Solution ID: **sk61221** Technical Level: **Advanced**

Issues requiring adjustment of the Maximum Segment Size (MSS) of TCP SYN and TCP SYN-ACK packets on Security Gateway

Product

IPSec VPN, SecureXL

Version

R77.20 (EOS), R77.30 (EOS), R80.10 (EOS), R80.20 (EOS), R80.20SP (EOS), R80.30 (EOS), R80.30SP (EOS), R80.40 (EOS), R81 (EOS), R81.10, R81.20

OS

Gaia

Platform

All

Last Modified

2022-12-21

Symptoms

- Traffic is not passing through Gateway, as expected, due to MTU and/or TCP MSS issues.
- Latency in traffic running via Site-to-Site VPN.
- "**IPSEC_mtu_icmp**" kernel table is getting full by the relevant connection.
- Web traffic is dropped when using a PPPoE link, cannot go to any website in a Web browser.
- Kernel debug ('**fw ctl debug -m fw + drop**') shows:
  fw_log_drop: Packet proto= ... dropped by fwchain_frag Reason: wait for more fragments;
  ..dropped by fwlinux_nfipout Reason: packet with IP_DF larger than MTU;

Solution

Table of Contents:

Notes:

- This article does *not* apply to Security Gateway in VSX Mode.
- MSS adjustments instructions for IPsec VPN traffic (including VSX environments) are listed in sk101219 - New VPN features in R77.20.

(1) Introduction

"Clamping" the negotiated TCP MSS between hosts may be desirable in some scenarios, due to network needs and inability to control MTU/MSS settings on end-hosts. The following procedure can be used on Security Gateway to "clamp" TCP MSS value on an intermediary host, affecting the negotiated MSS value.

Formula:

TCP MSS value = [MTU value on interface - TCP Header Length]

Note: The minimum TCP header size is 20 bytes and maximum is 60 bytes (allowing for up to 40 bytes of options in the header). Generally, TCP Header Length is 40 bytes.

Example for TCP Header Length of 40 bytes:

- Connection is initiated from 10.10.1.0/24 network to 20.20.1.0/24 network:

  [Host 1] (10.10.1.0/24) --- (eth0)[Security Gateway](eth1) --- (20.20.1.0/24) [Host 2]

- Suppose, we need to "clamp" the TCP MSS at 1280 bytes on the outgoing interface *eth1*.
- Therefore, the MTU on interface *eth1* has to be set to:

  [TCP MSS + TCP Header Length] = [1280 + 40] = 1320 bytes.

After enforcing the "clamping" per the above procedure:

- TCPdump on incoming interface *eth0* will show:

*13:45:33.002185 10.10.1.3.62912 > 20.20.1.3.23: S 4273442499:4273442499(0) win 65535 <**mss 1460**,nop,wscale 1,nop,nop,timestamp[|tcp]> (DF) [tos 0x10]*

- TCPdump on outgoing interface *eth1* will show:

  *13:45:33.002511 10.10.1.3.62912 > 20.20.1.3.23: S 4273442499:4273442499(0) win 65535 <**mss 1280**,nop,wscale 1,nop,nop,timestamp[|tcp]> (DF) [tos 0x10]*

As you can see, TCP SYN comes in through interface *eth0* with an advertised MSS of 1460 (default 1500 bytes - 40 bytes), which is clamped on the interface *eth1* to 1280 (1320 bytes - 40 bytes).

(2) Procedure

This problem was fixed. The fix is included in:

- **Check Point R80.20 (PMTR-13421)**
- **Check Point R77.30 integrated with Jumbo take 345**

Check Point recommends to always upgrade to the most recent version (Security Gateway).

For **other supported versions**, Check Point Support can supply a **Hotfix**.
A Support Engineer will make sure the Hotfix is compatible with your environment before providing the Hotfix.
For faster resolution and verification, please collect CPInfo files from the Security Management Server and Security Gateways involved in the case.

The enforcement of "clamping" of the TCP Maximum Segment Size (MSS) on Security Gateway is controlled by the parameter "***fw_clamp_tcp_mss***" (hotfix for Issue ID 02489940 is required).

Important Notes:

1. To enforce the "clamping" of TCP MSS, the value of "*fw_clamp_tcp_mss*" parameter has to set on ***both*** sides - on Security Gateway (its value set to "*1*") and on Security Management Server (its value set to "*true*").

