QUANTUM

14 May 2024

PERFORMANCE TUNING

R81

Administration Guide

CHECK POINT™

Check Point Copyright Notice

Important Information

Latest Software

We recommend that you install the most recent software release to stay up-to-date with the latest functional improvements, stability fixes, security enhancements and protection against new and evolving attacks.

Certifications

For third party independent certification of Check Point products, see the Check Point Certifications page.

Check Point R81

For more about this release, see the R81 home page.

Latest Version of this Document in English

Open the latest version of this document in a Web browser.
Download the latest version of this document in PDF format.

Feedback

Check Point is engaged in a continuous effort to improve its documentation.
Please help us by sending your comments.

Revision History

Date	Description
02 March 2023	Updated: ■ *"Running the 'fw ctl affinity -s' command in Gateway Mode" on page 326* ■ *"Running the 'fw ctl affinity -s' command in VSX Mode" on page 329*
08 February 2023	Updated: ■ *"Configuring Affinity Settings" on page 265*
23 October 2022	Updated: ■ *"Kernel Debug Procedure" on page 432*
16 June 2022	In the HTML version, added glossary terms in the text Updated: ■ *"Default Configuration of CoreXL" on page 256* ■ *"Configuring IPv4 and IPv6 CoreXL Firewall instances" on page 258* ■ *"Configuring Affinity Settings" on page 265* ■ *"Configuring Affinities for Interfaces" on page 275* ■ *"Dynamic Balancing of CoreXL Instances" on page 278* ■ *"fwaccel dos allow" on page 52* (corrected the command name from "`fwaccel dos whitelist`") ■ *"fwaccel dos deny" on page 62* (corrected the command name from "`fwaccel dos blacklist`") ■ *"fwaccel templates" on page 145* ■ *"fw ctl multik prioq" on page 310* ■ *"fwboot ht" on page 347* ■ *"SecureXL Kernel Parameters" on page 403* ■ *"Kernel Debug Modules and Debug Flags" on page 443* Removed: ■ All "`sim`" and "`sim6`" commands as deprecated
11 November 2021	Updated: ■ *"Default Configuration of CoreXL" on page 256*
13 October 2020	First release of this document

Table of Contents

Introduction to Performance Tuning 13

SecureXL 14

Accelerated Features 15

Packet Flow 16

Connection Templates 18

Policy Installation Acceleration 19

Scalable Performance 20

Configuring SecureXL 21

Analyzing the Accelerated Traffic 24

Rate Limiting for DoS Mitigation 25

Introduction 25

Monitoring Events Related to DoS Mitigation 26

Accelerated SYN Defender 27

Introduction 27

Command Line Interface 28

Configuring the 'SYN Attack' protection in SmartConsole 28

SecureXL Commands and Debug 30

Syntax Legend 31

cpview 33

Overview of CPView 33

CPView User Interface 33

Using CPView 34

'fwaccel' and 'fwaccel6' 35

fwaccel cfg 38

fwaccel conns 41

fwaccel dbg 44

fwaccel dos 50

fwaccel dos allow 52
fwaccel dos config 56
fwaccel dos deny 62
fwaccel dos pbox 65
fwaccel dos rate 70
fwaccel dos stats 72
fwaccel feature 74
fwaccel off 77
fwaccel on 81
fwaccel ranges 85
fwaccel stat 92
fwaccel stats 98
Description of the Statistics Counters in the "fwaccel stats" Output 100
Example Outputs of the "fwaccel stats" Commands 109
fwaccel synatk 117
fwaccel synatk -a 119
fwaccel synatk -c <Configuration File> 120
fwaccel synatk -d 121
fwaccel synatk -e 122
fwaccel synatk -g 123
fwaccel synatk -m 124
fwaccel synatk -t <Threshold> 125
fwaccel synatk allow 127
fwaccel synatk config 132
fwaccel synatk monitor 135
fwaccel synatk state 140
fwaccel tab 142
fwaccel templates 145
fwaccel ver 149
fw monitor 150

fw sam_policy 184
fw sam_policy add 187
fw sam_policy batch 200
fw sam_policy del 202
fw sam_policy get 206
The /proc/ppk/ and /proc/ppk6/ entries 212
/proc/ppk/affinity 214
/proc/ppk/conf 215
/proc/ppk/conns 216
/proc/ppk/cpls 217
/proc/ppk/cqstats 218
/proc/ppk/drop_statistics 219
/proc/ppk/ifs 220
/proc/ppk/mcast_statistics 225
/proc/ppk/nac 226
/proc/ppk/notify_statistics 227
/proc/ppk/profile_cpu_stat 229
/proc/ppk/rlc 230
/proc/ppk/statistics 231
/proc/ppk/stats 233
/proc/ppk/viol_statistics 234
SecureXL Debug 235
fwaccel dbg 236
SecureXL Debug Procedure 242
SecureXL Debug Modules and Debug Flags 246
CoreXL 254
Enabling and Disabling CoreXL 255
Default Configuration of CoreXL 256
Configuring IPv4 and IPv6 CoreXL Firewall instances 258
IPv4 and IPv6 CoreXL Firewall Instances 258

Configuring the Number of IPv4 CoreXL Firewall Instances 260
Configuring the Number of IPv6 CoreXL Firewall Instances 261
Example CoreXL Configuration 262
CoreXL Limitations 264
Configuring Affinity Settings 265
Introduction 265
The $FWDIR/conf/fwaffinity.conf Configuration File 265
The $FWDIR/scripts/fwaffinity_apply Script 268
Performance Tuning 269
Allocation of Processing CPU Cores 270
Adding Processing CPU Cores to the Hardware 271
Allocating Additional CPU Cores to the CoreXL SND 272
Allocating a CPU Core for Heavy Logging 273
Configuring Affinities for Interfaces 275
Dynamic Balancing of CoreXL Instances 278
Introduction 278
Syntax 279
Monitoring 282
CoreXL Commands 284
Syntax Legend 285
cp_conf corexl 287
cpconfig 289
cpview 292
Overview of CPView 292
CPView User Interface 292
Using CPView 293
fw ctl multik 294
fw ctl multik add_bypass_port 296
fw ctl multik del_bypass_port 298
fw ctl multik dynamic_dispatching 300

fw ctl multik gconn 301

fw ctl multik get_instance 306

fw ctl multik print_heavy_conn 308

fw ctl multik prioq 310

fw ctl multik show_bypass_ports 311

fw ctl multik stat 312

fw ctl multik start 314

fw ctl multik stop 315

fw ctl multik utilize 316

fw ctl affinity 317

Running the 'fw ctl affinity -l' command in Gateway Mode 318

Running the 'fw ctl affinity -l' command in VSX Mode 323

Running the 'fw ctl affinity -s' command in Gateway Mode 326

Running the 'fw ctl affinity -s' command in VSX Mode 329

fw -i 333

fwboot bootconf 334

fwboot corexl 338

fwboot cpuid 345

fwboot ht 347

fwboot multik_reg 348

fwboot post_drv 350

Multi-Queue 351

Multi-Queue Requirements and Limitations 352

Deciding Whether to Enable the Multi-Queue 354

Multi-Queue Basic Configuration 357

Multi-Queue Configuration in the Expert mode 357

Multi-Queue Configuration in Gaia Clish 362

Multi-Queue Special Scenarios and Configurations 365

Default Number of Active RX Queues 365

Gateway Mode 365

VSX Mode 365
Adding a Network Interface 367
Changing the Affinity of CoreXL Firewall instances 367
Processing Packets that Arrive in the Wrong Order on an Interface that Works in Monitor Mode 367
Multi-Queue Troubleshooting 368
CPView 370
Overview of CPView 370
CPView User Interface 370
Using CPView 371
CPU Spike Detective 373
Command Line Reference 374
Working with Kernel Parameters on Security Gateway 375
Introduction to Kernel Parameters 376
Firewall Kernel Parameters 377
Working with Integer Kernel Parameters 378
Working with String Kernel Parameters 389
SecureXL Kernel Parameters 403
Working with Integer Kernel Parameters 404
Working with String Kernel Parameters 410
Kernel Debug on Security Gateway 417
Kernel Debug Syntax 418
Kernel Debug Filters 427
Kernel Debug Procedure 432
Kernel Debug Procedure with Connection Life Cycle 436
Kernel Debug Modules and Debug Flags 443
Module 'accel_apps' (Accelerated Applications) 445
Module 'accel_pm_mgr' (Accelerated Pattern Match Manager) 446
Module 'APPI' (Application Control Inspection) 447
Module 'BOA' (Boolean Analyzer for Web Intelligence) 449

Module 'CI' (Content Inspection) 450
Module 'cluster' (ClusterXL) 452
Module 'cmi_loader' (Context Management Interface / Infrastructure Loader) 455
Module 'CPAS' (Check Point Active Streaming) 457
Module 'cpcode' (Data Loss Prevention - CPcode) 459
Module 'CPSSH' (SSH Inspection) 461
Module 'crypto' (SSL Inspection) 463
Module 'dlpda' (Data Loss Prevention - Download Agent for Content Awareness) 464
Module 'dlpk' (Data Loss Prevention - Kernel Space) 466
Module 'dlpuk' (Data Loss Prevention - User Space) 467
Module 'DOMO' (Domain Objects) 469
Module 'fg' (FloodGate-1 - QoS) 470
Module 'FILE_SECURITY' (File Inspection) 472
Module 'FILEAPP' (File Application) 473
Module 'fw' (Firewall) 474
Module 'gtp' (GPRS Tunneling Protocol) 481
Module 'h323' (VoIP H.323) 483
Module 'ICAP_CLIENT' (Internet Content Adaptation Protocol Client) 484
Module 'IDAPI' (Identity Awareness API) 486
Module 'kiss' (Kernel Infrastructure) 488
Module 'kissflow' (Kernel Infrastructure Flow) 491
Module 'MALWARE' (Threat Prevention) 492
Module 'multik' (Multi-Kernel Inspection - CoreXL) 493
Module 'MUX' (Multiplexer for Applications Traffic) 495
Module 'NRB' (Next Rule Base) 497
Module 'PSL' (Passive Streaming Library) 499
Module 'RAD_KERNEL' (Resource Advisor - Kernel Space) 500
Module 'RTM' (Real Time Monitoring) 501
Module 'seqvalid' (TCP Sequence Validator and Translator) 503
Module 'SFT' (Stream File Type) 504

Module 'SGEN' (Struct Generator) 505
Module 'synatk' (Accelerated SYN Defender) 506
Module 'TPUTILS' (Threat Prevention Utilities) 507
Module 'UC' (UserCheck) 508
Module 'UP' (Unified Policy) 509
Module 'upconv' (Unified Policy Conversion) 511
Module 'UPIS' (Unified Policy Infrastructure) 512
Module 'VPN' (Site-to-Site VPN and Remote Access VPN) 514
Module 'WS' (Web Intelligence) 517
Module 'WS_SIP' (Web Intelligence VoIP SIP Parser) 520
Module 'WSIS' (Web Intelligence Infrastructure) 522
Glossary 524

Introduction to Performance Tuning

There features improve the performance of Check Point Security Gateway:

- **SecureXL** - accelerates traffic (see *"SecureXL" on page 14*)
- **CoreXL** - runs multiple Firewall instances at the same time (see *"CoreXL" on page 254*)
- **Multi-Queue** - configures multiple traffic queues for each network interface (see *"Multi-Queue" on page 351*)

SecureXL

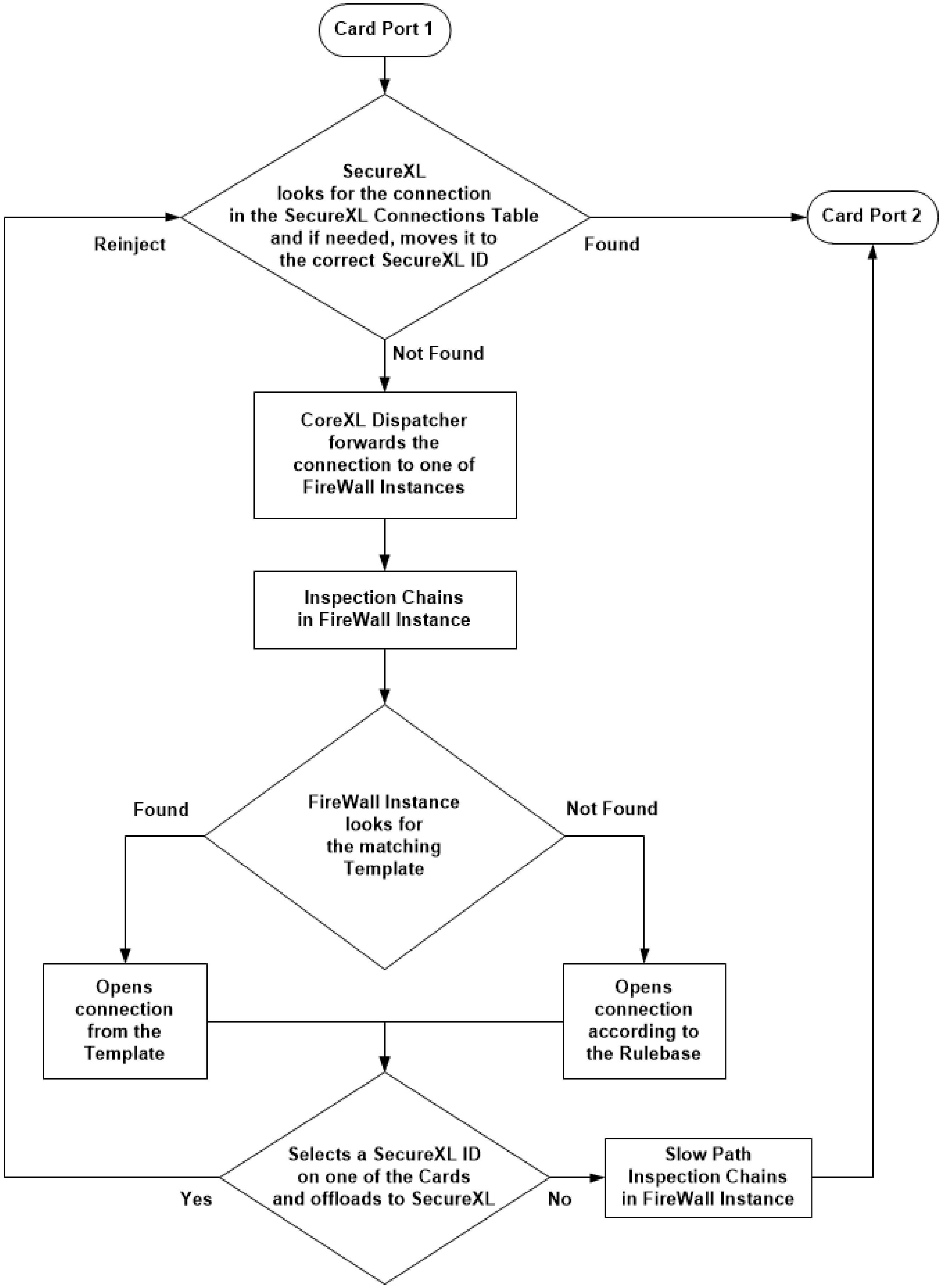
This feature accelerates traffic that passes through a Security Gateway.

