IP SLAs TWAMP Responder

The Two-Way Active Measurement Protocol (TWAMP) defines a flexible method for measuring round-trip IP performance between any two devices.

TWAMP enables complete IP performance measurement. TWAMP also provides a flexible choice of solutions because it supports all devices deployed in the network.

This chapter describes how to configure the Two-Way Active Measurement Protocol (TWAMP) responder on a Cisco device to measure IP performance between the Cisco device and a non-Cisco TWAMP control device on your network.

Note IPv6 is supported for IP SLA TWAMP Responder on the RSP3 module.

Prerequisites for IP SLAs TWAMP Responder

- A TWAMP control client and a session sender must be configured in your network.
- IP SLA responder must be configured on the device. Use the command **ip sla responder twamp** to configure IP SLA responder.

Restrictions for IP SLAs TWAMP Responder

- Time stamping is not supported for TWAMP test packets that ingress or egress through management interfaces. Time stamping is supported only on routed interfaces and BDI interfaces.
- TWAMP client and session sender are not supported.
- TWAMP Light mode is not supported until the Cisco IOS XE Bengaluru 17.4.1 release.

IP SLAs TWAMP Architecture

Two-Way Active Measurement Protocol (TWAMP)

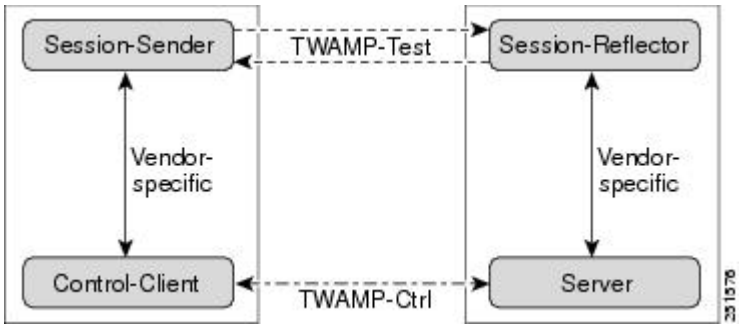
The IETF Two-Way Active Measurement Protocol (TWAMP) defines a standard for measuring round-trip network performance between any two devices that support the TWAMP protocols. The TWAMP-Control protocol is used to set up performance measurement sessions. The TWAMP-Test protocol is used to send and receive performance measurement probes.

The TWAMP architecture is composed of the following four logical entities that are responsible for starting a monitoring session and exchanging packets:

- The control client: It sets up, starts, and stops TWAMP test sessions.
- The session sender: It instantiates TWAMP test packets that are sent to the session reflector.
- The session reflector: It reflects a measurement packet upon receiving a TWAMP test packet. The session reflector does not collect packet statistics in TWAMP.
- The TWAMP server: It is an end system that manages one or more TWAMP sessions and is also capable of configuring each session ports in the end points. The server listens on the TCP port. The session-reflector and server make up the TWAMP responder in an IP SLAs operation.

Although TWAMP defines the different entities for flexibility, it also allows for logical merging of the roles on a single device for ease of implementation. The figure below shows the interactions of four entities of the TWAMP architecture.

Figure 1: TWAMP Architecture



TWAMP Protocols

The TWAMP protocol includes three distinct message exchange categories, they are:

- Connection setup exchange—Messages establish a session connection between the control client and the server. First, the identities of the communicating peers are established via a challenge response mechanism. The server sends a randomly generated challenge, to which the control client then sends a response by encrypting the challenge using a key derived from the shared secret. Once the identities are established, the next step negotiates a security mode that is binding for the subsequent TWAMP-Control commands as well as the TWAMP-Test stream packets.

Note A server can accept connection requests from multiple control clients.

- TWAMP control exchange—The TWAMP control protocol runs over TCP and is used to instantiate and control measurement sessions. The sequence of commands is as follows:
  - request session
  - start session
  - stop session

  However, unlike the connection setup exchanges, the TWAMP control commands can be sent multiple times. However, the messages cannot occur out of sequence although multiple request session commands can be sent before a session start command.
- TWAMP test stream exchange—The TWAMP test runs over UDP and exchanges TWAMP test packets between session sender and session reflector. These packets include timestamp fields that contain the instant of packet egress and ingress. The packet also includes a sequence number.

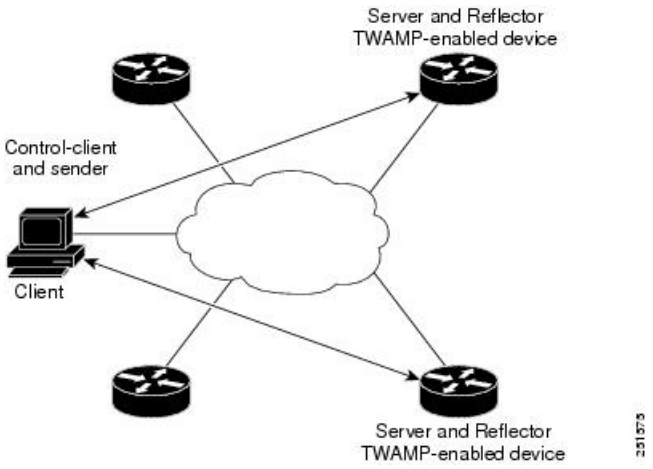
Note TWAMP control and TWAMP test stream support only unauthenticated security mode.

