試行回数は 5 回です。

アイドル タイムアウト

アイドル タイマーが期限切れになり、最後のアイドル タイムアウト以降にトランザクションが存在しない場合、DTLS セッションは閉じたままになります。

DTLS セッションを確立した後、アイドル タイマーを開始できます。アイドル タイマーを 30 秒間にわたって開始し、RADIUS DTLS パケットの 1 つが送信されると、30 秒後にアイドル タイマーが期限切れになり、RADIUS DTLS トランザクションの数がチェックされます。

アイドル タイマーの値がゼロを超えると、アイドル タイマーはトランザクション カウンタをリセットし、タイマーを再開します。

（注） デフォルトのアイドル タイムアウトは 60 秒です。

サーバーおよびサーバー グループのフェールオーバーの処理

RADIUS サーバーは DTLS ありおよび DTLS なしで設定できます。DTLS 対応サーバーと非 DTLS サーバーを使用して AAA サーバー グループを作成することをお勧めします。ただし、AAA サーバー グループの設定時にはこのような制限は受けません。

DTLS サーバーを選択し、DTLS サーバーが接続を確立し、RADIUS 要求パケットが DTLS サーバーに送信されるとします。すべての RADIUS の再試行後も DTLS サーバーが応答しない場合は、同じサーバー グループ内で次に設定されているサーバーに引き継がれます。次のサーバが DTLS サーバの場合、RADIUS 要求パケットの処理は次のサーバで続行されます。次のサーバーが非 DTLS サーバーの場合、RADIUS 要求パケットの処理はそのサーバー グループでは行われません。その後、サーバー グループのフェールオーバーが発生し、次のサーバー グループが使用可能であれば、同じシーケンスが次のサーバー グループで続行されます。

（注） サーバー グループ内では、DTLS サーバーか非 DTLS サーバーのいずれかのみを使用する必要があります。

前提条件

IOS および BINOS AAA のサポート

AAA サーバーは、IOS および BINOS プラットフォームで動作します。IOS で RADIUS DTLS のサポートを完了したら、同じサポートを BINOS にも移植する必要があります。

RADIUS DTLS サーバーの設定

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 モードを開始します。
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** *server-name* 例： `デバイス(config)# radius server R1`	RADIUS サーバー名を指定します。
ステップ 4	**dtls** 例： `デバイス(config-radius-server)# dtls`	DTLS パラメータを設定します。
ステップ 5	**end** 例： `デバイス(config-radius-server)# end`	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

RADIUS DTLS 接続タイムアウトの設定

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 モードを開始します。
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** *server-name* 例：	RADIUS サーバー名を指定します。

	コマンドまたはアクション	目的
	デバイス(config)# **radius server R1**	
ステップ 4	**dtls connectiontimeout** *timeout* 例： デバイス(config-radius-server)# **dtls connectiontimeout 1**	RADIUS DTLS 接続タイムアウトを設定します。 ここで、各変数は次のように定義されます。 *timeout* は、DTLS 接続タイムアウト値を指します。有効な範囲は 1 ～ 65535 です。
ステップ 5	**end** 例： デバイス(config-radius-server)# **end**	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

RADIUS DTLS アイドル タイムアウトの設定

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 モードを開始します。
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** *server-name* 例： デバイス(config)# **radius server R1**	RADIUS サーバー名を指定します。
ステップ 4	**dtls idletimeout** *idle_timeout* 例： デバイス(config-radius-server)# **dtls idletimeout 2**	RADIUS DTLS アイドル タイムアウトを設定します。 ここで、各変数は次のように定義されます。 *idle_timeout* は、DTLS アイドル タイムアウト値を指します。有効な範囲は 1 ～ 65535 です。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： `デバイス(config-radius-server)# end`	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

RADIUS DTLS サーバー用の送信元インターフェイスの設定

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 モードを開始します。
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** *server-name* 例： `デバイス(config)# radius server R1`	RADIUS サーバー名を指定します。
ステップ 4	**dtls ip** {**radius source-interface Ethernet-Internal** *interface_number* 例： `デバイス(config-radius-server)# dtls ip radius source-interface Ethernet-Internal 0`	RADIUS DTLS サーバーの送信元インターフェイスを設定します。 ここで、各変数は次のように定義されます。 • *interface_number* は、イーサネット内部インターフェイス番号を指します。デフォルト値は 0 です。
ステップ 5	**end** 例： `デバイス(config-radius-server)# end`	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

RADIUS DTLS ポート番号の設定

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 モードを開始します。
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** *server-name* 例： `デバイス(config)# radius server R1`	RADIUS サーバー名を指定します。
ステップ 4	**dtls port** *port_number* 例： `デバイス(config-radius-server)# dtls port 2`	RADIUS DTLS ポート番号を設定します。 ここで、各変数は次のように定義されます。 *port_number* は、DTLS ポート番号を指します。
ステップ 5	**end** 例： `デバイス(config-radius-server)# end`	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