Accelerated Features

- Access Control
- Encryption
- NAT
- Software Blades
  - Firewall
  - IPS features
  - Application Control
  - URL Filtering
  - Anti-Virus
  - Anti-Bot
  - Identity Awareness (SecureXL does not create templates for traffic from Identity Agents)
  - VPN Site-to-Site
  - HTTPS Inspection
  - QoS
- Policy installation
- Accounting and logging
- Connection/session rate
- General security checks
- ClusterXL High Availability and Load Sharing
- TCP Sequence Verification
- Dynamic VPN
- Passive streaming
- Active streaming

Packet Flow

This is the general description of the packet flow through the Host Security Appliance:


Card Port 1
SecureXL looks for the connection in the SecureXL Connections Table and if needed, moves it to the correct SecureXL ID
Found
Card Port 2
Reinject
Not Found
CoreXL Dispatcher forwards the connection to one of FireWall Instances
Inspection Chains in FireWall Instance
FireWall Instance looks for the matching Template
Found
Not Found
Opens connection from the Template
Opens connection according to the Rulebase
Selects a SecureXL ID on one of the Cards and offloads to SecureXL
Yes
No
Slow Path Inspection Chains in FireWall Instance

Connection Templates

The Connection Templates feature accelerates the speed, at which new connections from the same source IP address to the same destination IP address and to the same destination port are established.

To achieve the maximum acceleration enhancement, only the Firewall on the Host Security Appliance creates these Connection Templates from active connections according to the Rule Base.

Important - For the list of restrictions that apply to the Connection Templates, see sk32578.

Policy Installation Acceleration

Acceleration is enabled during policy installation.

SecureXL continues to run and stay enabled during a policy installation.

This decreases the load on the Security Gateway's CPU.

Scalable Performance

R80.20 and higher versions include improved SecureXL scalability during high session rate.

As a result, there are no longer limitations on the number of CoreXL SND cores (see *"CoreXL" on page 254*).

Configuring SecureXL

The Gaia First Time Configuration Wizard automatically installs and enables SecureXL on your Security Gateway. No additional configuration is required.

Starting from R80.20, you can disable the SecureXL only *temporarily*.

The SecureXL starts automatically when you start Check Point services (with the `cpstart` command), or reboot the Security Gateway.

Important:

- Disable the SecureXL only for debug purposes, if Check Point Support explicitly instructs you to do so.
- If you disable the SecureXL, this change does **not** survive reboot.
  SecureXL remains disabled until you enable it again on-the-fly, or reboot the Security Gateway.
- If you disable the SecureXL, this change applies only to new connections that arrive after you disabled the acceleration.
  SecureXL continues to accelerate the connections that are already accelerated.
  Other non-connection oriented processing continues to function (for example, virtual defragmentation and VPN decrypt).
- In a Cluster, you must configure all the Cluster Members in the same way.

To disable SecureXL for IPv4 temporarily

Step	Instructions
1	Connect to the command line on your Security Gateway.
2	Log in to Gaia Clish, or Expert mode.
3	Examine the SecureXL status: `fwaccel stat`
4	Disable the SecureXL: `fwaccel off [-a]`
5	Examine the SecureXL status again: `fwaccel stat`

To disable SecureXL for IPv6 temporarily

Step	Instructions
1	Connect to the command line on your Security Gateway.
2	Log in to Gaia Clish, or Expert mode.
3	Examine the SecureXL status: `fwaccel6 stat`
4	Disable the SecureXL: `fwaccel6 off [-a]`
5	Examine the SecureXL status again: `fwaccel6 stat`

To enable SecureXL again for IPv4

Step	Instructions
1	Connect to the command line on your Security Gateway.
2	Log in to Gaia Clish, or Expert mode.
3	Examine the SecureXL status: `fwaccel stat`
4	Enable the SecureXL: `fwaccel on [-a]`
5	Examine the SecureXL status again: `fwaccel stat`

To enable SecureXL again for IPv6

Step	Instructions
1	Connect to the command line on your Security Gateway.
2	Log in to Gaia Clish, or Expert mode.

Step	Instructions
3	Examine the SecureXL status: `fwaccel6 stat`
4	Enable the SecureXL: `fwaccel6 on [-a]`
5	Examine the SecureXL status again: `fwaccel6 stat`

For more information on these commands, see:

- *"fwaccel stat" on page 92*
- *"fwaccel off" on page 77*
- *"fwaccel on" on page 81*

Analyzing the Accelerated Traffic

To capture and analyze the accelerated traffic, use the *"fw monitor" on page 150* command.

Rate Limiting for DoS Mitigation

Introduction

Rate Limiting is a defense against DoS (Denial of Service) attacks.

Rate Limiting rules allow to limit traffic coming from specified sources, or sent to specified destination and using specific services.

Rate limiting is enforced by SecureXL on these:

- Bandwidth and packet rate
- Number of concurrent connections
- Connection rate

For additional information, see sk112454.

Use these commands to configure Rate Limiting for DoS Mitigation:

- "`fw sam_policy`" and "`fw6 sam_policy`" (see *"fw sam_policy" on page 184* - you must use the parameter "`quota <Quota Filter Arguments>`")
- "`fwaccel dos config`" and "`fwaccel6 dos config`" (see *"fwaccel dos config" on page 56*)

Note - You cannot use the Rate Limiting feature for specific URLs. This feature applies to all traffic.

Monitoring Events Related to DoS Mitigation

To see some information related to DoS Mitigation, run these commands:

Command	Description

`fwaccel stats`
`fwaccel6 stats`
	Shows all SecureXL statistics (for IPv4 and IPv6 kernel modules).
	See:
	■ *"fwaccel stats" on page 98*
	■ *"The /proc/ppk/ and /proc/ppk6/ entries" on page 212*

`fwaccel stats -d`
`or`
`cat /proc/ppk/drop_statistics`
`fwaccel6 stats -d`
`or`
`cat /proc/ppk6/drop_statistics`
	Shows SecureXL drop statistics only (for IPv4 and IPv6 kernel modules).
	See:
	■ *"fwaccel stats" on page 98*
	■ *"The /proc/ppk/ and /proc/ppk6/ entries" on page 212*
	■ *"fw sam_policy" on page 184*

`fw samp get -l |\`
`grep '^<[0-9a-f,]*>$' |\`
`xargs fwaccel dos rate get`
`fw samp get -l |\`
`grep '^<[0-9a-f,]*>$' |`
`xargs fwaccel6 dos rate get`
	Shows details of active policy rules in long format (for IPv4 and IPv6 kernel modules).
	See *"fw sam_policy get" on page 206*.

`cat /proc/ppk/rlc`	Shows: ■ Total drop packets ■ Total drop bytes See *"The /proc/ppk/ and /proc/ppk6/ entries" on page 212*.

In addition, see *"SecureXL Debug" on page 235*.

Accelerated SYN Defender

Introduction

A TCP SYN Flood attack occurs when a host, typically with a forged IP address, sends a flood of TCP [SYN] packets. Each of these TCP [SYN] packets is handled as a connection request, which causes the server to create a half-open (unestablished) TCP connection. This occurs because the server sends a TCP [SYN+ACK] packet, and waits for a response TCP packet that does not arrive.

These half-open TCP connections eventually exceed the maximum available TCP connections. This causes a denial of service condition.

The Check Point Accelerated SYN Defender protects the Security Gateway by preventing excessive TCP connections from being created.

The Accelerated SYN Defender uses TCP [SYN] Cookies (particular choices of initial TCP sequence numbers) when under a suspected TCP SYN Flood attack. Using TCP [SYN] Cookies can reduce the load on Security Gateway and on computers behind the Security Gateway. The Accelerated SYN Defender acts as proxy for TCP connections and adjusts TCP {SEQ} and TCP {ACK} values in TCP packets.

This is a sample TCP timeline diagram that shows a TCP connection through the Security Gateway with the enabled Accelerated SYN Defender:

Note - In this example, we assume that there no TCP retransmissions and no early data.

```
              Security Gateway
Client         with Accelerated         Server
   |             SYN Defender              |
   |                   |                   |
   | -(1)--SYN-------> |                   |
   | <---SYN+ACK--(2)- |                   |
   | -(3)--ACK-------> |                   |
   |                   |                   |
   |                  (4)                  |
   |                   |                   |
   |                   | -(5)--SYN-------> |
   |                   | <---SYN+ACK--(6)- |
   |                   | -(7)--ACK-------> |
   |                   |                   |
```

1. A Client sends a TCP [SYN] packet to a Server.
2. The Accelerated SYN Defender replies to the Client with a TCP [SYN+ACK] packet that contains a special cookie in the `Seq` field.

Security Gateway does not maintain the connection state at this time.

3. The Client sends a reply TCP [ACK] packet. This completes the Client-side of the TCP connection.
4. The Accelerated SYN Defender checks if the SYN cookie in the Client's TCP [ACK] packet is legitimate.
5. If the SYN cookie in the Client's TCP [ACK] packet is legitimate, the Accelerated SYN Defender sends a TCP [SYN] packet to the Server to begin the Server-side of the TCP connection.
6. The Server replies with a TCP [SYN+ACK] packet.
7. The Accelerated SYN Defender sends a TCP [ACK] packet to complete the Server-size of the TCP 3-way handshake.
8. The Accelerated SYN Defender marks the TCP connection as established and records the TCP sequence adjustment between the two sides.

SecureXL handles the TCP [SYN] packets. The Host Security Gateway handles the rest of the TCP connection setup.

For each TCP connection the Accelerated SYN Defender establishes, the Security Gateway adjusts the TCP sequence number for the life of that TCP connection.

Command Line Interface

Use the *"fwaccel synatk" on page 117* commands to configure the Accelerated SYN Defender.

Configuring the 'SYN Attack' protection in SmartConsole

The 'SYN Attack' protection is intended for mitigating SYN Flood attacks:

Step	Instructions
1	Connect with SmartConsole to the Management Server.
2	From the left navigation panel, click Security Policies.
3	In the **Shared Policies** section, click **Inspection Settings**.
4	In the top field, search for **SYN Attack**.
5	Double-click on the **SYN Attack** protection.
6	Edit the applicable Inspection profile.

Step	Instructions
7	Configure the applicable settings in the profile: - On the **General Properties** page: If you select **Override with Action** and then **Accept** or **Drop**, it overrides the settings you make on the Security Gateway with the *"fwaccel synatk" on page 117* commands. - On the **Advanced** page: The option you select in the **Activation Settings** (**Protect all interfaces** or **Protect external interfaces only**) overrides the settings you make on the Security Gateway with the *"fwaccel synatk" on page 117* commands.
9	Install the Access Control Policy.

For more information about the **SYN Attack** protection in SmartConsole, see sk120476.

SecureXL Commands and Debug

This section describes:

- SecureXL CLI commands
- SecureXL CLI Debug

Syntax Legend

Whenever possible, this guide lists commands, parameters and options in the alphabetical order.

This guide uses this convention in the Command Line Interface (CLI) syntax:

<table>
<tr><th>Character</th><th>Description</th></tr>
<tr><td>TAB</td><td>Shows the available nested subcommands:
<pre>
main command
→ nested subcommand 1
→ → nested subsubcommand 1-1
→ → nested subsubcommand 1-2
→ nested subcommand 2
</pre>
Example:
<pre>
cpwd_admin
    config
        -a <options>
        -d <options>
        -p
        -r
    del <options>
</pre>
Meaning, you can run only <b>one</b> of these commands:
<ul>
<li>This command:<pre>cpwd_admin config -a <options></pre></li>
<li>Or this command:<pre>cpwd_admin config -d <options></pre></li>
<li>Or this command:<pre>cpwd_admin config -p</pre></li>
<li>Or this command:<pre>cpwd_admin config -r</pre></li>
<li>Or this command:<pre>cpwd_admin del <options></pre></li>
</ul>
</td></tr>
<tr><td>Curly brackets or braces
<b>{ }</b></td><td>Enclose a list of available commands or parameters, separated by the vertical bar <b>|</b>.
User can enter only one of the available commands or parameters.</td></tr>
</table>

Character	Description
Angle brackets **< >**	Enclose a variable. User must explicitly specify a supported value.
Square brackets or brackets **[ ]**	Enclose an optional command or parameter, which user can also enter.

cpview

Overview of CPView

Description

CPView is a text based built-in utility on a Check Point computer.

CPView Utility shows statistical data that contain both general system information (CPU, Memory, Disk space) and information for different Software Blades (only on Security Gateway).

The CPView continuously updates the data in easy to access views.

On Security Gateway, you can use this statistical data to monitor the performance.

For more information, see sk101878.

Syntax

```
cpview --help
```

CPView User Interface

The CPView user interface has three sections:

Section	Description
Header	This view shows the time the statistics in the third view are collected. It updates when you refresh the statistics.
Navigation	This menu bar is interactive. Move between menus with the arrow keys and mouse. A menu can have sub-menus and they show under the menu bar.
View	This view shows the statistics collected in that view. These statistics update at the refresh rate.

Using CPView

Use these keys to navigate the CPView:

Key	Description
Arrow keys	Moves between menus and views. Scrolls in a view.
Home	Returns to the **Overview** view.
Enter	Changes to the **View Mode**. On a menu with sub-menus, the **Enter** key moves you to the lowest level sub-menu.
Esc	Returns to the **Menu Mode**.
Q	Quits CPView.

Use these keys to change CPView interface options:

Key	Description
R	Opens a window where you can change the refresh rate. The default refresh rate is 2 seconds.
W	Changes between wide and normal display modes. In wide mode, CPView fits the screen horizontally.
S	Manually sets the number of rows or columns.
M	Switches on/off the mouse.
P	Pauses and resumes the collection of statistics.

Use these keys to save statistics, show help, and refresh statistics:

Key	Description
C	Saves the current page to a file. The file name format is: `cpview_<ID of the cpview process>.cap<Number of the capture>`
H	Shows a tooltip with CPView options.
Space bar	Immediately refreshes the statistics.

'fwaccel' and 'fwaccel6'

Description

The *fwaccel* commands control the acceleration for IPv4 traffic.

The *fwaccel6* commands control the acceleration for IPv6 traffic.

Syntax for IPv4

```
fwaccel help
```

```
fwaccel [-i <SecureXL ID>]
      cfg <options>
      conns <options>
      dbg <options>
      dos <options>
            feature <options>
      off <options>
      on <options>
      ranges <options>
      stat <options>
      stats <options>
      synatk <options>
      tab <options>
      templates <options>
      ver
```

Syntax for IPv6

```
fwaccel6 help
```

```
fwaccel6
      conns <options>
      dbg <options>
      dos <options>
            feature <options>
      off <options>
      on <options>
      ranges <options>
      stat <options>
      stats <options>
      synatk <options>
      tab <options>
      templates <options>
      ver
```

Parameters and Options

Parameter and Options	Description
`help`	Shows the built-in help.
`-i <SecureXL ID>`	Specifies the SecureXL instance ID (for IPv4 only).
`cfg <options>`	Controls the SecureXL acceleration parameters (for IPv4 only). See *"fwaccel cfg" on page 38*.
`conns <options>`	Shows all connections that pass through SecureXL. See *"fwaccel conns" on page 41*.
`dbg <options>`	Controls the *"SecureXL Debug" on page 235*. See *"fwaccel dbg" on page 236*.
`dos <options>`	Controls the Rate Limiting for DoS Mitigation in SecureXL. See *"fwaccel dos" on page 50*.
`feature <options>`	Controls the specified SecureXL features. See *"fwaccel feature" on page 74*.
`off <options>`	Stops the acceleration on-the-fly. This does **not** survive reboot. See *"fwaccel off" on page 77*.

Parameter and Options	Description
`on <options>`	Starts the acceleration on-the-fly, if it was previously stopped. See *"fwaccel on" on page 81*.
`ranges <options>`	Shows the loaded ranges. See *"fwaccel ranges" on page 85*.
`stat <options>`	Shows the SecureXL status. See *"fwaccel stat" on page 92*.
`stats <options>`	Shows the acceleration statistics. See *"fwaccel stats" on page 98*.
`synatk <options>`	Controls the Accelerated SYN Defender. See *"fwaccel synatk" on page 117*.
`tab <options>`	Shows the contents of the specified SecureXL table. See *"fwaccel tab" on page 142*.
`templates <options>`	Shows the SecureXL templates. See *"fwaccel templates" on page 145*.
`ver`	Shows the SecureXL and FireWall version. See *"fwaccel ver" on page 149*.

fwaccel cfg

Description

The *fwaccel cfg* command controls the SecureXL acceleration parameters.

