

WAN Monitoring

This chapter contains the following sections:

- Information About WANMon, on page 1
- Prerequisites , on page 2
- Guidelines and Limitations, on page 2
- Configuring WANMon, on page 3
- Verifying WANMon Configuration, on page 4
- Configuration Examples, on page 5

Information About WANMon

WANMon is a flexible solution to address the WAN link recovery requirements for the following products and interfaces:

- Physical networks: 4G LTE and Ethernet (WAN port)
- Virtual links: Non-crypto map based IPSec tunnels (either legacy or FlexVPN); that is, any IPSec tunnel you configure as an interface.

You enable WANMon to monitor your WAN links and initiate link recovery actions on receipt of link failure triggers.

Built-in Recovery Actions

The following are the three levels of built-in recovery processes specific to the link type:

Link Type	Recovery Actions		
	Level 0 (Immediate)	Level 1 (Active)	Level 2 (Last-Resort)
4G LTE	Clear interface, and then shut/no-shut	Module reload	System reload
Ethernet	Clear interface, and then shut/no-shut	No action taken	System reload
Tunnel	Shut/no-shut	No action taken	System reload

Each level has two time-based thresholds based on which built-in recovery actions are taken. The following are the default settings for each level:

- *threshold* is the wait time in minutes after receipt of a link failure trigger to initiate the recovery action as set in the specified level.
- *mintime* is the frequency to perform the recovery action if the link remains down.

The built-in values are:

Level	Threshold	Mintime	Description
Level 0	10 min	10 min	Triggers Level 0 actions 10 minutes after the link went down. Repeat no more than every 10 minutes.
Level 1	60 min	60 min	Triggers Level 1 actions 10 minutes after the link went down. Repeat no more than every 60 minutes.
Level 2	480 min	60 min	Triggers Level 2 actions 480 minutes after the link went down. Repeat no more than every 60 minutes.

Note If threshold values are specified as 0, no recovery actions are taken for that level. You can use this to avoid system reload (the built-in Level 2 recovery action) on receipt of a link failure trigger where other WAN links may be operational.

Prerequisites

Ensure that the WANMon module is available. The WANMon module is included in the IOS-XE image as the *tm_wanmon.tcl* policy file.

Guidelines and Limitations

- WANMon automatically performs IP address checking (no user configuration) as required for cellular interfaces.
- For all other interfaces, WANMon never performs IP address checking.
- WANMon indirectly triggers user-specified actions by generating an application event that link resetter applets monitor.

- If your network is live, ensure that you understand the potential impact of any command.

Configuring WANMon

You can enable WANMon on the router and assign WAMMon support to specific interfaces. Optionally, you can override the built-in recovery actions, define custom recovery links, and define an event manager environment policy to set the track object value and disable IP address checking. WANMon is disabled by default.

Procedure

Step 1 **event manager policy** *tm_wanmon.tcl* **authorization bypass**

Enables the WANMon link recovery module.

Use **authorization bypass** to avoid authorization for CLIs invoked by this policy.

Step 2 **event manager environment wanmon_if_list** <*instance*> {*interface name* {**ipsla** <*instance*>}}

Configures WANMon for the interfaces in your WAN, and indicates that this is an interface configuration command.

Note Any environment variable with the prefix wanmon_if_list constitutes an interface configuration.

Multiple interfaces are allowed by specifying an instance.

Be sure to specify the full interface name (for example, cellular0/1/0 or cellular0/3/0).

You can set the IP SLA icmp-echo trigger, if desired. Multiple IP SLA triggers are allowed by specifing an instance.

Note WANMon only looks at the status of the SLA ID. Even though *icmp-echo* is most common, if needed any other type of SLA probe (for example, *udp-echo*) can be used instead.

Step 3 **event manager environment wanmon_if_listx** {*interface name* {**recovery** *Level0* {*Level1* } *Level2*}}

(Optional) Overrides the built-in thresholds.

Step 4 **publish-event sub-system 798 type 2000 arg1** <*interface name*> **arg2** <*level* >

(Optional) Configures custom recovery actions using link resetter applets.

<*interface* > is the full interface name (for example, cellular0/1/0 or cellular0/3/0).

<*level* > is 0, 1, or 2 to match the desired link recovery action.

Step 5 **{stub** <*track-stub-id* > **}**

(Optional) Allows an event manager environment policy to set the track object value. WANMon can set a track-stub-object value to reflect the link state so that an external applet can track the stub object.

Step 6 **event manager environment wanmon_if_listx** {<*interface name* > {**checkip** <*instance* >}}

(Optional) Disables IP address checking.

What to do next

EXAMPLES

```
event manager policy tm_wanmon.tcl authorization bypass
```

The following examples are Event Manager commands to configure cellular and Ethernet interfaces:

```
event manager environment wanmon_if_list1 {cellular0/1/0 {ipsla 1}}
event manager environment wanmon_if_list2 {GigabitEthernet0/0/0 {ipsla 2}}
```

This example sets custom recovery thresholds:

```
event manager environment wanmon_if_list {cellular0/1/0 {recovery 20 {90 75} 600}
```

where:

- The Level 0 threshold is set to 20 minutes after the link failure trigger. Level 0 recovery actions are performed for the cellular interface. Repeats indefinitely, no more than every 10 minutes (default).
- Level 1 threshold is set to 90 minutes. Level 1 recovery actions are performed for the cellular interface. Repeats no more frequently than every 75 minutes.
- The Level 2 threshold is set to 600 minutes (10 hours).

The following sets the track-stub-object value to 21:

```
conf t
track 21 stub-object
event manager environment wanmon_if_list {cellular0/1/0 {ipsla 1} {stub 21}
```

Verifying WANMon Configuration

Use the following steps to verify your WANMon configuraion.

Procedure

Step 1 **show event manager policy registered**

Displays the WAN monitoring policy.

Step 2 **show event manager environment**

Displays the interface environment variables set during interface configuration.

What to do next

EXAMPLE

```
show event manager policy registered
1    script    system  multiple   Off   Thu Jan 16 18:44:29 2014  tm_wanmon.tcl

show event manager environment
1 wanmon_if_list   {cell0/1/0 {ipsla 1}}
```

Configuration Examples

The following examples are provided:

WANMon Cellular Interface Configuration Example

```
track 1 ip sla 1
ip sla 1
 icmp-echo 172.27.166.250
 timeout 6000
 frequency 300
ip sla schedule 1 life forever start-time now
event manager environment wanmon_if_list {cellular0/1/0 {ipsla 1}}
event manager policy tm_wanmon.tcl authorization bypass
```

Multiple WAN Link Monitoring Example

```
track 1 ip sla 1
track 21 stub-object
ip sla 1
 icmp-echo 172.27.166.250
 timeout 6000
 frequency 300
ip sla schedule 1 life forever start-time now
track 2 ip sla 2
track 22 stub-object
ip sla 2
 icmp-echo 10.27.16.25
 timeout 6000
 frequency 300
ip sla schedule 2 life forever start-time now

event manager environment wanmon_if_list1 {cellular0/1/0 {ipsla 1} {stub 21}}

event manager policy tm_wanmon.tcl authorization bypass
```