RADIUS DTLS 接続再試行回数の設定

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 モードを開始します。
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。

	コマンドまたはアクション	目的
ステップ 3	**radius server** *server-name* 例： デバイス(config)# **radius server R1**	RADIUS サーバー名を指定します。
ステップ 4	**dtls retries** *retry_number* 例： デバイス(config-radius-server)# **dtls retries 3**	RADIUS 接続の再試行回数を設定します。 ここで、各変数は次のように定義されます。 *retry_number* は、DTLS 接続の再試行回数を指します。有効な範囲は 1 ～ 65535 です。
ステップ 5	**end** 例： デバイス(config-radius-server)# **end**	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

RADIUS DTLS トラストポイントの設定

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 モードを開始します。
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** *server-name* 例： デバイス(config)# **radius server R1**	RADIUS サーバー名を指定します。
ステップ 4	**dtls trustpoint** {**client** *LINE* **dtls** | **server** *LINE* **dtls**} 例： デバイス(config-radius-server)# **dtls trustpoint client client1 dtls** デバイス(config-radius-server)# **dtls trustpoint server server1 dtls**	クライアントとサーバーにトラストポイントを設定します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： `デバイス(config-radius-server)# end`	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

DTLS ダイナミック認証の設定

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 モードを開始します。
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： `デバイス(config)# aaa server radius dynamic-author`	RFC 3576 サポート用のローカル サーバー プロファイルを設定します。
ステップ 4	**dtls** 例： `デバイス(config-locsvr-da-radius)# dtls`	DTLS 送信元パラメータを設定します。
ステップ 5	**end** 例： `デバイス(config-locsvr-da-radius)# end`	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

クライアントの DTLS の有効化

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 モードを開始します。
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： `デバイス(config)# aaa server radius dynamic-author`	RFC 3576 サポート用のローカル サーバー プロファイルを設定します。
ステップ 4	**client** *IP_addr* **dtls** 例： `デバイス(config-locsvr-da-radius)# client 10.104.49.14 dtls`	クライアントの DTLS を有効にします。
ステップ 5	**end** 例： `デバイス(config-locsvr-da-radius)# end`	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

DTLS のクライアント トラストポイントの設定

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 モードを開始します。
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。

	コマンドまたはアクション	目的
ステップ 3	**aaa server radius dynamic-author** 例： `デバイス(config)# aaa server radius dynamic-author`	RFC 3576 サポート用のローカル サーバー プロファイルを設定します。
ステップ 4	**client** *IP_addr* **dtls** {**client-tp** *client-tp-name* | **server-tp** *server-tp-name*} 例： `デバイス(config-locsvr-da-radius)# client 10.104.49.14 dtls client-tp client_tp_name`	DTLS のクライアント トラストポイントを設定します。
ステップ 5	**end** 例： `デバイス(config-locsvr-da-radius)# end`	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

DTLS アイドル タイムアウトの設定

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 モードを開始します。
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： `デバイス(config)# aaa server radius dynamic-author`	RFC 3576 サポート用のローカル サーバー プロファイルを設定します。
ステップ 4	**client** *IP_addr* **dtls idletimeout** *timeout-interval* {**client-tp** *client_tp_name* | **server-tp** *server_tp_name*} 例： `デバイス(config-locsvr-da-radius)# client 10.104.49.14 dtls idletimeout 62 client-tp dtls_ise`	DTLS のアイドル時間を設定します。 ここで、各変数は次のように定義されます。 *timeout-interval* は、アイドル タイムアウト間隔を指します。有効な範囲は 60 ～ 600 です。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： `デバイス(config-locsvr-da-radius)# end`	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

DTLS のサーバー トラストポイントの設定

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 モードを開始します。
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： `デバイス(config)# aaa server radius dynamic-author`	RFC 3576 サポート用のローカル サーバー プロファイルを設定します。
ステップ 4	**client** *IP_addr* **dtls server-tp** *server_tp_name* 例： `デバイス(config-locsvr-da-radius)# client 10.104.49.14 dtls server-tp dtls_client`	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： `デバイス(config-locsvr-da-radius)# end`	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

RADIUS DTLS サーバーの設定の確認

DTLS 対応サーバーに関する情報を表示するには、次のコマンドを使用します。

```
Device# show aaa servers
DTLS: Packet count since last idletimeout 1,
Send handshake count 3,
```

```
Handshake Success 1,
Total Packets Transmitted 1,
Total Packets Received 1,
Total Connection Resets 2,
Connection Reset due to idle timeout 0,
Connection Reset due to No Response 2,
Connection Reset due to Malformed packet 0,
```

RADIUS DTLS 固有の統計情報のクリア

Radius DTLS 固有の統計情報をクリアするには、次のコマンドを使用します。

```
Device# clear aaa counters servers radius {<server-id> | all}
```

（注） *server-id* は、**show aaa servers** によって表示されるサーバー ID を指します。0 ～ 2147483647 の範囲の値を指定できます。

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。