IP SLAs TWAMP Responder

A TWAMP responder interoperates with the control client and session sender on another device that supports TWAMP. In the current implementation, the session reflector and TWAMP server that make up the responder must be co-located on the same device.

In the figure below, one device is the control client and session-sender (TWAMP control device), and the other two devices are Cisco devices that are configured as IP SLAs TWAMP responders. Each IP SLAs TWAMP responder is both a TWAMP server and a session-reflector.

Figure 2: IP SLAs TWAMP Responders in a Basic TWAMP Deployment



Note ASR 920 supports only hardware time stamping.

Configure an IP SLAs TWAMP Responder

Note Effective Cisco IOS-XE Everest 16.6.1, time stamping for sender (T1, T4) and receiver (T3, T2) is performed by the hardware, instead of the software. This time stamping is done bythe hardware to improve the accuracy of jitter and latency measurements.

Note Software time stamping is implemented for TWAMP IP SLA packets on the RSP3 module.

Configuring the TWAMP Server

Note In the current implementation of IP SLAs TWAMP Responder, the TWAMP server and the session reflector must be configured on the same device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip sla server twamp**
4. **port** *port-number*
5. **timer inactivity** *seconds*
6. **end**

DETAILED STEPS

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 **ip sla server twamp**

Example:

```
Device(config)#  ip sla server twamp
```

Configures the device as a TWAMP server and enters TWAMP server configuration mode.

Step 4 **port** *port-number*

Example:

```
Device(config-twamp-srvr)# port 9000
```

(Optional) Configures the port to be used by the TWAMP server to listen for connection and control requests.

Step 5 **timer inactivity** *seconds*

Example:

```
Device(config-twamp-srvr)# timer inactivity 300
```

(Optional) Configures the inactivity timer for a TWAMP control session.

Step 6 **end**

Example:

```
Device(config-twamp-srvr)# end
```

Returns to privileged EXEC mode.

Configuring the Session Reflector

Note In the current implementation of IP SLAs TWAMP Responder, the TWAMP server and the session reflector must be configured on the same device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip sla responder twamp**
4. **timeout** *seconds*
5. **end**

DETAILED STEPS

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 **ip sla responder twamp**

Example:

```
Device(config)# ip sla responder twamp
```

Configures the device as a TWAMP responder and enters TWAMP reflector configuration mode.

Step 4 **timeout** *seconds*

Example:

```
Device(config-twamp-ref)#  timeout 300
```

(Optional) Configures an inactivity timer for a TWAMP test session.

Step 5 **end**

Example:

```
Device(config-twamp-ref)# end
```

Exits to privileged EXEC mode.

Configuration Example for IP SLAs TWAMP Responder

The following example and partial output shows how to configure the TWAMP server and the session reflector on the same Cisco device. In this configuration, port 862 is the (default) port to be used by the TWAMP server to listen for connection and control requests. The port for the server listener is the RFC-specified port and if required, can be reconfigured.

Note For the IP SLAs TWAMP responder to function, a control client and the session sender must be configured in your network.

The following examples are for non-VRF scenarios (default):

```
Device> enable
Device# configure terminal
Router(config)# ip sla serv twamp
Router(config-twamp-srvr)# port 12000
Router(config-twamp-srvr)# timer inactivity 1200
Router(config-twamp-srvr)# exit
Router(config)# ip sla responder tw
Router(config)# ip sla responder twamp
Router(config-twamp-ref)# resp
Router(config-twamp-ref)# time
Router(config-twamp-ref)# timeout 2000
Router(config-twamp-ref)# exit

Router# show ip sla twamp connection requests
    Connection-Id     Client Address   Client Port      Client VRF
          A3              100.1.0.1         59807         default

Router# show ip sla twamp connection detail
Connection Id:          A3
  Client IP Address:    100.1.0.1
  Client Port:          59807
  Client VRF:           intf2
  Mode:                 Unauthenticated
  Connection State:     Connected
  Control State:        Active
  Number of Test Requests - 0:1

Router# show ip sla twamp session
IP SLAs Responder TWAMP is: Enabled
Recvr Addr: 100.1.0.2
Recvr Port: 7
Sender Addr: 100.1.0.1
Sender Port: 34608
Sender VRF: default
Session Id: 100.1.0.2:15833604877498391199:6D496912
Connection Id: 101

Router# sh running-config | b twamp
ip sla responder twamp
 timeout 2000
ip sla responder
ip sla enable reaction-alerts
ip sla server twamp
 port 12000
 timer inactivity 1200
!
!
```

The following examples are for VRF scenarios:

```
Router# show ip sla twamp session
IP SLAs Responder TWAMP is: Enabled
Recvr Addr: 100.1.0.2
Recvr Port: 7
Sender Addr: 100.1.0.1
Sender Port: 51486
```

```
Sender VRF: intf1
Session Id: 100.1.0.2:9487538053959619969:73D5EDEA
Connection Id: D0

Router# show ip sla twamp connection detail
Connection Id:          A3
  Client IP Address:    100.1.0.1
  Client Port:          52249
  Client VRF:           intf2
  Mode:                 Unauthenticated
  Connection State:     Connected
  Control State:        Active
  Number of Test Requests - 0:1

Router# show ip sla twamp connection requests
  Connection-Id    Client Address   Client Port    Client VRF
             A3       100.1.0.1       52249          intf2
 Total number of current connections: 1
```

Note The default port for IP SLA server is 862.