Important - In a Cluster, you must configure all the Cluster Members in the same way.

Syntax

```
fwaccel cfg
      -h
      -a {<Number of Interface> | <Name of Interface> | reset}
      -b {on | off}
      -c <Number>
      -d <Number>
      -e <Number>
      -i {on | off}
      -l <Number>
      -m <Seconds>
      -p {on | off}
      -r <Number>
      -v <Seconds>
      -w {on | off}
```

Important:

- These commands do not provide output. You cannot see the currently configured values.
- Changes made with these commands do **not** survive reboot.

Parameters

Parameter	Description
`-h`	Shows the applicable built-in help.

Parameter	Description

`-a` *`<Number of Interface>`*
`-a` *`<Name of Interface>`*
`-a reset`
	■ `-a` *`<Number of Interface>`*
	Configures the SecureXL not to accelerate traffic on the interface specified by its internal number in Check Point kernel.
	■ `-a` *`<Name of Interface>`*
	Configures the SecureXL not to accelerate traffic on the interface specified by its name.
	■ `-a reset`
	Configures the SecureXL to accelerate traffic on all interfaces (resets the non-accelerated configuration).
	Notes:
	■ This command does not support Falcon Acceleration Cards.
	■ To see the required information about the interfaces, run these commands in the specified order:
	`fw getifs`
	`fw ctl iflist`
	■ To see if the "`fwaccel cfg -a ...`" command failed, run this command:
	`tail -n 10 /var/log/messages`

`-b {on | off}`	Controls the SecureXL Drop Templates match (sk66402): ■ `on` - Enables the SecureXL Drop Templates match ■ `off` - Disables the SecureXL Drop Templates match **Note** - In R81, SecureXL does not support this parameter yet..
`-c` *`<Number>`*	Configures the maximal number of connections, when SecureXL disables the templates.
`-d` *`<Number>`*	Configures the maximal number of delete retries.
`-e` *`<Number>`*	Configures the maximal number of general errors.
`-i {on | off}`	Configures SecureXL to ignore API version mismatch: ■ `on` - Ignore API version mismatch. ■ `off` - Do not ignore API version mismatch (this is the default).

Parameter	Description
`-l <Number>`	Configures the maximal number of entries in the SecureXL templates database. Valid values are: - 0 - To disable the limit (this is the default). - Between 10 and 524288 - To configure the limit. **Important** - If you configure a limit, you must stop and start the acceleration for this change to take effect. Run the *"fwaccel off" on page 77* command and then the *"fwaccel on" on page 81* command.
`-m <Seconds>`	Configures the timeout for entries in the SecureXL templates database. Valid values are: - 0 - To disable the timeout (this is the default). - Between 10 and 524288 - To configure the timeout.
`-p {on | off}`	Configures the offload of Connection Templates (if possible): - `on` - Enables the offload of new templates (this is the default). - `off` - Disables the offload of new templates.
`-r <Number>`	Configures the maximal number of retries for SecureXL API calls.
`-v <Seconds>`	Configures the interval between SecureXL statistics request. Valid values are: - 0 - To disable the interval. - 1 and greater - To configure the interval.
`-w {on | off}`	Configures the support for warnings about the IPS protection **Sequence Verifier**: - `on` - Enable the support for these warnings. - `off` - Disables the support for these warnings.

fwaccel conns

Description

The *fwaccel conns* and *fwaccel6 conns* commands show the list of the SecureXL connections on the local Security Gateway, or Cluster Member.

Warning - If the number of concurrent connections is large, when you run these commands, they can consume memory and CPU at very high level (see sk118716).

Syntax for IPv4

```
fwaccel [-i <SecureXL ID>] conns
      -h
      -f <filter>
      -m <Number of Entries>
      -s
```

Syntax for IPv6

```
fwaccel6 conns
      -h
      -f <Filter>
      -m <Number of Entries>
      -s
```

Parameters

Parameter	Description
`-h`	Shows the applicable built-in help.
`-i <SecureXL ID>`	Specifies the SecureXL instance ID (for IPv4 only).

Parameter	Description

`-f <Filter>`	Show the SecureXL Connections Table entries based on the specified filter flags.
	Notes:
	■ To see the available filter flags, run:
	`fwaccel conns -h`
	■ Each filter flag is one letter - capital, or small.
	■ You can specify more than one flag.
	For example:
	`fwaccel conns -f AaQq`
	Available filter flags are:
	■ `A` - Shows accounted connections (for which SecureXL counted the number of packets and bytes).
	■ `a` - Shows not accounted connections.
	■ `C` - Shows encrypted (VPN) connections.
	■ `c` - Shows clear-text (not encrypted) connections.
	■ `F` - Shows connections that SecureXL forwarded to Firewall.
	Note - In R81, SecureXL does not support this parameter.
	■ `f` - Shows cut-through connections (which SecureXL accelerated).
	Note - In R81, SecureXL does not support this parameter.
	■ `H` - Shows connections offloaded to the SAM card.
	Note - R81, does not support the SAM card (Known Limitation PMTR-18774).
	■ `h` - Shows connections created in the SAM card.
	Note - R81, does not support the SAM card (Known Limitation PMTR-18774).
	■ `L` - Shows connections, for which SecureXL created internal links.
	■ `l` - Shows connections, for which SecureXL did not create internal links.
	■ `N` - Shows connections that undergo NAT.
	Note - In R81, SecureXL does not support this parameter.
	■ `n` - Shows connections that do not undergo NAT.
	Note - R81, SecureXL does not support this parameter.
	■ `Q` - Shows connections that undergo QoS.
	■ `q` - Shows connections that do not undergo QoS.
	■ `S` - Shows connections that undergo PXL.
	■ `s` - Shows connections that do not undergo PXL.
	■ `U` - Shows unidirectional connections.
	■ `u` - Shows bidirectional connections.

Parameter	Description
`-m <Number of Entries>`	Specifies the maximal number of connections to show. **Note** - In R81, SecureXL does not support this parameter.
`-s`	Shows the summary of SecureXL Connections Table (number of connections). **Warning** - Depending on the number of current connections, might consume memory at very high level.

Example - Default output from a non-VSX Gateway

```
[Expert@MyGW:0]# fwaccel conns
Source          SPort Destination     DPort PR Flags     C2S i/f S2C i/f   Inst Identity
--------------- ----- --------------- ----- -- ----------- ------- ------- ---- -------
      1.1.1.200 50586       1.1.1.100 18191  6 F............. 2/2     2/-     3       0
  192.168.0.244 35925   192.168.0.242 18192  6 F............. 1/1     -/-     1       0
   192.168.0.93   257   192.168.0.242 53932  6 F............. 1/1     1/-     0       0
  192.168.0.242    22   172.30.168.15 57914  6 F............. 1/1     -/-     2       0
  192.168.0.244 34773   192.168.0.242 18192  6 F............. 1/1     -/-     2       0
   192.168.0.88   138   192.168.0.255   138 17 F............. 1/1     -/-     0       0
      1.1.1.100 18191       1.1.1.200 55336  6 F............. 2/2     2/-     4       0
  192.168.0.242 18192   192.168.0.244 38567  6 F............. 1/1     -/-     4       0
  192.168.0.242 53932    192.168.0.93   257  6 F............. 1/1     1/-     0       0
  192.168.0.242 18192   192.168.0.244 62714  6 F............. 1/1     -/-     1       0
  192.168.0.244 33558   192.168.0.242 18192  6 F............. 1/1     -/-     5       0
      1.1.1.200 36359       1.1.1.100 18191  6 F............. 2/2     2/-     5       0
      1.1.1.200 55336       1.1.1.100 18191  6 F............. 2/2     2/-     4       0
  192.168.0.242 60756    192.168.0.93   257  6 F............. 1/1     1/-     4       0
      1.1.1.100 18191       1.1.1.200 36359  6 F............. 2/2     2/-     5       0
      1.1.1.100 18191       1.1.1.200 50586  6 F............. 2/2     2/-     3       0
  192.168.0.244 38567   192.168.0.242 18192  6 F............. 1/1     -/-     4       0
  192.168.0.242 18192   192.168.0.244 32877  6 F............. 1/1     -/-     5       0
  192.168.0.242 53806   192.168.47.45    53 17 F............. 1/1     1/-     3       0
  192.168.0.242 18192   192.168.0.244 33558  6 F............. 1/1     -/-     5       0
  172.30.168.15 57914   192.168.0.242    22  6 F............. 1/1     -/-     2       0
  192.168.0.255   138    192.168.0.88   138 17 F............. 1/1     -/-     0       0
   192.168.0.93   257   192.168.0.242 60756  6 F............. 1/1     1/-     4       0
      1.1.1.200 18192       1.1.1.100 37964  6 F............. 2/2     -/-     1       0
      1.1.1.100 37964       1.1.1.200 18192  6 F............. 2/2     -/-     1       0
  192.168.0.244 32877   192.168.0.242 18192  6 F............. 1/1     -/-     5       0
  192.168.0.242 18192   192.168.0.244 34773  6 F............. 1/1     -/-     2       0
  192.168.0.242 18192   192.168.0.244 35925  6 F............. 1/1     -/-     1       0
  192.168.47.45    53   192.168.0.242 53806 17 F............. 1/1     1/-     3       0
  192.168.0.244 62714   192.168.0.242 18192  6 F............. 1/1     -/-     1       0

Idx Interface
--- ---------
  0 lo
  1 eth0
  2 eth1

Total number of connections: 30
[Expert@MyGW:0]#
```

fwaccel dbg

Description

The *fwaccel dbg* command controls the SecureXL debug. See *"SecureXL Debug Procedure" on page 242*.

Important - In a Cluster, you must configure all the Cluster Members in the same way.

Syntax in Gaia Clish or the Expert mode on a Security Gateway / ClusterXL:

```
fwaccel dbg
      -h
      -m <Name of SecureXL Debug Module>
      all
      + <Debug Flags>
      - <Debug Flags>
      reset
      -f {"<5-Tuple Debug Filter>" | reset}
      list
      resetall
```

Parameters

Parameter	Description
`-h`	Shows the applicable built-in help.
`-m <Name of SecureXL Debug Module>`	Specifies the name of the SecureXL debug module. To see the list of available debug modules, run: `fwaccel dbg`
`all`	Enables all debug flags for the specified debug module.
`+ <Debug Flags>`	Enables the specified debug flags for the specified debug module: Syntax: `+ Flag1 [Flag2 Flag3 ... FlagN]` **Note** - You must press the space bar key after the plus (+) character.

Parameter	Description
`- <Debug Flags>`	Disables all debug flags for the specified debug module. Syntax: `- Flag1 [Flag2 Flag3 ... FlagN]` **Note** - You must press the space bar key after the minus (-) character.
`reset`	Resets all debug flags for the specified debug module to their default state.
`-f "<5-Tuple Debug Filter>"`	Configures the debug filter to show only debug messages that contain the specified connection. The filter is a string of five numbers separated with commas: `"<Source IP Address>,<Source Port>,<Destination IP Address>,<Destination Port>,<Protocol Number>"` **Notes:** ■ You can configure only one debug filter at one time. ■ You can use the asterisk "`*`" as a wildcard for an IP Address, Port number, or Protocol number. ■ For more information, see *IANA Service Name and Port Number Registry* and *IANA Protocol Numbers*.
`-f reset`	Resets the current debug filter.
`list`	Shows all enabled debug flags in all debug modules.
`resetall`	Reset all debug flags for all debug modules to their default state.

Examples

Example 1 - Default output

```
[Expert@MyGW:0]# fwaccel dbg
Usage: fwaccel dbg [-m <...>] [resetall | reset | list | all | +/- <flags>]
   -m <module>            - module of debugging
   -h                     - this help message
   resetall               - reset all debug flags for all modules
   reset                  - reset all debug flags for module
   all                    - set all debug flags for module
   list                   - list all debug flags for all modules
   -f reset | "<5-tuple>"  - filter debug messages
   + <flags>              - set the given debug flags
   - <flags>              - unset the given debug flags

List of available modules and flags:

Module: default (default)
err init drv tag lock cpdrv routing kdrv gtp tcp_sv gtp_pkt svm iter conn htab del update
acct conf stat queue ioctl corr util rngs relations ant conn_app rngs_print infra_ids offload
nat

Module: db
err get save del tmpl tmo init ant profile nmr nmt

Module: api
err init add update del acct conf stat vpn notif tmpl sv pxl qos gtp infra tmpl_info upd_conf
upd_if_inf add_sa del_sa del_all_sas misc get_features get_tab get_stat reset_stat tag long_
ver del_all_tmpl get_state upd_link_sel

Module: pkt
err f2f frag spoof acct notif tcp_state tcp_state_pkt sv cpls routing drop pxl qos user
deliver vlan pkt nat wrp corr caf

Module: infras
err reorder pm

Module: tmpl
err dtmpl_get dtmpl_notif tmpl

Module: vpn
err vpnpkt linksel routing vpn

Module: nac
err db db_get pkt pkt_ex signature offload idnt ioctl nac

Module: cpaq
init client server exp cbuf opreg transport transport_utils error

Module: synatk
init conf conn err log pkt proxy state msg

Module: adp
err rt nh eth heth wrp inf mbs bpl bplinf mbeinf if drop bond xmode ipsctl xnp

Module: dos
fw1-cfg fw1-pkt sim-cfg sim-pkt err detailed drop

[Expert@MyGW:0]#
```

Example 2 - Enabling and disabling of debug flags

```
[Expert@MyGW:0]# fwaccel dbg -m default + err conn
Debug flags updated.
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dbg list

Module: default (2001)
err conn

Module: db (1)
err

Module: api (1)
err

Module: pkt (1)
err

Module: infras (1)
err

Module: tmpl (1)
err

Module: vpn (1)
err

Module: nac (1)
err

Module: cpaq (100)
error

Module: synatk (0)


Module: adp (1)
err

Module: dos (10)
err

Debug filter not set.
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dbg -m default - conn
Debug flags updated.
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dbg list

Module: default (1)
err

Module: db (1)
err

Module: api (1)
err

Module: pkt (1)
err

Module: infras (1)
err

Module: tmpl (1)
```

```
err

Module: vpn (1)
err

Module: nac (1)
err

Module: cpaq (100)
error

Module: synatk (0)


Module: adp (1)
err

Module: dos (10)
err

Debug filter not set.
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dbg -m default reset
Debug flags updated.
[Expert@MyGW:0]#
```

Example 3 - Resetting all debug flags in all debug modules

```
[Expert@MyGW:0]# fwaccel dbg resetall
Debug state was reset to default.
[Expert@MyGW:0]#
```

Example 4 - Configuring debug filter for an SSH connection from 192.168.20.30 to 172.16.40.50

```
[Expert@MyGW:0]# fwaccel dbg -f 192.168.20.30,*,172.16.40.50,22,6
Debug filter was set.
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dbg list

... ...

Debug filter: "<*,*,*,*,*>"
[Expert@MyGW:0]#
```

fwaccel dos

Description

The *fwaccel dos* and *fwaccel6 dos* commands control the Rate Limiting for DoS mitigation techniques in SecureXL on the local Security Gateway, or Cluster Member. See *"Rate Limiting for DoS Mitigation" on page 25*.