2. Since R80.30, the change will be done only using the GuiDbEdit Tool, and the relevant parameter needs to be modified is: fw_clamp_tcp_mss_control.

Value of "*fw_clamp_tcp_mss*" on Security Gateway	**Value of "*fw_clamp_tcp_mss*"**	**Final result**

	on Security Management Server	
0	false	Disables the "clamping" of TCP MSS (default values).
1	***true***	*Enables the "clamping" of TCP MSS*.
0	true	The "clamping" of TCP MSS is disabled.
1	false	The "clamping" of TCP MSS will be enforced *only until the policy is loaded* by the Security Gateway (either during policy installation from Security Management Server, or manually using the "*fw fetch*" command).

Note: Cisco does this "clamping" by ***ip tcp adjust-mss [mss_size]*** command.

Instructions

To enforce "clamping" of the TCP Maximum Segment Size (MSS) *permanently*, follow these steps on Check Point machines:

1. **On Security Gateway / *each* cluster member**

   Note: In cluster, these changes must be made on *all* members of the cluster.

   A. Install the required hotfix (Issue ID 02489940).

   This hotfix will allow to enforce the "clamping" of TCP MSS using the "*fw_clamp_tcp_mss*" parameter.

   - Using CPUSE - On Security Gateway running Gaia OS R75.40 and higher:

     Make sure to install the latest build of the CPUSE Agent.

     Refer to sk92449: CPUSE - Gaia Software Updates (including Gaia Software Updates Agent):

     - Section "(4-A-c)" / "(4-A-d)" - refer to import instructions for *Offline procedure*
     - Section "(4-B-a)" - refer to installation instructions for *Hotfixes*

You can also use the sk111158 - Central Deployment Tool (CDT) to install this hotfix on Security Gateways.

Note: Reboot is required.

- Using Legacy CLI - On VSX Gateway running Gaia OS R75.40VS and higher; On Security Gateway running SecurePlatform OS:

  Note: On these versions of VSX, the Gaia CPUSE does not support installation of hotfixes (refer to sk92449 - section "(2)" - "VSX Gateways").

  A. Transfer the hotfix package to the machine (into some directory, e.g., */some_path_to_fix/*).

  B. Unpack and install the hotfix package:

  [Expert@HostName]# cd /some_path_to_fix/
  [Expert@HostName]# tar -zxvf fw1_wrapper_<HOTFIX_NAME>.tgz
  [Expert@HostName]# ./fw1_wrapper_<HOTFIX_NAME>

  Note: The script will stop all of Check Point services (*cpstop*) - read the output on the screen.

  C. Reboot the machine.

B. Add the following line to *$FWDIR/boot/modules/fwkern.conf* file (spaces and comments are *not* allowed):

fw_clamp_tcp_mss=1

Note: Refer to sk26202 - Changing the kernel global parameters on all platforms.

C. Set the desired MTU on relevant interfaces:

Note: Make sure that the configured MTU matches the MTU on the next hop devices.

Formula: *TCP MSS value = [MTU value on interface - TCP Header Length]*

- On Gaia OS:

  Run these commands in Clish:

  i. Set the desired MTU on the relevant interface:
  HostName> ***set interface <INTERFACE_NAME> mtu <VALUE>***

  ii. Save the changes in Gaia Database:
  HostName> ***save config***

- On SecurePlatform OS:

  i. Go to ***sysconfig*** menu:
  [Expert@HostName]# sysconfig
  ii. Go to ***Network Connections***.
  iii. Choose ***Configure Connection***.
  iv. Choose the relevant interface.
  v. Choose ***Change MTU Settings***.
  vi. Set the desired MTU value.
  vii. Press ***e*** to exit from the ***sysconfig*** menu to usual prompt.

D. Reboot the Security Gateway / each cluster member.

2. **On Security Management Server / Domain Management Server (Pre-R80.30)**

Note: This is a global parameter, and will be applied for ***all*** Security Gateways / Clusters that are managed by this Management Server - depending on the value of the kernel parameter "**`fw_clamp_tcp_mss`**" on the Security Gateways / Cluster Members.

