Configuring TCP MSS Adjustment

- Restrictions for TCP MSS Adjustment, on page 1
- Information about TCP MSS Adjustment, on page 1
- How to Configure TCP MSS Adjustment, on page 2
- Configuration Examples for TCP MSS Adjustment, on page 3
- Feature History for TCP MSS Adjustment, on page 4

Restrictions for TCP MSS Adjustment

- Subinterfaces do not support TCP MSS Adjust.
- TCP MSS adjustment configuration works only if applied on an ingress interface. This configuration does not work if applied on an egress interface.

Information about TCP MSS Adjustment

The Transmission Control Protocol (TCP) Maximum Segment Size (MSS) Adjustment feature enables the configuration of the maximum segment size for transient packets that traverse a router, specifically TCP segments with the SYN bit set. Use the **ip tcp adjust-mss** command in interface configuration mode to specify the MSS value on the intermediate router of the SYN packets to avoid truncation.

When a host (usually a PC) initiates a TCP session with a server, it negotiates the IP segment size by using the MSS option field in the TCP SYN packet. The value of the MSS field is determined by the MTU configuration on the host. The default MSS value for a PC is 1500 bytes.

The PPP over Ethernet (PPPoE) standard supports an MTU of only 1492 bytes. The disparity between the host and PPPoE MTU size can cause the router in between the host and the server to drop 1500-byte packets and terminate TCP sessions over the PPPoE network. Even if the path MTU (which detects the correct MTU across the path) is enabled on the host, sessions may be dropped because system administrators sometimes disable the ICMP error messages that must be relayed from the host in order for path MTU to work.

The **ip tcp adjust-mss** command helps prevent TCP sessions from being dropped by adjusting the MSS value of the TCP SYN packets.

The **ip tcp adjust-mss** command is effective only for TCP connections passing through the router.

In most cases, the optimum value for the *max-segment-size* argument of the **ip tcp adjust-mss** command is 1452 bytes. This value plus the 20-byte IP header, the 20-byte TCP header, and the 8-byte PPPoE header add up to a 1500-byte packet that matches the MTU size for the Ethernet link.

Note TCP MSS adjustment-based traffic is always software switched.

Supported Interfaces

TCP MSS Adjust is supported only on the following interfaces:

- Physical Layer 3 interface
- SVI
- Layer 3 port channel
- Layer 3 GRE tunnel

How to Configure TCP MSS Adjustment

The following sections provide configuration information for TCP MSS adjustment.

Configuring the MSS Value for Transient TCP SYN Packets

Before you begin

Perform this task to configure the MSS for transient packets that traverse a router, specifically TCP segments with the SYN bit set.

We recommend that you use **ip tcp adjust-mss 1452** command.

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#config terminal`	Enters the global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)#interface GigabitEthernet 1/0/0`	Configures an interface type and enters interface configuration mode.

	Command or Action	Purpose
Step 4	**ip tcp adjust-mss** *max-segment-size* **Example:** `Device(config-if)#ip tcp adjust-mss 1452`	Adjusts the MSS value of TCP SYN packets going through a router. The max-segment-size argument is the maximum segment size, in bytes. The range is from 500 to 1460.
Step 5	**end** **Example:** `Device(config-if)#end`	Exits to global configuration mode.

Configuring the MSS Value for IPv6 Traffic

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#config terminal`	Enters the global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)#interface GigabitEthernet 1/0/0`	Configures an interface type and enters interface configuration mode.
Step 4	**ipv6 tcp adjust-mss** *max-segment-size* **Example:** `Device(config-if)#ipv6 tcp adjust-mss 1440`	Adjusts the MSS value of TCP DF packets going through a device. The max-segment-size argument is the maximum segment size, in bytes. The range is from 40 to 1440.
Step 5	**end** **Example:** `Device(config-if)#end`	Exits interface configuration mode and returns to privileged EXEC mode.

Configuration Examples for TCP MSS Adjustment

The following sections provide configuration examples for TCP MSS adjustment.

Example: Configuring the TCP MSS Adjustment

```
Device(config)#vpdn enable
Device(config)#no vpdn logging
Device(config)#vpdn-group 1
Device(config-vpdn)#request-dialin
Device(config-vpdn-req-in)#protocol pppoe
Device(config-vpdn-req-in)#exit
Device(config-vpdn)#exit
Device(config)#interface GigabitEthernet 0/0/0
Device(config-if)#ip address 192.168.100.1.255.255.255.0
Device(config-if)#ip tcp adjust-mss 1452
Device(config-if)#ip nat inside
Device(config-if)#exit
```

Example: Configuring the TCP MSS Adjustment for IPv6 traffic

```
Device>enable
Device#configure terminal
Device(config)#interface GigabitEthernet 0/0/0
Device(config)#ipv6 tcp adjust-mss 1440
Device(config)#end
```

Feature History for TCP MSS Adjustment

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Gibraltar 16.11.1	Transmission Control Protocol (TCP) Maximum Segment Size (MSS) Adjustment	The TCP MSS Adjustment feature enables the configuration of the maximum segment size for transient packets that traverse a router, specifically TCP segments with the SYN bitset. This feature helps prevent TCP sessions from being dropped by adjusting the MSS value of the TCP SYN packets.

Use the Cisco Feature Navigator to find information about platform and software image support.