Important:

- In VSX mode, you must go to the context of an applicable Virtual System.
  - In Gaia Clish, run: `set virtual-system <VSID>`
  - In the Expert mode, run: `vsenv <VSID>`
- In a Cluster, you must configure all the Cluster Members in the same way.

Syntax for IPv4

```
fwaccel dos
      allow <options>
      config <options>
      deny <options>
      pbox <options>
      rate <options>
      stats <options>
```

Syntax for IPv6

```
fwaccel6 dos
      allow <options>
      config <options>
      deny <options>
      pbox <options>
      rate <options>
      stats <options>
```

Parameters

Parameter	Description
`allow <options>`	Configures the allow-list for source IP addresses in the SecureXL Penalty Box. See *"fwaccel dos allow" on page 52*.

Parameter	Description
`config <options>`	Controls the DoS mitigation configuration in SecureXL. See *"fwaccel dos config" on page 56*.
`deny <options>`	Controls the IP deny-list in SecureXL. See *"fwaccel dos deny" on page 62*.
`pbox <options>`	Controls the Penalty Box whitelist in SecureXL. See *"fwaccel dos pbox" on page 65*.
`rate <options>`	Shows and installs the Rate Limiting policy in SecureXL. See *"fwaccel dos rate" on page 70*.
`stats <options>`	Shows and clears the DoS real-time statistics in SecureXL. See *"fwaccel dos stats" on page 72*.

fwaccel dos allow

Description

The *fwaccel dos allow* command configures the allow-list for source IP addresses in the SecureXL Penalty Box.

This allow-list overrides which packet the SecureXL Penalty Box drops.

Important:

- This command supports only IPv4.
- In VSX mode, you must go to the context of an applicable Virtual System.In Gaia Clish, run: `set virtual-system <VSID>`In the Expert mode, run: `vsenv <VSID>`
- In a Cluster, you must configure all the Cluster Members in the same way.
- This allow-list overrides entries in the blacklist.
  Before you use a 3rd-party or automatic blacklists, add trusted networks and hosts to the allow-list to avoid outages.
- This allow-list unblocks IP Options and IP fragments from trusted sources when you explicitly configure one these SecureXL features:
  - `--enable-drop-opts`
  - `--enable-drop-frags`

  See the *"fwaccel dos config" on page 56* command.

Notes:

- To allow-list the Rate Limiting policy, refer to the bypass action of the `fw samp` command.
  For example, `fw samp -a b ...`
  For more information about the `fw sam_policy` command, see *"fw sam_policy" on page 184*.
- This command is similar to the "`fwaccel dos pbox allow`" command (see *"fwaccel dos pbox" on page 65*).
- Also, see the *"fwaccel synatk allow" on page 127* command.

Syntax for IPv4

```
fwaccel dos allow
       -a <IPv4 Address>[/<Subnet Prefix>]
       -d <IPv4 Address>[/<Subnet Prefix>]
       -F
       -l /<Path>/<Name of File>
       -L
       -s
```

Parameters

Parameter	Description
No Parameters	Shows the applicable built-in usage.

`-a <IPv4 Address> [/<Subnet Prefix>]`	Adds the specified IP address to the Penalty Box allow-list.
	■ `<IPv4 Address>`
	Can be an IPv4 address of a network or a host.
	■ `<Subnet Prefix>`
	Must specify the length of the subnet mask in the format `/<bits>`.
	Optional for a host IPv4 address.
	Mandatory for a network IPv4 address.
	Range - from /1 to /32.
	Important - If you do not specify the subnet prefix explicitly, this command uses the subnet prefix /32.
	Examples:
	■ For a host:
	`192.168.20.30`
	`192.168.20.30/32`
	■ For a network:
	`192.168.20.0/24`

`-d <IPv4 Address> [/<Subnet Prefix>]`	Removes the specified IPv4 address from the Penalty Box allow-list. ■ `<IPv4 Address>` Can be an IPv4 address of a network or a host. ■ `<Subnet Prefix>` Optional. Must specify the length of the subnet mask in the format `/<bits>`. Optional for a host IPv4 address. Mandatory for a network IPv4 address. Range - from /1 to /32. **Important** - If you do not specify the subnet prefix explicitly, this command uses the subnet prefix /32.
`-F`	Removes (flushes) all entries from the Penalty Box allow-list.

Parameter	Description
`-l /<Path>/<Name of File>`	Loads the Penalty Box allow-list entries from the specified plain-text file. **Note** - To replace the current allow-list with the contents of a new file, use both the "`-F`" and "`-l`" parameters on the same command line. **Important:** ■ You must manually create and configure this file with the `touch` or `vi` command. ■ You must assign at least the read permission to this file with the `chmod +x` command. ■ Each entry in this file must be on a separate line. ■ Each entry in this file must be in this format: `<IPv4 Address>[/<Subnet Prefix>]` ■ SecureXL ignores empty lines and lines that start with the # character in this file.
`-L`	Loads the Penalty Box allow-list entries from the plain-text file with a predefined name: `$FWDIR/conf/pbox-allow-list-v4.conf` Security Gateway automatically runs this command "`fwaccel dos pbox allow -L`" during each boot. **Note** - To replace the current allow-list with the contents of a new file, use both the "`-F`" and "`-L`" parameters on the same command line. **Important:** ■ This file does not exist by default. ■ You must manually create and configure this file with the `touch` or `vi` command. ■ You must assign at least the read permission to this file with the `chmod +x` command. ■ Each entry in this file must be on a separate line. ■ Each entry in this file must be in this format: `<IPv4 Address>[/<Subnet Prefix>]` ■ SecureXL ignores empty lines and lines that start with the # character in this file.
`-s`	Shows the current Penalty Box allow-list entries.

Example 1 - Adding a host IP address without optional subnet prefix

```
[Expert@MyGW:0]# fwaccel dos allow -a 192.168.20.40
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos allow -s
192.168.20.40/32
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos allow -F
[Expert@MyGW:0]# fwaccel dos allow -s
[Expert@MyGW:0]#
```

Example 2 - Adding a host IP address with optional subnet prefix

```
[Expert@MyGW:0]# fwaccel dos allow -a 192.168.20.40/32
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos allow -s
192.168.20.40/32
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos allow -F
[Expert@MyGW:0]# fwaccel dos allow -s
[Expert@MyGW:0]#
```

Example 3 - Adding a network IP address with mandatory subnet prefix

```
[Expert@MyGW:0]# fwaccel dos allow -a 192.168.20.0/24
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos allow -s
192.168.20.0/24
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos allow -F
[Expert@MyGW:0]# fwaccel dos allow -s
[Expert@MyGW:0]#
```

Example 4 - Deleting an entry

```
[Expert@MyGW:0]# fwaccel dos allow -a 192.168.20.40/32
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos allow -a 192.168.20.70/32
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos allow -s
192.168.20.40/32
192.168.20.70/32
[Expert@MyGW:0]# fwaccel dos allow -d 192.168.20.70/32
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos allow -s
192.168.20.40/32
[Expert@MyGW:0]#
```

fwaccel dos config

Description

The *fwaccel dos config* and *fwaccel6 dos config* commands control the global configuration parameters of the Rate Limiting for DoS mitigation in SecureXL.

These global parameters apply to all configured Rate Limiting rules.

Important:

- In VSX mode, you must go to the context of an applicable Virtual System.In Gaia Clish, run: `set virtual-system <VSID>`In the Expert mode, run: `vsenv <VSID>`
- In a Cluster, you must configure all the Cluster Members in the same way.

Syntax for IPv4

```
fwaccel dos config
      get
      set
            {--disable-blacklists | --enable-blacklists}
            {--disable-drop-frags | --enable-drop-frags}
            {--disable-drop-opts | --enable-drop-opts}
            {--disable-internal | --enable-internal}
            {--disable-log-drops | --enable-log-drops}
            {--disable-log-pbox | --enable-log-pbox}
            {--disable-monitor | --enable-monitor}
            {--disable-pbox | --enable-pbox}
            {--disable-rate-limit | --enable-rate-limit}
            {--disable-rule-cache | --enable-rule-cache}
            {-n <NOTIF_RATE> | --notif-rate <NOTIF_RATE>}
            {-p <PBOX_RATE> | --pbox-rate <PBOX_RATE>}
            {-t <PBOX_TMO> | --pbox-tmo <PBOX_TMO>}
```

Syntax for IPv6

```
fwaccel6 dos config
      get
      set
            {--disable-blacklists | --enable-blacklists}
            {--disable-drop-frags | --enable-drop-frags}
            {--disable-drop-opts | --enable-drop-opts}
            {--disable-internal | --enable-internal}
            {--disable-log-drops | --enable-log-drops}
            {--disable-log-pbox | --enable-log-pbox}
            {--disable-monitor | --enable-monitor}
            {--disable-pbox | --enable-pbox}
            {--disable-rate-limit | --enable-rate-limit}
            {--disable-rule-cache | --enable-rule-cache}
            {-n <NOTIF_RATE> | --notif-rate <NOTIF_RATE>}
            {-p <PBOX_RATE> | --pbox-rate <PBOX_RATE>}
            {-t <PBOX_TMO> | --pbox-tmo <PBOX_TMO>}
```

Parameters and Options

Parameter or Option	Description
No Parameters	Shows the applicable built-in usage.
`get`	Shows the configuration parameters.
`set <options>`	Configuration the parameters.
`--disable-blacklists`	Disables the IP blacklists. This is the default configuration.
`--disable-drop-frags`	Disables the drops of all fragmented packets. This is the default configuration. **Important** - This option applies to only VSX, and only for traffic that arrives at a Virtual System through a Virtual Switch (packets received through a Warp interface). From R80.20, IP Fragment reassembly occurs in SecureXL before the Warp-jump from a Virtual Switch to a Virtual System. To block IP fragments, the Virtual Switch must be configured with this option. Otherwise, this has no effect, because the IP fragments would already be reassembled when they arrive at the Virtual System's Warp interface.

Parameter or Option	Description
`--disable-drop-opts`	Disables the drops of all packets with IP options. This is the default configuration.
`--disable-internal`	Disables the enforcement on internal interfaces. This is the default configuration.
`--disable-log-drops`	Disables the notifications when the DoS module drops a packet due to rate limiting policy.
`--disable-log-pbox`	Disables the notifications when administrator adds an IP address to the penalty box.
`--disable-monitor`	Disables the monitor-only mode. This is the default configuration. This command affects all Rate Limiting features. Also, see the *"fwaccel dos deny" on page 62* command.
`--disable-pbox`	Disables the IP penalty box. This is the default configuration. Also, see the *"fwaccel dos pbox" on page 65* command.
`--disable-rate-limit`	Disables the enforcement of the rate limiting policy. This is the default configuration.
`--disable-rule-cache`	Disables the caching of Rate Limiting rule matches. This optimizes the performance for large numbers of connections-per-second.
`--enable-blacklists`	Enables IP blacklists. Also, see the *"fwaccel dos deny" on page 62* command.
`--enable-drop-frags`	Enables the drops of all fragmented packets.
`--enable-drop-opts`	Enables the drops of all packets with IP options.
`--enable-internal`	Enables the enforcement on internal interfaces.
`--enable-log-drops`	Enables the notifications when the DoS module drops a packet due to rate limiting policy. This is the default configuration.

Parameter or Option	Description
`--enable-log-pbox`	Enables the notifications when administrator adds an IP address to the penalty box. This is the default configuration.
`--enable-monitor`	Enables the monitor-only mode (accepts all packets that otherwise are dropped). This command affects all Rate Limiting features. Also, see the *"fwaccel dos deny" on page 62* command.
`--enable-pbox`	Enables the IP penalty box. Also, see the *"fwaccel dos pbox" on page 65* command.
`--enable-rate-limit`	Enables the enforcement of the rate limiting policy. ℹ **Important** - After you run this command, you must install the Access Control policy.
`--enable-rule-cache`	Enables the caching of Rate Limiting rule matches. This optimizes the performance for large numbers of packets-per-connection. This is the default configuration.

`-n <NOTIF_RATE>`
`--notif-rate <NOTIF_RATE>`
	Configures the maximal number of drop notifications per second for each SecureXL device.
	Range: 0 - (2^{32}-1)
	Default: 100

`-p <PBOX_RATE>`
`--pbox-rate <PBOX_RATE>`
	Configures the minimal number of reported dropped packets before SecureXL adds a source IPv4 address to the penalty box.
	Range: 0 - (2^{32}-1)
	Default: 500

`-t <PBOX_TMO>`
`--pbox-tmo <PBOX_TMO>`
	Configures the number of seconds until SecureXL removes an IP is from the penalty box.
	Range: 0 - (2^{32}-1)
	Default: 180

Example 1 - Get the current DoS configuration on a non-VSX Gateway

```
[Expert@MyGW:0]# fwaccel dos config get
   rate limit: disabled (without policy)
         pbox: disabled
   blacklists: disabled
log blacklist: disabled
   drop frags: disabled
    drop opts: disabled
     internal: disabled
      monitor: disabled
    log drops: disabled
     log pbox: disabled
   notif rate: 100 notifications/second
    pbox rate: 500 packets/second
     pbox tmo: 180 seconds
[Expert@MyGW:0]#
```

Example 2 - Enabling the Penalty Box on a non-VSX Gateway

```
[Expert@MyGW:0]# fwaccel dos config set --enable-pbox
OK
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos config get
   rate limit: disabled (without policy)
         pbox: enabled
   blacklists: disabled
   drop frags: disabled
    drop opts: disabled
     internal: disabled
      monitor: disabled
    log drops: enabled
     log pbox: enabled
   notif rate: 100 notifications/second
    pbox rate: 500 packets/second
     pbox tmo: 180 seconds
[Expert@MyGW:0]#
```

Making the configuration persistent

The settings defined with the "`fwaccel dos config set`" and the "`fwaccel6 dos config set`" commands return to their default values during each reboot. To make these settings persistent, add the applicable commands to these configuration files:

File	Description

`$FWDIR/conf/fwaccel_dos_rate_on_install`	This shell script for IPv4 must contain only the "`fwaccel dos config set`" commands:
	`#!/bin/bash`
	`fwaccel dos config set <options>`

`$FWDIR/conf/fwaccel6_dos_rate_on_install`	This shell script for IPv6 must contain only the "`fwaccel6 dos config set`" commands:
	`#!/bin/bash`
	`fwaccel6 dos config set <options>`

Important - Do not include the *"fw sam_policy" on page 184* commands in these configuration files. The configured Rate Limiting policy survives reboot. If you add the "`fw sam_policy`" commands, the rate policy installer runs in an infinite loop.

Notes:

- To create or edit these files, log in to the Expert mode.
- On VSX Gateway, before you create these files, go to the context of an applicable Virtual System:
  `vsenv <VSID>`
- If these files do not already exist, create them with one of these commands:
  - `touch $FWDIR/conf/<Name of File>`
  - `vi $FWDIR/conf/<Name of File>`
- These files must start with the "`#!/bin/bash`" line.
- These files must end with a new empty line.
- After you edit these files, you must assign the execute permission to them:
  `chmod +x $FWDIR/conf/<Name of File>`

Example of a `$FWDIR/conf/fwaccel_dos_rate_on_install` file:

```
!/bin/bash
fwaccel dos config set --enable-internal
fwaccel dos config set --enable-pbox
```

fwaccel dos deny

Description

The *fwaccel dos deny* and *fwaccel6 dos deny* commands control the IP deny-list in SecureXL.

The deny-list blocks all traffic to and from the specified IP addresses.

The deny-list drops occur in SecureXL, which is more efficient than an Access Control Policy to drop the packets.