A. Close all SmartConsole windows (SmartDashboard, SmartView Tracker, etc.).

B. Connect with GuiDbEdit Tool to Security Management Server / Domain Management Server.

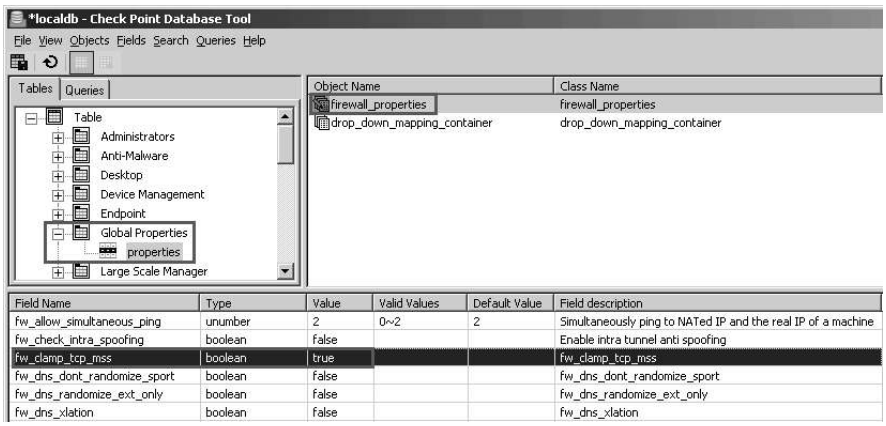
C. In the left upper pane go to ***Table - Global Properties - properties***.

D. In the right upper pane, click on the ***firewall_properties***.

E. Press CTRL+F (or go to ***Search*** menu - ***Find***) - paste ***fw_clamp_tcp_mss*** - click on ***Find Next***.

F. In the lower pane, double-click on ***fw_clamp_tcp_mss*** field.

G. Change the value from ***false*** to ***true*** - click ***OK***.



H. Save the changes: go to ***File*** menu - click on ***Save All***.

I. Close the GuiDbEdit Tool.

J. Connect with SmartDashboard to Security Management Server / Domain Management Server.

K. Install the policy onto Security Gateway / Cluster object.

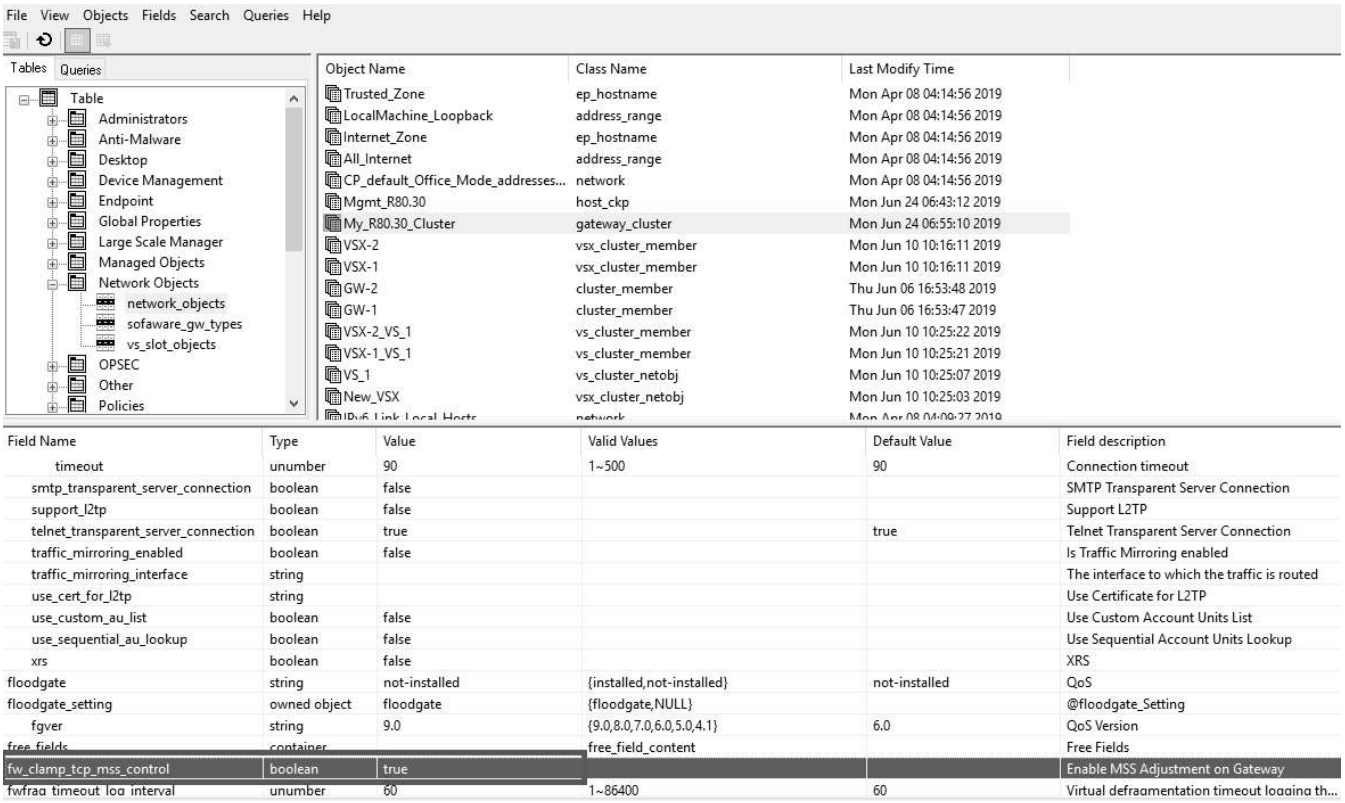
3. To verify that the "clamping" was enabled, connect to command line on Security Gateway and run this command (in Expert Mode):

