Configuring RADIUS over DTLS

Prerequisites for RADIUS over DTLS

Following are the prerequisites for RADIUS over DTLS:

- Ensure that the device is running Cisco IOS crypto K9 image.
- Ensure that crypto PKI is configured on the device.
- Support for RADIUS over DTLS is available on Cisco ISE 2.2 and above.

Information about RADIUS over DTLS

DTLS provides encryption services over RADIUS, which is transported over a secure tunnel. RADIUS over DTLS is implemented in both client and server. Client side controls radius Authentication, Authorization, and Accounting (AAA) and server side controls Change of Authorization (CoA).

You can configure the following parameters:

- Per client specific idle_timeout, client trustpoint and server trustpoint.
- Global CoA specific DTLS listening port and list of source interfaces.

You can disable DTLS for a specific server by using the command **no dtls** in the radius server configuration mode.

How to Configure RADIUS over DTLS

How to Configure DTLS Server

Although there is no configuration restriction, it is recommended to use the same type, either only DTLS or only non-DTLS, for server under a AAA server group.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**radius server** *radius-server-name* **Example:** `Device(config)# radius server R1`	Enters radius server configuration mode.

Step 4	**dtls** [**connectiontimeout** *connection-timeout-value*] [**idletimeout** *idle-timeout-value*] [**ip** {**radius source-interface** *interface-name* |**vrf forwarding** *forwarding-table-name*}] [**port** *port-number*] [**retries** *number-of-connection-retries*] [**trustpoint** {**client** *trustpoint name*|**server***trustpoint name*}]
	Example:
	`Device(config-radius-server)# dtls connectiontimeout 10`
	`Device(config-radius-server)# dtls idletimeout 5`
	`Device(config-radius-server)# dtls retries 15`
	`Device(config-radius-server)# dtls ip radius source-interface Ethernet 0/0`
	`Device(config-radius-server)# dtls ip vrf forwarding table-1`
		Configures DTLS parameters. You can configure the following parameters:
		• **connectiontimeout** —
		Configures DTLS connection timeout value.
		• **idletimeout** —
		Configures DTLS idle timeout value.
		• **ip** —
		Configures IP source parameters.
		• **port** —
		Configures DTLS port number.
		• **retries** —
		Configures number of DTLS connection retries.
		• **trustpoint** —
		Configures DTLS trustpoint for client and server.

	Command or Action	Purpose

	`Device(config-radius-server)# dtls port 10`
	`Device(config-radius-server)# dtls trustpoint`
	`Device(config-radius-server)# dtls trustpoint client TP-self-signed-721943660`
	`Device(config-radius-server)# dtls trustpoint server isetp`	

Step 5	**end** **Example:** `Device(config-radius-server)# end`	Returns to privileged EXEC mode.

How to Configure Dynamic Authorization for DTLS CoA

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**aaa server radius dynamic-author** **Example:** `Device(config)# aaa server radius dynamic-author`	Enters dynamic authorization local server configuration mode and specifies a RADIUS client from which a device accepts Change of Authorization (CoA) and disconnect requests. Configures the device as a AAA server to facilitate interaction with an external policy server.
Step 4	**client {ip-addr | hostname}** [**dtls** [**client-tp** *client-tp-name*] [**idletimeout** *idletimeout-interval*] [**server-tp** *server-tp-name*] | **vrf** *vrf-id*] **Example:** `Device(config-locsvr-da-radius)# client 10.104.49.14 dtls idletimeout 100 client-tp dtls_ise server-tp dtls_client`	Configures the IP address or hostname of the AAA server client. You can configure the following optional parameters: • **dtls** — Enables DTLS for the client. • **client-tp** —

	Command or Action	Purpose
		Configures client trustpoint. • **idletimeout** — Configures DTLS idle timeout value. • **server-tp** — Configures server trustpoint. • **vrf** — Virtual routing and forwarding (VRF) ID of the client.

Step 5	**dtls** {**ip radius source-interface** *interface-name* | **port** *radius-dtls-server-port-number*}
	Example:
	`Device(config-locsvr-da-radius)# dtls ip radius source-interface GigabitEthernet 1/0/24`
	`Device(config-locsvr-da-radius)# dtls port 100`
		Configures RADIUS CoA server. You can configure the following parameters:
		• **ip radius source-interface***interface-name* —
		Specifies the interface for source address in RADIUS CoA Server.
		• **port** *radius-dtls-server-port-number* —
		Specifies port on which local DTLS RADIUS server listens.

Step 6	**end** **Example:** `Device(config-radius-server)# end`	Returns to privileged EXEC mode.

Monitoring RADIUS over DTLS

The following commands can be used to monitor DTLS server statistics:

Table 1: Monitoring DTLS Server Statistics Command

Command	Purpose
show aaa servers	Displays information related to DTLS server. Following statistics information is displayed using **show aaa servers** command: • pkt_cnt_since_idle_tiemout • send_hs_start_cnt • hs_success_cnt • total_tx_pkt_cnt • total_rx_pkt_cnt • total_conn_reset_cnt • conn_reset_cnt_idle_timeout • conn_reset_cnt_no_resp • conn_reset_cnt_malformed_pkt • conn_reset_cnt_error_case
clear aaa counters servers radius{ *server id*|**all**}	Clears the RADIUS DTLS specific statistics.
debug radius dtls	Enables RADIUS DTLS specific debugs.

Examples of RADIUS over DTLS

The following is a sample output for the statistics per DTLS connection:

```
Device# show aaa servers
RADIUS: id 53, priority 1, host 11.22.45.45, DTLS port 2083
State: current UP, duration 1596921s, previous duration 0s
Dead: total time 0s, count 0
Platform State from SMD: current DEAD, duration 53s, previous duration
0s
SMD Platform Dead: total time 1197328s, count 19957
Platform State from WNCD: current UP, duration 0s, previous duration 0s
Platform Dead: total time 0s, count 0
Quarantined: No
Authen: request 0, timeouts 0, failover 0, retransmission 0
Response: accept 0, reject 0, challenge 0
Response: unexpected 0, server error 0, incorrect 0, time 0ms
Transaction: success 0, failure 0
Throttled: transaction 0, timeout 0, failure 0
Author: request 0, timeouts 0, failover 0, retransmission 0
Response: accept 0, reject 0, challenge 0
Response: unexpected 0, server error 0, incorrect 0, time 0ms
Transaction: success 0, failure 0
Throttled: transaction 0, timeout 0, failure 0
Account: request 0, timeouts 0, failover 0, retransmission 0
Request: start 0, interim 0, stop 0
Response: start 0, interim 0, stop 0
```

```
Response: unexpected 0, server error 0, incorrect 0, time 0ms
Transaction: success 0, failure 0
Throttled: transaction 0, timeout 0, failure 0
DTLS: Packet count since last idletimeout 0,
Send handshake count 0,
Handshake Success 0,
Total Packets Transmitted 0,
Total Packets Received 0,
Total Connection Resets 0,
Connection Reset due to idle timeout 0,
Connection Reset due to No Response 0,
Connection Reset due to Malformed packet 0,
Connection Reset by Peer 0,
Connection Reset due to other Errors 0,
Elapsed time since counters last cleared: 0m
Estimated Outstanding Access Transactions: 0
Estimated Outstanding Accounting Transactions: 468
Estimated Throttled Access Transactions: 0
Estimated Throttled Accounting Transactions: 0
Maximum Throttled Transactions: access 0, accounting 0
```