Important:

- In VSX mode, you must go to the context of an applicable Virtual System.In Gaia Clish, run: `set virtual-system <VSID>`In the Expert mode, run: `vsenv <VSID>`
- In a Cluster, you must configure all the Cluster Members in the same way.
- To enforce the IP deny-list in SecureXL, you must first enable the IP deny-lists. See these commands:
  - *"fwaccel dos config" on page 56*
  - *"fw sam_policy" on page 184* (configures more granular rules)

Syntax for IPv4

```
fwaccel dos deny
      -a <IPv4 Address>
      -d <IPv4 Address>
      -F
      -M {on | off}
      -m
      -N "<Name of IP Deny-list>"
      -n
      -s
```

Syntax for IPv6

```
fwaccel6 dos deny
      -a <IPv6 Address>
      -d <IPv6 Address>
      -F
      -M {on | off}
      -m
      -N "<Name of IP Deny-list>"
      -n
      -s
```

Parameters

Parameter	Description
No Parameters	Shows the applicable built-in usage.
`-a <IP Address>`	Adds the specified IP address to the deny-list. To add more than one IP address, run this command for each applicable IP address.
`-d <IP Address>`	Removes the specified IP addresses from the deny-list. To remove more than one IP address, run this command for each applicable IP address.
`-F`	Removes (flushes) all IP addresses from the IP deny-list.
`-M {on | off}`	Enables (`on`) or disables (`off`) the monitor-only mode for the IP deny-list. By default, the monitor-only mode is disabled. In the monitor-only mode you can test the IP deny-list without blocking the traffic. This command affects only the IP deny-list (does not affect the `fw samp` rules, etc.).
`-m`	Shows the current status of the monitor-only mode for the IP deny-list (enabled or disabled).
`-N "<Name of IP Deny-list>"`	Configures the name for the IP deny-list. This name appears in the Security Gateway logs. **Notes:** ■ Maximal length is 79 characters. ■ You must only use ASCII characters.
`-n`	Shows the configured name for the IP deny-list.
`-s`	Shows the configured deny-list.

Example from a non-VSX Gateway

```
[Expert@MyGW:0]# fwaccel dos deny -s
The deny list is empty
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos deny -a 1.1.1.1
Adding 1.1.1.1
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos deny -s
1.1.1.1
[Expert@MyGW:0]# fwaccel dos deny -a 2.2.2.2
Adding 2.2.2.2
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos deny -s
2.2.2.2
1.1.1.1
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos deny -d 2.2.2.2
Deleting 2.2.2.2
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos deny -s
1.1.1.1
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos deny -F
All deny list entries deleted
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos deny -s
The deny list is empty
[Expert@MyGW:0]#
```

fwaccel dos pbox

Description

The *fwaccel dos pbox* command controls the Penalty Box allow-list in SecureXL.

The SecureXL Penalty Box is a mechanism that performs an early drop of packets that arrive from suspected sources. The purpose of this feature is to allow the Security Gateway to cope better under high traffic load, possibly caused by a DoS/DDoS attack.

The SecureXL Penalty Box detects clients that send packets, which the Access Control Policy drops, and clients that violate the IPS protections. If the SecureXL Penalty Box detects a specific client frequently, it puts that client in a penalty box. From that point, SecureXL drops all packets that arrive from the blocked source IP address.

The Penalty Box allow-list in SecureXL configures the source IP addresses, which the SecureXL Penalty Box never blocks.

Important:

- This command supports only IPv4.
- In VSX mode, you must go to the context of an applicable Virtual System.In Gaia Clish, run: `set virtual-system <VSID>`In the Expert mode, run: `vsenv <VSID>`
- In a Cluster, you must configure all the Cluster Members in the same way.
- To enforce the Penalty Box in SecureXL, you must first enable the Penalty Box. See these commands:
  - *"fwaccel dos config" on page 56*
  - *"fwaccel dos allow" on page 52*
  - *"fwaccel synatk allow" on page 127*

Syntax for IPv4

```
fwaccel dos pbox
      allow
            -a <IPv4 Address>[/<Subnet Prefix>]
            -d <IPv4 Address>[/<Subnet Prefix>]
            -F
            -l /<Path>/<Name of File>
            -L
            -s
      flush
```

Parameters

Parameter	Description
No Parameters	Shows the applicable built-in usage.
`allow <options>`	Configures the allow-list for source IP addresses in the SecureXL Penalty Box. **Important** - This allow-list overrides which packet the SecureXL Penalty Box drops. Before you use a 3rd-party or automatic blacklists, add trusted networks and hosts to the allow-list to avoid outages. **Note** - This command is similar to the *"fwaccel dos allow" on page 52* command.

`allow -a <IPv4 Address>[/<Subnet Prefix>]`	Adds the specified IP address to the Penalty Box allow-list.
	- `<IPv4 Address>`
	Can be an IP address of a network or a host.
	- `<Subnet Prefix>`
	Must specify the length of the subnet mask in the format `/<bits>`.
	Optional for a host IP address.
	Mandatory for a network IP address.
	Range - from /1 to /32.
	Important - If you do not specify the subnet prefix explicitly, this command uses the subnet prefix /32.
	Examples:
	- For a host:
	`192.168.20.30`
	`192.168.20.30/32`
	- For a network:
	`192.168.20.0/24`

Parameter	Description
`allow -d <IPv4 Address>[/<Subnet Prefix>]`	Removes the specified IP address from the Penalty Box allow-list. ■ `<IPv4 Address>` Can be an IP address of a network or a host. ■ `<Subnet Prefix>` Optional. Must specify the length of the subnet mask in the format `/<bits>`. Optional for a host IP address. Mandatory for a network IP address. Range - from /1 to /32. **Important** - If you do not specify the subnet prefix explicitly, this command uses the subnet prefix /32.
`allow -F`	Removes (flushes) all entries from the Penalty Box allow-list.
`allow -l /<Path>/<Name of File>`	Loads the Penalty Box allow-list entries from the specified plain-text file. **Important:** ■ You must manually create and configure this file with the `touch` or `vi` command. ■ You must assign at least the read permission to this file with the `chmod +x` command. ■ Each entry in this file must be on a separate line. ■ Each entry in this file must be in this format: `<IPv4 Address>[/<Subnet Prefix>]` ■ SecureXL ignores empty lines and lines that start with the # character in this file.

Parameter	Description

`allow -L`	Loads the Penalty Box allow-list entries from the plain-text file with a predefined name:
	`$FWDIR/conf/pbox-allow-list-v4.conf`
	Security Gateway automatically runs this command "`fwaccel dos pbox allow -L`" during each boot.
	Important:
	■ This file does not exist by default.
	■ You must manually create and configure this file with the `touch` or `vi` command.
	■ You must assign at least the read permission to this file with the `chmod +x` command.
	■ Each entry in this file must be on a separate line.
	■ Each entry in this file must be in this format:
	`<`*`IPv4 Address`*`>[/<`*`Subnet Prefix`*`>]`
	■ SecureXL ignores empty lines and lines that start with the # character in this file.

`allow -s`	Shows the current Penalty Box allow-list entries.
`flush`	Removes (flushes) all source IP addresses from the Penalty Box.

Example 1 - Adding a host IP address without optional subnet prefix

```
[Expert@MyGW:0]# fwaccel dos pbox allow -a 192.168.20.40
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos pbox allow -s
192.168.20.40/32
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos pbox allow -F
[Expert@MyGW:0]# fwaccel dos pbox allow -s
[Expert@MyGW:0]#
```

Example 2 - Adding a host IP address with optional subnet prefix

```
[Expert@MyGW:0]# fwaccel dos pbox allow -a 192.168.20.40/32
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos pbox allow -s
192.168.20.40/32
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos pbox allow -F
[Expert@MyGW:0]# fwaccel dos pbox allow -s
[Expert@MyGW:0]#
```

Example 3 - Adding a network IP address with mandatory subnet prefix

```
[Expert@MyGW:0]# fwaccel dos pbox allow -a 192.168.20.0/24
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos pbox allow -s
192.168.20.0/24
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos pbox allow -F
[Expert@MyGW:0]# fwaccel dos pbox allow -s
[Expert@MyGW:0]#
```

Example 4 - Deleting an entry

```
[Expert@MyGW:0]# fwaccel dos pbox allow -a 192.168.20.40/32
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos pbox allow -a 192.168.20.70/32
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos pbox allow -s
192.168.20.40/32
192.168.20.70/32
[Expert@MyGW:0]# fwaccel dos pbox allow -d 192.168.20.70/32
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dos pbox allow -s
192.168.20.40/32
[Expert@MyGW:0]#
```

fwaccel dos rate

Description

The *fwaccel dos rate* and *fwaccel6 dos rate* commands show and install the Rate Limiting policy in SecureXL.

Important:

- In VSX mode, you must go to the context of an applicable Virtual System.In Gaia Clish, run: `set virtual-system <VSID>`In the Expert mode, run: `vsenv <VSID>`
- In a Cluster, you must configure all the Cluster Members in the same way.

Syntax for IPv4

```
fwaccel dos rate
        get '<Rule UID>'
        install
```

Syntax for IPv6

```
fwaccel6 dos rate
        get '<Rule UID>'
        install
```

Parameters

Parameter	Description
No Parameters	Shows the applicable built-in usage.
`get '<Rule UID>'`	Shows information about the rule specified by its Rule UID or its zero-based rule index. The quote marks and angle brackets (`'<...>'`) are mandatory.
`install`	Installs a new rate limiting policy. **Important** - This command requires input from the *stdin*. To use this command, run: `fw sam_policy get -l -k req_type -t in -v quota | fwaccel dos rate install` For more information about the "`fw sam_policy`" command, see *"fw sam_policy" on page 184*.

Notes

- If you install a new rate limiting policy with more than one rule, it automatically enables the rate limiting feature.

  To disable the rate limiting feature manually, run this command (see *"fwaccel dos config" on page 56*):

  ```
  fwaccel dos config set --disable-rate-limit
  ```

- To delete the current rate limiting policy, install a new policy with zero rules.

fwaccel dos stats

Description

The *fwaccel dos stats* and *fwaccel6 dos stats* commands show and clear the DoS real-time statistics in SecureXL.

Important:

- In VSX mode, you must go to the context of an applicable Virtual System.In Gaia Clish, run: `set virtual-system <VSID>`In the Expert mode, run: `vsenv <VSID>`
- In a Cluster, you must configure all the Cluster Members in the same way.

Syntax for IPv4

```
fwaccel stats
      clear
      get
```

Syntax for IPv6

```
fwaccel6 dos stats
      clear
      get
```

Parameters

Parameter	Description
No Parameters	Shows the applicable built-in usage.
`clear`	Clears the real-time statistics counters.
`get`	Shows the real-time statistics counters.

Example - Get the current DoS statistics

```
[Expert@MyGW:0]# fwaccel dos stats get

Firewall Instances in Aggregate:
    Memory Usage:                      0
    Total Active Connections:  (FW connection limiting inactive)
    New Connections/Second:    (FW connection limiting inactive)
    Number of Elements in Tables:
        Penalty Box Violating IPs:                     0
        Rate Limit Source Only Tracks:                 0
        Rate Limit Source and Service Tracks:          0
        Rate Limit Dest Only Tracks:                   0
        Rate Limit Dest and Service Tracks:            0

SecureXL:
    Memory Usage:                      0
    Packets/Second:                    (rate limiting inactive)
    Bytes/Second:                      (rate limiting inactive)
    Reasons Packets Dropped:
        IP Fragment:          0
        IP Option:            0
        Penalty Box:          0
        Deny List:            0
        Rate Limit:           0
    Number of Elements in Tables:
        Penalty Box:                                   0
        Non-Empty Deny Lists:                          0
        Deny List IPs:                                 0
        Rate Limit Matches:                            0
        Rate Limit Source Only Tracks:                 0
        Rate Limit Source and Service Tracks:          0
        Rate Limit Dest Only Tracks:                   0
        Rate Limit Dest and Service Tracks:            0
[Expert@MyGW:0]#
```

fwaccel feature

Description

The *fwaccel feature* and *fwaccel6 feature* commands enable and disable the specified SecureXL features.

Important:

- If you disable a SecureXL feature, SecureXL does not accelerate the applicable traffic anymore.
- This change does **not** survive reboot.
- In VSX Gateway, this change is global and applies to all Virtual Systems.
- In a Cluster, you must configure all the Cluster Members in the same way.

Syntax for IPv4

```
fwaccel feature <Name of Feature>
      get
      off
      on
```

Syntax for IPv6

```
fwaccel6 feature <Name of Feature>
      get
      off
      on
```

Parameters

Parameter	Description
No Parameters	Shows the applicable built-in usage.
<Name of Feature>	Specifies the SecureXL feature. R81 SecureXL supports only this feature: ■ Name: sctp ■ Description: Stream Control Transmission Protocol (SCTP) - see sk35113
get	Shows the current state of the specified SecureXL feature.
off	Disables the specified SecureXL feature. This means that SecureXL does not accelerate the applicable traffic anymore.
on	Enables the specified SecureXL feature. This means that SecureXL accelerates the applicable traffic again.

Disabling the 'sctp' feature permanently

See *"Working with Kernel Parameters on Security Gateway" on page 375*.

1. Add this line to the `$FWDIR/boot/modules/fwkern.conf` file:

   ```
   sim_sctp_disable_by_default=1
   ```

2. Reboot.

Example 1 - Default output

```
[Expert@MyGW:0]# fwaccel feature
Usage: fwaccel feature <name> {on|off|get}

Available features: sctp
[Expert@MyGW:0]#
```

Example 2 - Disabling and enabling a feature

```
[Expert@MyGW:0]# fwaccel feature sctp get
sim_sctp_disable_by_default = 0
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel feature sctp off
Set operation succeeded
[Expert@MyGW:0]# fwaccel feature sctp get
sim_sctp_disable_by_default = 1
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel feature sctp on
Set operation succeeded
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel feature sctp get
sim_sctp_disable_by_default = 0
[Expert@MyGW:0]#
```

fwaccel off

Description

The *fwaccel off* and *fwaccel6 off* commands stop the SecureXL on-the-fly.

Starting from R80.20, you can stop the SecureXL only *temporarily*. The SecureXL starts automatically when you start Check Point services (with the `cpstart` command), or reboot the Security Gateway.

Important:

- Disable the SecureXL only for debug purposes, if Check Point Support explicitly instructs you to do so.
- If you disable the SecureXL, this change does **not** survive reboot.
  SecureXL remains disabled until you enable it again on-the-fly, or reboot the Security Gateway.
- If you disable the SecureXL, this change applies only to new connections that arrive after you disable the acceleration.
  SecureXL continues to accelerate the connections that are already accelerated.
  Other non-connection oriented processing continues to function (for example, virtual defragmentation, VPN decrypt).
- On a VSX Gateway:
  - If you wish to stop the acceleration only for a specific Virtual System, go to the context of that Virtual System.
    In Gaia Clish, run: `set virtual-system <VSID>`
    In Expert mode, run: `vsenv <VSID>`
  - If you wish to stop the acceleration for all Virtual Systems, you must use the "`-a`" parameter.
    In this case, it does not matter from which Virtual System context you run this command.
- In a Cluster, you must configure all the Cluster Members in the same way.

Syntax for IPv4

```
fwaccel off [-a] [-q]
```

Syntax for IPv6

```
fwaccel6 off [-a] [-q]
```

Parameters

Parameter	Description
`-a`	On a VSX Gateway, stops acceleration on all Virtual Systems.
`-q`	Suppresses the output (does not show a returned output).