   [Expert@HostName]# fw ctl get int fw_clamp_tcp_mss

   It should return:
   fw_clamp_tcp_mss = 1

Procedure for R80.20 and higher:

A. Close all SmartConsole windows (SmartDashboard, SmartView Tracker, etc.).
B. Connect with GuiDbEdit Tool to Security Management Server / Domain Management Server.
C. In the left upper pane go to Table - Network Objects - network_objects.
D. Click on the cluster GW object.
E. Press CTRL+F (or go to Search menu - Find) - paste fw_clamp_tcp_mss_control - click on Find Next.
F. In the lower pane, double-click on fw_clamp_tcp_mss field.
G. Change the value from false to true - click OK.

H. 

I. Save the changes: go to File menu - click on Save All.

J. Close the GuiDbEdit Tool.

K. Connect with SmartDashboard to Security Management Server / Domain Management Server.

L. Install the policy onto Security Gateway / Cluster object.

To verify that the "clamping" was enabled, connect to command line on Security Gateway and run this command (in Expert Mode):

[Expert@HostName]# fw ctl get int fw_clamp_tcp_mss

It should return:
fw_clamp_tcp_mss = 1

(3) Kernel Debug

To see the value of ***fw_clamp_tcp_mss*** that Security Gateway applies during policy installation, and to see whether MTU is being set on an interface while the traffic is passing, run the following kernel debug (before policy installation starts and while the traffic is passing):

1. Prepare:

   [Expert@HostName]# fw ctl debug 0
   [Expert@HostName]# fw ctl debug -buf 32000
   [Expert@HostName]# fw ctl debug -m fw + filter if

2. Verify:

   [Expert@HostName]# fw ctl debug -m fw

   Should see:

   Kernel debugging buffer size: 32000KB
   Module: fw
   Enabled Kernel debugging options: error warning filter if

3. Start:

   [Expert@HostName]# fw ctl kdebug -T -f > /var/log/debug.txt

4. Install policy. Let the TCP traffic pass for some time.

5. Stop:

   Press *CTRL+C*, and run this command:
   [Expert@HostName]# fw ctl debug 0

6. Analyze:

   In the debug output file ***/var/log/debug.txt***, search for this line:

   fwk_get_new_global_settings: fw_clamp_tcp_mss param is:

(4) Important Note for R77.10 and lower

On Security Gateways **R77.10 and lower**, it is *not* possible to offer any solution for controlling the Maximum Segment Size (MSS) when ***SecureXL is enabled*** (this was resolved in R77.20).

MSS Clamping is *not* working while SecureXL is enabled. SecureXL accelerates the SYN-ACK packets (does not forward them to the FireWall kernel). Therefore, the TCP Options cannot be changed by Check Point FireWall.

(5) Related solutions

- sk56840 - Explanation of "dropped by fwchain_frag Reason: wait for more fragments"
- sk90200 - Latency when working via Site-to-Site VPN
- sk92465 - Slow Site-to-Site VPN effected by fragmentation error

- sk101219 - New VPN features in R77.20