Possible returned output

- `SecureXL device disabled`
- `SecureXL device is not active`
- `Failed to disable SecureXL device`
- `fwaccel_off: failed to set process context <VSID>`

Example 1 - Output from a non-VSX Gateway

```
[Expert@MyGW:0]# fwaccel off
SecureXL device disabled.
[Expert@MyGW:0]#
```

Example 2 - Output from a VSX Gateway for a specific Virtual System

```
[Expert@MyVSXGW:1]# vsx stat -v
VSX Gateway Status
==================
Name:            VSX2_192.168.3.242
Access Control Policy: VSX_GW_VSX
Installed at:    17Sep2018 13:17:14
Threat Prevention Policy: <No Policy>
SIC Status:      Trust

Number of Virtual Systems allowed by license:          25
Virtual Systems [active / configured]:                  2 / 2
Virtual Routers and Switches [active / configured]:     0 / 0
Total connections [current / limit]:                    4 / 44700

Virtual Devices Status
======================

 ID  | Type &amp; Name     | Access Control Policy | Installed at    | Threat Prevention Policy
| SIC Stat
-----+---------------------+-----------------------+-----------------+--------------------------
+---------
   1 | S VS1               | VS1_Policy            | 17Sep2018 12:47 | <No Policy>
| Trust
   2 | S VS2               | VS2_Policy            | 17Sep2018 12:47 | <No Policy>
| Trust

Type: S - Virtual System, B - Virtual System in Bridge mode,
      R - Virtual Router, W - Virtual Switch.

[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# vsenv 1
Context is set to Virtual Device VS1 (ID 1).
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# fwaccel stat -t
+------------------------------------------------------------------------------+
|Id|Name |Status     |Interfaces               |Features                       |
+------------------------------------------------------------------------------+
|0 |SND  |enabled    |eth1,eth2,eth3           |Acceleration,Cryptography      |
+------------------------------------------------------------------------------+

[Expert@MyVSXGW:1]#

[Expert@MyVSXGW:1]# fwaccel off
SecureXL device disabled. (Virtual ID 1)
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# fwaccel stat -t
+------------------------------------------------------------------------------+
|Id|Name |Status     |Interfaces               |Features                       |
+------------------------------------------------------------------------------+
|0 |SND  |disabled   |eth1,eth2,eth3           |Acceleration,Cryptography      |
+------------------------------------------------------------------------------+

[Expert@MyVSXGW:1]#
```

Example 3 - Output from a VSX Gateway for all Virtual Systems

```
[Expert@MyVSXGW:1]# vsx stat -v
VSX Gateway Status
==================
Name:            VSX2_192.168.3.242
Access Control Policy: VSX_GW_VSX
Installed at:    17Sep2018 13:17:14
Threat Prevention Policy: <No Policy>
SIC Status:      Trust

Number of Virtual Systems allowed by license:          25
Virtual Systems [active / configured]:                  2 / 2
Virtual Routers and Switches [active / configured]:     0 / 0
Total connections [current / limit]:                    4 / 44700

Virtual Devices Status
======================

 ID  | Type &amp; Name     | Access Control Policy | Installed at    | Threat Prevention Policy
| SIC Stat
-----+---------------------+-----------------------+-----------------+--------------------------
+---------
   1 | S VS1               | VS1_Policy            | 17Sep2018 12:47 | <No Policy>
| Trust
   2 | S VS2               | VS2_Policy            | 17Sep2018 12:47 | <No Policy>
| Trust

Type: S - Virtual System, B - Virtual System in Bridge mode,
      R - Virtual Router, W - Virtual Switch.

[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# vsenv 1
Context is set to Virtual Device VS1 (ID 1).
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# fwaccel off -a
SecureXL device disabled. (Virtual ID 0)
SecureXL device disabled. (Virtual ID 1)
SecureXL device disabled. (Virtual ID 2)
[Expert@MyVSXGW:1]#
```

fwaccel on

Description

The *fwaccel on* and *fwaccel6 on* commands start the acceleration on-the-fly, if it was previously stopped with the *fwaccel off* or *fwaccel6 off* command (see *"fwaccel off" on page 77*).

Important:

- On a VSX Gateway:
  - If you wish to start the acceleration only for a specific Virtual System, go to the context of that Virtual System.
    In Gaia Clish, run: `set virtual-system <VSID>`
    In Expert mode, run: `vsenv <VSID>`
  - If you wish to start the acceleration for all Virtual Systems, you must use the "`-a`" parameter.
    In this case, it does not matter from which Virtual System context you run this command.
- In a Cluster, you must configure all the Cluster Members in the same way.

Syntax for IPv4

```
fwaccel on [-a] [-q]
```

Syntax for IPv6

```
fwaccel6 on [-a] [-q]
```

Parameters

Parameter	Description
`-a`	On a VSX Gateway, starts the acceleration on all Virtual Systems.
`-q`	Suppresses the output (does not show a returned output).

Possible returned output

- `SecureXL device is enabled.`
- `Failed to start SecureXL.`
- `No license for SecureXL.`
- `SecureXL is disabled by the firewall. Please try again later.`

- The installed SecureXL device is not compatible with the installed firewall (version mismatch).
- The SecureXL device is in the process of being stopped. Please try again later.
- SecureXL cannot be started while "flows" are active.
- SecureXL is already started.
- SecureXL will be started after a policy is loaded.
- fwaccel: Failed to check FloodGate-1 status. Acceleration will not be started.
- FW-1: SecureXL acceleration cannot be started while QoS is running in express mode.

  Please disable FloodGate-1 express mode or SecureXL.
- FW-1: SecureXL acceleration cannot be started while QoS is running with citrix printing rule.

  Please remove the citrix printing rule to enable SecureXL.
- FW-1: SecureXL acceleration cannot be started while QoS is running with UAS rule.

  Please remove the UAS rule to enable SecureXL.
- FW-1: SecureXL acceleration cannot be started while QoS is running.

  Please remove the QoS blade to enable SecureXL.
- Failed to enable SecureXL device
- fwaccel_on: failed to set process context <*VSID*>

Example 1 - Output from a non-VSX Gateway

```
[Expert@MyGW:0]# fwaccel on
SecureXL device is enabled.
[Expert@MyGW:0]#
```

Example 2 - Output from a VSX Gateway for a specific Virtual System

```
[Expert@MyVSXGW:1]# vsx stat -v
VSX Gateway Status
==================
Name:            VSX2_192.168.3.242
Access Control Policy: VSX_GW_VSX
Installed at:    17Sep2018 13:17:14
Threat Prevention Policy: <No Policy>
SIC Status:      Trust

Number of Virtual Systems allowed by license:          25
Virtual Systems [active / configured]:                  2 / 2
Virtual Routers and Switches [active / configured]:     0 / 0
Total connections [current / limit]:                    4 / 44700

Virtual Devices Status
======================

 ID  | Type &amp; Name     | Access Control Policy | Installed at    | Threat Prevention Policy
| SIC Stat
-----+---------------------+-----------------------+-----------------+--------------------------
+---------
   1 | S VS1               | VS1_Policy            | 17Sep2018 12:47 | <No Policy>
| Trust
   2 | S VS2               | VS2_Policy            | 17Sep2018 12:47 | <No Policy>
| Trust

Type: S - Virtual System, B - Virtual System in Bridge mode,
      R - Virtual Router, W - Virtual Switch.

[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# vsenv 1
Context is set to Virtual Device VS1 (ID 1).
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# fwaccel stat -t
+-----------------------------------------------------------------------------+
|Id|Name |Status     |Interfaces               |Features                      |
+-----------------------------------------------------------------------------+
|0 |SND  |disabled   |eth1,eth2,eth3           |Acceleration,Cryptography     |
+-----------------------------------------------------------------------------+

[Expert@MyVSXGW:1]#

[Expert@MyVSXGW:1]# fwaccel on
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# fwaccel stat -t
+-----------------------------------------------------------------------------+
|Id|Name |Status     |Interfaces               |Features                      |
+-----------------------------------------------------------------------------+
|0 |SND  |enabled    |eth1,eth2,eth3           |Acceleration,Cryptography     |
+-----------------------------------------------------------------------------+

[Expert@MyVSXGW:1]#
```

Example 3 - Output from a VSX Gateway for all Virtual Systems

```
[Expert@MyVSXGW:1]# vsx stat -v
VSX Gateway Status
==================
Name:            VSX2_192.168.3.242
Access Control Policy: VSX_GW_VSX
Installed at:    17Sep2018 13:17:14
Threat Prevention Policy: <No Policy>
SIC Status:      Trust

Number of Virtual Systems allowed by license:          25
Virtual Systems [active / configured]:                  2 / 2
Virtual Routers and Switches [active / configured]:     0 / 0
Total connections [current / limit]:                    4 / 44700

Virtual Devices Status
======================

 ID  | Type &amp; Name     | Access Control Policy | Installed at    | Threat Prevention Policy
| SIC Stat
-----+---------------------+-----------------------+-----------------+-------------------------
+---------
   1 | S VS1               | VS1_Policy            | 17Sep2018 12:47 | <No Policy>
| Trust
   2 | S VS2               | VS2_Policy            | 17Sep2018 12:47 | <No Policy>
| Trust

Type: S - Virtual System, B - Virtual System in Bridge mode,
      R - Virtual Router, W - Virtual Switch.

[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# vsenv 1
Context is set to Virtual Device VS1 (ID 1).
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# fwaccel on -a
[Expert@MyVSXGW:1]#
```

fwaccel ranges

Description

The *fwaccel ranges* and *fwaccel6 ranges* commands show the SecureXL loaded ranges:

- Ranges of Rule Base source IP addresses
- Ranges of Rule Base destination IP addresses
- Ranges of Rule Base destination ports and protocols

The Security Gateway creates these ranges during the policy installation. The Firewall creates and offloads ranges to SecureXL when any of these feature is enabled:

- Rulebase ranges for Drop Templates
- Anti-Spoofing enforcement ranges on per-interface basis
- NAT64 ranges
- NAT46 ranges

These ranges are related to matching of connections to SecureXL Drop Templates. These ranges represent the **Source**, **Destination** and **Service** columns of the Rule Base.

These ranges are not exactly the same as the Rule Base, because as there are objects that cannot be represented as real (deterministic) IP addresses. For example, Domain objects and Dynamic objects. The Security Gateway converts such non-deterministic objects to "Any" IP address.

In addition, implied rules are represented in these ranges, except for some specific implied rules.

You can use these commands for troubleshooting.

Syntax for IPv4

```
fwaccel ranges
      -h
      -a
      -l
      -p <Range ID>
      -s <Range ID>
```

Syntax for IPv6

```
fwaccel6 ranges
        -h
        -a
        -l
        -p <Range ID>
        -s <Range ID>
```

Parameters

Parameter	Description
`-h`	Shows the applicable built-in usage.
`-a` *or* No Parameters	Shows the full information for all loaded ranges. **Note** - In the list of SecureXL Drop Templates (output of the *"fwaccel templates" on page 145* command), each Drop Template is assembled from ranges indexes. To see mapping between range index and the range itself, run this command "`fwaccel ranges -a`". This way you understand better the practical ranges for Drop Templates and when it is appropriate to use them.
`-l`	Shows the list of loaded ranges: ■ 0 - Ranges of Rule Base source IP addresses ■ 1 - Ranges of Rule Base destination IP addresses ■ 2 - Ranges of Rule Base destination ports and protocols
`-p` *`<Range ID>`*	Shows the full information for the specified range.
`-s` *`<Range ID>`*	Shows the summary information for the specified range.

Examples

Example 1 - Show the list of ranges from a non-VSX Gateway

```
[Expert@MyGW:0]# fwaccel ranges -l
SecureXL device 0:
        0 Rule base source ranges (ip):
        1 Rule base destination ranges (ip):
        2 Rule base dport ranges (port, proto):
[Expert@MyGW:0]#
```

Example 2 - Show the full information for all loaded ranges from a non-VSX Gateway

```
[Expert@MyGW:0]# fwaccel ranges
SecureXL device 0:
    Rule base source ranges (ip):
        (0) 0.0.0.0 - 192.168.204.0
        (1) 192.168.204.1 - 192.168.204.1
        (2) 192.168.204.2 - 192.168.204.39
        (3) 192.168.204.40 - 192.168.204.40
        (4) 192.168.204.41 - 192.168.254.39
        (5) 192.168.254.40 - 192.168.254.40
        (6) 192.168.254.41 - 255.255.255.255
    Rule base destination ranges (ip):
        (0) 0.0.0.0 - 192.168.204.0
        (1) 192.168.204.1 - 192.168.204.1
        (2) 192.168.204.2 - 192.168.204.39
        (3) 192.168.204.40 - 192.168.204.40
        (4) 192.168.204.41 - 192.168.254.39
        (5) 192.168.254.40 - 192.168.254.40
        (6) 192.168.254.41 - 255.255.255.255
    Rule base dport ranges (port, proto):
        (0) 0, 0 - 138, 6
        (1) 139, 6 - 139, 6
        (2) 140, 6 - 18189, 6
        (3) 18190, 6 - 18190, 6
        (4) 18191, 6 - 18191, 6
        (5) 18192, 6 - 18192, 6
        (6) 18193, 6 - 19008, 6
        (7) 19009, 6 - 19009, 6
        (8) 19010, 6 - 136, 17
        (9) 137, 17 - 138, 17
        (10) 139, 17 - 65535, 65535
[Expert@MyGW:0]#
```

Example 3 - Show the full information for the specified range from a non-VSX Gateway

```
[Expert@MyGW:0]# fwaccel ranges -p 0
SecureXL device 0:
    Rule base source ranges (ip):
        (0) 0.0.0.0 - 192.168.204.0
        (1) 192.168.204.1 - 192.168.204.1
        (2) 192.168.204.2 - 192.168.204.39
        (3) 192.168.204.40 - 192.168.204.40
        (4) 192.168.204.41 - 192.168.254.39
        (5) 192.168.254.40 - 192.168.254.40
        (6) 192.168.254.41 - 255.255.255.255
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel ranges -p 1
SecureXL device 0:
    Rule base destination ranges (ip):
        (0) 0.0.0.0 - 192.168.204.0
        (1) 192.168.204.1 - 192.168.204.1
        (2) 192.168.204.2 - 192.168.204.39
        (3) 192.168.204.40 - 192.168.204.40
        (4) 192.168.204.41 - 192.168.254.39
        (5) 192.168.254.40 - 192.168.254.40
        (6) 192.168.254.41 - 255.255.255.255
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel ranges -p 2
SecureXL device 0:
    Rule base dport ranges (port, proto):
        (0) 0, 0 - 138, 6
        (1) 139, 6 - 139, 6
        (2) 140, 6 - 18189, 6
        (3) 18190, 6 - 18190, 6
        (4) 18191, 6 - 18191, 6
        (5) 18192, 6 - 18192, 6
        (6) 18193, 6 - 19008, 6
        (7) 19009, 6 - 19009, 6
        (8) 19010, 6 - 136, 17
        (9) 137, 17 - 138, 17
        (10) 139, 17 - 65535, 65535
[Expert@MyGW:0]#
```

Example 4 - Show the summary information for the specified range from a non-VSX Gateway

```
[Expert@MyGW:0]# fwaccel ranges -s 0
SecureXL device 0:
    List name "Rule base source ranges (ip):", ID 0, Number of ranges 7
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel ranges -s 1
SecureXL device 0:
    List name "Rule base destination ranges (ip):", ID 1, Number of ranges 7
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel ranges -s 2
SecureXL device 0:
    List name "Rule base dport ranges (port, proto):", ID 2, Number of ranges 11
[Expert@MyGW:0]#
```

Example 5 - Show the list of ranges from a VSX Gateway

```
[Expert@MyVSXGW:2]# vsenv 0
Context is set to Virtual Device VSX2_192.168.3.242 (ID 0).
[Expert@MyVSXGW:0]# fwaccel ranges -l
SecureXL device 0:
        0 Anti spoofing ranges eth0:
        1 Anti spoofing ranges eth1:
[Expert@MyVSXGW:0]# vsenv 1
Context is set to Virtual Device VS1 (ID 1).
[Expert@MyVSXGW:1]# fwaccel ranges -l
SecureXL device 0:
        0 Anti spoofing ranges eth3:
        1 Anti spoofing ranges eth2.52:
[Expert@MyVSXGW:1]# vsenv 2
Context is set to Virtual Device VS2 (ID 2).
[Expert@MyVSXGW:2]# fwaccel ranges -l
SecureXL device 0:
        0 Anti spoofing ranges eth4:
        1 Anti spoofing ranges eth2.53:
[Expert@MyVSXGW:2]#
```

Example 6 - Show the full information for all loaded ranges from a VSX Gateway

```
[Expert@MyVSXGW:2]# vsenv 0
Context is set to Virtual Device VSX2_192.168.3.242 (ID 0).
[Expert@MyVSXGW:0]# fwaccel ranges
SecureXL device 0:
    Anti spoofing ranges eth0:
        (0) 0.0.0.0 - 10.20.29.255
        (1) 10.20.31.0 - 126.255.255.255
        (2) 128.0.0.0 - 192.168.2.255
        (3) 192.168.3.1 - 192.168.3.241
        (4) 192.168.3.243 - 192.168.3.254
        (5) 192.168.4.0 - 223.255.255.255
        (6) 240.0.0.0 - 255.255.255.254
    Anti spoofing ranges eth1:
        (0) 10.20.30.1 - 10.20.30.241
        (1) 10.20.30.243 - 10.20.30.254
[Expert@MyVSXGW:0]#
[Expert@MyVSXGW:1]# vsenv 1
Context is set to Virtual Device VS1 (ID 1).
[Expert@MyVSXGW:1]# fwaccel ranges
SecureXL device 0:
    Anti spoofing ranges eth3:
        (0) 40.50.60.0 - 40.50.60.255
        (1) 192.168.196.17 - 192.168.196.17
        (2) 192.168.196.19 - 192.168.196.30
    Anti spoofing ranges eth2.52:
        (0) 70.80.90.0 - 70.80.90.255
        (1) 192.168.196.1 - 192.168.196.1
        (2) 192.168.196.3 - 192.168.196.14
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# vsenv 2
Context is set to Virtual Device VS2 (ID 2).
[Expert@MyVSXGW:2]# fwaccel ranges
SecureXL device 0:
    Anti spoofing ranges eth4:
        (0) 100.100.100.0 - 100.100.100.255
        (1) 192.168.196.17 - 192.168.196.17
        (2) 192.168.196.19 - 192.168.196.30
    Anti spoofing ranges eth2.53:
        (0) 192.168.196.1 - 192.168.196.1
        (1) 192.168.196.3 - 192.168.196.14
        (2) 200.200.200.0 - 200.200.200.255
[Expert@MyVSXGW:2]#
```

Example 7 - Show the summary information for the specified range from a VSX Gateway

```
[Expert@MyVSXGW:2]# vsenv 1
Context is set to Virtual Device VS1 (ID 1).
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# fwaccel ranges -s 0
SecureXL device 0:
    List name "Anti spoofing ranges eth3:", ID 0, Number of ranges 3
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# fwaccel ranges -s 1
SecureXL device 0:
    List name "Anti spoofing ranges eth2.52:", ID 1, Number of ranges 3
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# fwaccel ranges -s 2
SecureXL device 0:
        The requested range table is empty
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# vsenv 2
Context is set to Virtual Device VS2 (ID 2).
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:2]# fwaccel ranges -s 0
SecureXL device 0:
    List name "Anti spoofing ranges eth4:", ID 0, Number of ranges 3
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:2]# fwaccel ranges -s 1
SecureXL device 0:
    List name "Anti spoofing ranges eth2.53:", ID 1, Number of ranges 3
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:2]# fwaccel ranges -s 2
SecureXL device 0:
        The requested range table is empty
[Expert@MyVSXGW:2]#
```

fwaccel stat

Description

The *fwaccel stat* and *fwaccel6 stat* commands show the SecureXL status, the list of the accelerated interfaces and the list of the accelerated features on the local Security Gateway, or Cluster Member.

Syntax for IPv4

```
fwaccel stat [-a] [-t] [-v]
```

Syntax for IPv6

```
fwaccel6 stat [-a] [-t] [-v]
```

Parameters

Parameter	Description
No Parameters	Shows this information: ■ SecureXL instance ID ■ SecureXL instance role ■ SecureXL status ■ Accelerated interfaces ■ Accelerated features In addition, also shows: ■ More information about the Cryptography feature ■ The status of Accept Templates ■ The status of Drop Templates ■ The status of NAT Templates
`-a`	On a VSX Gateway, shows the information for all Virtual Systems.
`-t`	Shows this information only: ■ SecureXL instance ID ■ SecureXL instance role ■ SecureXL status ■ Accelerated interfaces ■ Accelerated features
`-v`	On a VSX Gateway, shows the information for all Virtual Systems. The same as the "`-a`" parameter.

Example 1 - Full output from a non-VSX Gateway

```
[Expert@MyGW:0]# fwaccel stat
+------------------------------------------------------------------------------+
|Id|Name |Status     |Interfaces               |Features                       |
+------------------------------------------------------------------------------+
|0 |SND  |enabled    |eth0,eth1,eth2,eth3,eth4,|
|  |     |           |eth5,eth6                |Acceleration,Cryptography      |
|  |     |           |                         |Crypto: Tunnel,UDPEncap,MD5,   |
|  |     |           |                         |SHA1,NULL,3DES,DES,CAST,       |
|  |     |           |                         |CAST-40,AES-128,AES-256,ESP,   |
|  |     |           |                         |LinkSelection,DynamicVPN,      |
|  |     |           |                         |NatTraversal,AES-XCBC,SHA256   |
+------------------------------------------------------------------------------+

Accept Templates : disabled by Firewall
                   Layer MyGW_Policy Network disables template offloads from rule #1
                   Throughput acceleration still enabled.
Drop Templates   : disabled
NAT Templates    : disabled by Firewall
                   Layer MyGW_Policy Network disables template offloads from rule #1
                   Throughput acceleration still enabled.
[Expert@MyGW:0]#
```

Example 2 - Brief output from a non-VSX Gateway

```
[Expert@MyGW:0]# fwaccel stat -t
+------------------------------------------------------------------------------+
|Id|Name |Status     |Interfaces               |Features                       |
+------------------------------------------------------------------------------+
|0 |SND  |enabled    |eth0,eth1,eth2,eth3,eth4,|
|  |     |           |eth5,eth6,eth7           |Acceleration,Cryptography      |
+------------------------------------------------------------------------------+

[Expert@MyGW:0]#
```

Example 3 - Full output from a VSX Gateway

```
[Expert@MyVSXGW:1]# vsx stat -v
VSX Gateway Status
==================
Name:            VSX2_192.168.3.242
Access Control Policy: VSX_GW_VSX
Installed at:    17Sep2018 13:17:14
Threat Prevention Policy: <No Policy>
SIC Status:      Trust

Number of Virtual Systems allowed by license:          25
Virtual Systems [active / configured]:                  2 / 2
Virtual Routers and Switches [active / configured]:     0 / 0
Total connections [current / limit]:                    4 / 44700

Virtual Devices Status
======================

 ID  | Type & Name         | Access Control Policy | Installed at
  | Threat Prevention Policy | SIC Stat
-----+---------------------+-----------------------+--------------
---+------------------------+---------
   1 | S VS1               | VS1_Policy            | 17Sep2018
12:47 | <No Policy>             | Trust
   2 | S VS2               | VS2_Policy            | 17Sep2018
12:47 | <No Policy>             | Trust

Type: S - Virtual System, B - Virtual System in Bridge mode,
      R - Virtual Router, W - Virtual Switch.

[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# vsenv 1
Context is set to Virtual Device VS1 (ID 1).
[Expert@MyVSXGW:1]#
[Expert@MyVSXGW:1]# fwaccel stat
+-----------------------------------------------------------------
------------+
|Id|Name |Status     |Interfaces               |Features
           |
+-----------------------------------------------------------------
------------+
|0 |SND  |enabled    |eth1,eth2,eth3
|Acceleration,Cryptography     |
|  |     |           |                         |Crypto:
Tunnel,UDPEncap,MD5,  |
|  |     |           |
|SHA1,NULL,3DES,DES,CAST,       |
|  |     |           |                         |CAST-40,AES-
128,AES-256,ESP,  |
|  |     |           |
|LinkSelection,DynamicVPN,     |
|  |     |           |                         |NatTraversal,AES-
XCBC,SHA256  |
+-----------------------------------------------------------------
------------+

Accept Templates : disabled by Firewall
                   Layer VS1_Policy Network disables template
offloads from rule #1
                   Throughput acceleration still enabled.
Drop Templates   : disabled
NAT Templates    : disabled by Firewall
                   Layer VS1_Policy Network disables template
offloads from rule #1
                   Throughput acceleration still enabled.
[Expert@MyVSXGW:1]#
```

fwaccel stats

Description

The *fwaccel stats* and *fwaccel6 stats* commands show acceleration statistics for IPv4 on the local Security Gateway, or Cluster Member.

Syntax for IPv4

```
fwaccel stats
      [-c]
      [-d]
      [-l]
      [-m]
      [-n]
      [-o]
      [-p]
      [-q]
      [-r]
      [-s]
      [-x]
```

Syntax for IPv6

```
fwaccel6 stats
      [-c]
      [-d]
      [-l]
      [-m]
      [-n]
      [-o]
      [-p]
      [-q]
      [-r]
      [-s]
      [-x]
```

Parameters

Parameter	Description
`-c`	Shows the statistics for Cluster Correction.
`-d`	Shows the statistics for drops from device.
`-l`	Shows the statistics in legacy mode - as one table.
`-m`	Shows the statistics for multicast traffic.
`-n`	Shows the statistics for Identity Awareness (NAC).
`-o`	Shows the statistics for Reorder Infrastructure.
`-p`	Shows the statistics for SecureXL violations (F2F packets).
`-q`	Shows the statistics notifications the SecureXL sent to the Firewall.
`-r`	Resets all the counters.
`-s`	Shows the statistics summary only.
`-x`	Shows the statistics for PXL. **Note** - PXL is the technology name for combination of SecureXL and PSL (Passive Streaming Library).

In addition, see:

- *"Description of the Statistics Counters in the "fwaccel stats" Output" on page 100*
- *"Example Outputs of the "fwaccel stats" Commands" on page 109*

Description of the Statistics Counters in the "fwaccel stats" Output

The "Accelerated Path" section

Counter	Description
`accel packets`	Number of accelerated packets.
`accel bytes`	Number of accelerated bytes.
`outbound packets`	Number of outbound packets.
`outbound bytes`	Number of outbound bytes.
`conns created`	Number of connections the SecureXL created.
`conns deleted`	Number of connections the SecureXL deleted.
`C total conns`	Total number of connections the SecureXL currently handles.
`C templates`	*Not in use* Total number of SecureXL templates the SecureXL currently handles.
`C TCP conns`	Number of TCP connections the SecureXL currently handles.
`C non TCP conns`	Number of non-TCP connections the SecureXL currently handles.
`conns from templates`	*Not in use* Number of connections the SecureXL created from SecureXL templates.
`nat conns`	Number of NAT connections.
`dropped packets`	Number of packets the SecureXL dropped.
`dropped bytes`	Number of bytes the SecureXL dropped.
`nat templates`	*Not in use*
`port alloc templates`	*Not in use*
`conns from nat tmpl`	*Not in use*
`port alloc conns`	*Not in use*

Counter	Description
`fragments received`	Number of received fragments.
`fragments transmit`	Number of transmitted fragments.
`fragments dropped`	Number of dropped fragments.
`fragments expired`	Number of expired fragments.
`IP options stripped`	Number of packets, from SecureXL stripped IP options.
`IP options restored`	Number of packets, in which SecureXL restored IP options.
`IP options dropped`	Number of packets with IP options that SecureXL dropped.
`corrs created`	Number of corrections the SecureXL made.
`corrs deleted`	Number of corrections the SecureXL deleted.
`C corrections`	Number of corrections the SecureXL currently handles.
`corrected packets`	Number of corrected packets.
`corrected bytes`	Number of corrected bytes.

The "Accelerated VPN Path" section

Counter	Description
`C crypt conns`	Number of encrypted connections the SecureXL currently handles.
`enc bytes`	Number of encrypted traffic bytes.
`dec bytes`	Number of decrypted traffic bytes.
`ESP enc pkts`	Number of ESP encrypted packets.
`ESP enc err`	Number of ESP encryption errors.
`ESP dec pkts`	Number of ESP decrypted packets.
`ESP dec err`	Number of ESP decryption errors.
`ESP other err`	Number of ESP general errors.
`espudp enc pkts`	*Not in use*
`espudp enc err`	*Not in use*
`espudp dec pkts`	*Not in use*
`espudp dec err`	*Not in use*
espudp other err	*Not in use*

The "Medium Streaming Path" section

Counter	Description
`PXL packets`	Number of PXL packets. PXL is combination of SecureXL and Passive Streaming Library (PSL), which is an IPS infrastructure that transparently listens to TCP traffic as network packets, and rebuilds the TCP stream out of these packets. Passive Streaming can listen to all TCP traffic, but process only the data packets, which belong to a previously registered connection.
`PXL async packets`	Number of PXL packets the SecureXL handled asynchronously.
`PXL bytes`	Number of PXL bytes.
`C PXL conns`	Number of PXL connections the SecureXL currently handles.
`C PXL templates`	*Not in use* Number of PXL templates.
`PXL FF conns`	Number of PXL Fast Forward connections.
`PXL FF packets`	Number of PXL Fast Forward packets.
`PXL FF bytes`	Number of PXL Fast Forward bytes.
`PXL FF acks`	Number of PXL Fast Forward acknowledgments.

The "Inline Streaming Path" section

Counter	Description
`PSL Inline packets`	Number of accelerated PSL packets.
`PSL Inline bytes`	Number of accelerated PSL bytes.
`CPAS Inline packets`	Number of accelerated CPAS packets.
`CPAS Inline bytes`	Number of accelerated CPAS bytes.

The "QoS General Information" section

Counter	Description
`Total QoS Conns`	Total number of QoS connections.
`QoS Classify Conns`	Number of classified QoS connections.
`QoS Classify flow`	Number of classified QoS flows.
`Reclassify QoS polic`	Number of reclassify QoS requests.

The "Firewall QoS Path" section

Counter	Description
`Enqueued IN packets`	Number of waiting packets in Firewall QoS inbound queue.
`Enqueued OUT packets`	Number of waiting packets in Firewall QoS outbound queue.
`Dequeued IN packets`	Number of processed packets in Firewall QoS inbound queue.
`Dequeued OUT packets`	Number of processed packets in Firewall QoS outbound queue.
`Enqueued IN bytes`	Number of waiting bytes in Firewall QoS inbound queue.
`Enqueued OUT bytes`	Number of waiting bytes in Firewall QoS outbound queue.
`Dequeued IN bytes`	Number of processed bytes in Firewall QoS inbound queue.
`Dequeued OUT bytes`	Number of processed bytes in Firewall QoS outbound queue.

The "Firewall QoS Path" section

Counter	Description
`Enqueued IN packets`	Number of waiting packets in SecureXL QoS inbound queue.
`Enqueued OUT packets`	Number of waiting packets in SecureXL QoS outbound queue.
`Dequeued IN packets`	Number of processed packets in SecureXL QoS inbound queue.
`Dequeued OUT packets`	Number of processed packets in SecureXL QoS outbound queue.
`Enqueued IN bytes`	Number of waiting bytes in SecureXL QoS inbound queue.
`Enqueued OUT bytes`	Number of waiting bytes in SecureXL QoS outbound queue.
`Dequeued IN bytes`	Number of processed bytes in SecureXL QoS inbound queue.
`Dequeued OUT bytes`	Number of processed bytes in SecureXL QoS outbound queue.

The "Firewall Path" section

Counter	Description
`F2F packets`	Number of packets that SecureXL forwarded to the Firewall kernel in Slow Path.
`F2F bytes`	Number of bytes that SecureXL forwarded to the Firewall kernel in Slow Path.
`TCP violations`	Number of packets, which are in violation of the TCP state.
`C anticipated conns`	Number of anticipated connections SecureXL currently handles.
`port alloc f2f`	*Not in use*
`F2V conn match pkts`	Number of packets that matched a SecureXL connection and SecureXL forwarded to the Firewall kernel.
`F2V packets`	Number of packets that SecureXL forwarded to the Firewall kernel and the Firewall re-injected back to SecureXL.
`F2V bytes`	Number of bytes that SecureXL forwarded to the Firewall kernel and the Firewall re-injected back to the SecureXL.

The "GTP" section

Counter	Description
`gtp tunnels created`	Number of created GTP tunnels.
`gtp tunnels`	Number of GTP tunnels the SecureXL currently handles.
`gtp accel pkts`	Number of accelerated GTP packets.
`gtp f2f pkts`	Number of GTP packets the SecureXL forwarded to the Firewall kernel.
`gtp spoofed pkts`	Number of spoofed GTP packets.
`gtp in gtp pkts`	Number of GTP-in-GTP packets.
`gtp signaling pkts`	Number of signaling GTP packets.
`gtp tcpopt pkts`	Number of GTP packets with TCP Options.
`gtp apn err pkts`	Number of GTP packets with APN errors.

The "General" section

Counter	Description
`memory used`	*Not in use*
`free memory`	*Not in use*
`C used templates`	*Not in use*
`pxl tmpl conns`	*Not in use*
`C conns from tmpl`	*Not in use* Number of current connections that SecureXL created from SecureXL Templates.
`C tcp handshake conn`	Number of current TCP connections that are not yet established.
`C tcp established co`	Number of established TCP connections the SecureXL currently handles.
`C tcp closed conns`	Number of closed TCP connections the SecureXL currently handles.
`C tcp pxl handshake`	Number of not yet established PXL TCP connections the SecureXL currently handles.
`C tcp pxl establishe`	Number of established PXL TCP connections the SecureXL currently handles.
`C tcp pxl closed con`	Number of closed PXL TCP connections the SecureXL currently handles.
`outbound pxl packets`	*Not in use*

Example Outputs of the "fwaccel stats" Commands

Example: fwaccel stats -s

Example of statistics summary:

```
Accelerated conns/Total conns : 0/0 (0%)
Accelerated pkts/Total pkts   : 0/8 (0%)
F2Fed pkts/Total pkts         : 8/8 (100%)
F2V pkts/Total pkts           : 0/8 (0%)
CPASXL pkts/Total pkts        : 0/8 (0%)
PSLXL pkts/Total pkts         : 0/8 (0%)
QOS inbound pkts/Total pkts   : 0/8 (0%)
QOS outbound pkts/Total pkts  : 0/8 (0%)
Corrected pkts/Total pkts     : 0/8 (0%)
```

Example: fwaccel stats

Example of the default output:

```
Name                          Value         Name                          Value
----------------------------  ------------  ----------------------------  ------------

Accelerated Path
--------------------------------------------------------------------------------------
accel packets                            0    accel bytes                              0
outbound packets                         0    outbound bytes                           0
conns created                            0    conns deleted                            0
C total conns                            0    C TCP conns                              0
C non TCP conns                          0    nat conns                                0
dropped packets                          0    dropped bytes                            0
fragments received                       0    fragments transmit                       0
fragments dropped                        0    fragments expired                        0
IP options stripped                      0    IP options restored                      0
IP options dropped                       0    corrs created                            0
corrs deleted                            0    C corrections                            0
corrected packets                        0    corrected bytes                          0

Accelerated VPN Path
--------------------------------------------------------------------------------------
C crypt conns                            0    enc bytes                                0
dec bytes                                0    ESP enc pkts                             0
ESP enc err                              0    ESP dec pkts                             0
ESP dec err                              0    ESP other err                            0
espudp enc pkts                          0    espudp enc err                           0
espudp dec pkts                          0    espudp dec err                           0
espudp other err                         0

Medium Streaming Path
--------------------------------------------------------------------------------------
CPASXL packets                           0    PSLXL packets                            0
CPASXL async packets                     0    PSLXL async packets                      0
CPASXL bytes                             0    PSLXL bytes                              0
C CPASXL conns                           0    C PSLXL conns                            0
CPASXL conns created                     0    PSLXL conns created                      0
PXL FF conns                             0    PXL FF packets                           0
PXL FF bytes                             0    PXL FF acks                              0
PXL no conn drops                        0

Inline Streaming Path
--------------------------------------------------------------------------------------
PSL Inline packets                       0    PSL Inline bytes                         0
CPAS Inline packets                      0    CPAS Inline bytes                        0

QoS Paths
--------------------------------------------------------------------------------------
QoS General Information:
------------------------
Total QoS Conns                          0    QoS Classify Conns                       0
QoS Classify flow                        0    Reclassify QoS policy                    0

FireWall QoS Path:
------------------
Enqueued IN packets                      0    Enqueued OUT packets                     0
Dequeued IN packets                      0    Dequeued OUT packets                     0
Enqueued IN bytes                        0    Enqueued OUT bytes                       0
Dequeued IN bytes                        0    Dequeued OUT bytes                       0

Accelerated QoS Path:
---------------------
Enqueued IN packets                      0    Enqueued OUT packets                     0
Dequeued IN packets                      0    Dequeued OUT packets                     0
Enqueued IN bytes                        0    Enqueued OUT bytes                       0
Dequeued IN bytes                        0    Dequeued OUT bytes                       0

Firewall Path
--------------------------------------------------------------------------------------
F2F packets                          35324    F2F bytes                          1797781
TCP violations                           0    F2V conn match pkts                      0
F2V packets                              0    F2V bytes                                0
```

```
GTP
--------------------------------------------------------------------------------------
gtp tunnels created                       0    gtp tunnels                              0
gtp accel pkts                            0    gtp f2f pkts                             0
gtp spoofed pkts                          0    gtp in gtp pkts                          0
gtp signaling pkts                        0    gtp tcpopt pkts                          0
gtp apn err pkts                          0

General
--------------------------------------------------------------------------------------
memory used                        38798784    C tcp handshake conns                    0
C tcp established conns                   0    C tcp closed conns                       0
C tcp pxl handshake conns                 0    C tcp pxl established conns              0
C tcp pxl closed conns                    0    outbound cpasxl packets                  0
outbound pslxl packets                    0    outbound cpasxl bytes                    0
outbound pslxl bytes                      0    DNS DoR stats                            0

(*) Statistics marked with C refer to current value, others refer to total value
```

Example: fwaccel stats -c

Example of statistics for Cluster Correction:

```
Cluster Correction stats:

 Name                     Value           Name                     Value
-----------------------  ------------    -----------------------  ------------
Sent pkts (total)                   0    Sent with metadata                   0
Received pkts (total)               0    Received with metadata               0
Sent bytes                          0    Received bytes                       0
Send errors                         0    Receive errors                       0
```

Example: fwaccel stats -d

Example of statistics for drops from device:

```
Reason                Value              Reason                Value
--------------------  ---------------    --------------------  ---------------
general reason                      0    CPASXL decision                      0
PSLXL decision                      0    clr pkt on vpn                       0
encrypt failed                      0    drop template                        0
decrypt failed                      0    interface down                       0
cluster error                       0    XMT error                            0
anti spoofing                       0    local spoofing                       0
sanity error                        0    monitored spoofed                    0
QOS decision                        0    C2S violation                        0
S2C violation                       0    Loop prevention                      0
DOS Fragments                       0    DOS IP Options                       0
DOS Blacklists                      0    DOS Penalty Box                      0
DOS Rate Limiting                   0    Syn Attack                           0
Reorder                             0    Expired Fragments                    0
```

Example: fwaccel stats -l

Example of the output in legacy mode (as one table):

```
Name                          Value         Name                          Value
----------------------------  ------------  ----------------------------  ------------
-                                        0   accel packets                           0
accel bytes                              0   outbound packets                        0
outbound bytes                           0   conns created                           0
conns deleted                            0   C total conns                           0
C TCP conns                              0   C non TCP conns                         0
nat conns                                0   dropped packets                         0
dropped bytes                            0   fragments received                      0
fragments transmit                       0   fragments dropped                       0
fragments expired                        0   IP options stripped                     0
IP options restored                      0   IP options dropped                      0
corrs created                            0   corrs deleted                           0
C corrections                            0   corrected packets                       0
corrected bytes                          0   C crypt conns                           0
enc bytes                                0   dec bytes                               0
ESP enc pkts                             0   ESP enc err                             0
ESP dec pkts                             0   ESP dec err                             0
ESP other err                            0   espudp enc pkts                         0
espudp enc err                           0   espudp dec pkts                         0
espudp dec err                           0   espudp other err                        0
acct update interval                  3600   CPASXL packets                          0
PSLXL packets                            0   CPASXL async packets                    0
PSLXL async packets                      0   CPASXL bytes                            0
PSLXL bytes                              0   C CPASXL conns                          0
C PSLXL conns                            0   CPASXL conns created                    0
PSLXL conns created                      0   PXL FF conns                            0
PXL FF packets                           0   PXL FF bytes                            0
PXL FF acks                              0   PXL no conn drops                       0
PSL Inline packets                       0   PSL Inline bytes                        0
CPAS Inline packets                      0   CPAS Inline bytes                       0
Total QoS Conns                          0   QoS Classify Conns                      0
QoS Classify flow                        0   Reclassify QoS policy                   0
Enqueued IN packets                      0   Enqueued OUT packets                    0
Dequeued IN packets                      0   Dequeued OUT packets                    0
Enqueued IN bytes                        0   Enqueued OUT bytes                      0
Dequeued IN bytes                        0   Dequeued OUT bytes                      0
Enqueued IN packets                      0   Enqueued OUT packets                    0
Dequeued IN packets                      0   Dequeued OUT packets                    0
Enqueued IN bytes                        0   Enqueued OUT bytes                      0
Dequeued IN bytes                        0   Dequeued OUT bytes                      0
F2F packets                          35383   F2F bytes                         1801493
TCP violations                           0   F2V conn match pkts                     0
F2V packets                              0   F2V bytes                               0
gtp tunnels created                      0   gtp tunnels                             0
gtp accel pkts                           0   gtp f2f pkts                            0
gtp spoofed pkts                         0   gtp in gtp pkts                         0
gtp signaling pkts                       0   gtp tcpopt pkts                         0
gtp apn err pkts                         0   memory used                      38798784
C tcp handshake conns                    0   C tcp established conns                 0
C tcp closed conns                       0   C tcp pxl handshake conns               0
C tcp pxl established conns              0   C tcp pxl closed conns                  0
outbound cpasxl packets                  0   outbound pslxl packets                  0
outbound cpasxl bytes                    0   outbound pslxl bytes                    0
DNS DoR stats                            0
(*) Statistics marked with C refer to current value, others refer to total value
```

Example: fwaccel stats -m

Example of statistics for multicast traffic:

```
Name                  Value              Name                  Value
--------------------  ---------------    --------------------  ---------------
in packets                          0    out packets                         0
if restricted                       0    conns with down if                  0
f2f packets                         0    f2f bytes                           0
dropped packets                     0    dropped bytes                       0
accel packets                       0    accel bytes                         0
mcast conns                         0
```

Example: fwaccel stats -n

Example of statistics for Identity Awareness (NAC):

```
Name                  Value              Name                  Value
--------------------  ---------------    --------------------  ---------------
NAC packets                         0    NAC bytes                           0
NAC connections                     0    compliance failure                  0
```

Example: fwaccel stats -o

Example of statistics for Reorder Infrastructure:

```
Appliaction: F2V
        Statistic                       Value
-----------------------------------     --------------------
                        Queued pkts                    0
                    Max queued pkts                    0
                    Timer triggered                    0
          Callback hahndling unhold                    0
 Callback hahndling unhold and drop                    0
           Callback hahndling reset                    0
              Dequeued pkts resumed                    0
                Queue ent allocated                    0
                    Queue ent freed                    0
                   Queues allocated                    0
                       Queues freed                    0
                     Ack notif sent                    0
              Ack respones handling                    0
              Dequeued pkts dropped                    0
       Reached max queued pkt limit                    0
                   Set timer failed                    0
                 Error already held                    0
             Queue ent alloc failed                    0
                 Queue alloc failed                    0
                   Ack notif failed                    0
       Ack respones handling failed                    0
----------------------------------------------------

Appliaction: Route
        Statistic                       Value
-----------------------------------     --------------------
                        Queued pkts                    0
                    Max queued pkts                    0
                    Timer triggered                    0
          Callback hahndling unhold                    0
 Callback hahndling unhold and drop                    0
           Callback hahndling reset                    0
              Dequeued pkts resumed                    0
                Queue ent allocated                    0
                    Queue ent freed                    0
                   Queues allocated                    0
                       Queues freed                    0
                     Ack notif sent                    0
              Ack respones handling                    0
              Dequeued pkts dropped                    0
       Reached max queued pkt limit                    0
                   Set timer failed                    0
                 Error already held                    0
             Queue ent alloc failed                    0
                 Queue alloc failed                    0
                   Ack notif failed                    0
       Ack respones handling failed                    0
----------------------------------------------------

Appliaction: New connection
        Statistic                       Value
-----------------------------------     --------------------
                        Queued pkts                    0
                    Max queued pkts                    0
                    Timer triggered                    0
          Callback hahndling unhold                    0
 Callback hahndling unhold and drop                    0
           Callback hahndling reset                    0
              Dequeued pkts resumed                    0
                Queue ent allocated                    0
                    Queue ent freed                    0
                   Queues allocated                    0
                       Queues freed                    0
                     Ack notif sent                    0
              Ack respones handling                    0
              Dequeued pkts dropped                    0
       Reached max queued pkt limit                    0
                   Set timer failed                    0
                 Error already held                    0
             Queue ent alloc failed                    0
                 Queue alloc failed                    0
                   Ack notif failed                    0
       Ack respones handling failed                    0
----------------------------------------------------

Appliaction: F2P
        Statistic                       Value
-----------------------------------     --------------------
                        Queued pkts                    0
                    Max queued pkts                    0
                    Timer triggered                    0
          Callback hahndling unhold                    0
 Callback hahndling unhold and drop                    0
           Callback hahndling reset                    0
              Dequeued pkts resumed                    0
                Queue ent allocated                    0
                    Queue ent freed                    0
                   Queues allocated                    0
                       Queues freed                    0
                     Ack notif sent                    0
              Ack respones handling                    0
              Dequeued pkts dropped                    0
       Reached max queued pkt limit                    0
                   Set timer failed                    0
                 Error already held                    0
             Queue ent alloc failed                    0
                 Queue alloc failed                    0
                   Ack notif failed                    0
       Ack respones handling failed                    0
----------------------------------------------------
```

Example: fwaccel stats -p

Example of statistics for SecureXL violations (F2F packets):

```
F2F packets:
--------------
Violation             Packets            Violation             Packets
--------------------  ---------------    --------------------  ---------------
pkt has IP options                  0    ICMP miss conn                   3036
TCP-SYN miss conn                   8    TCP-other miss conn             32224
UDP miss conn                    3772    other miss conn                     0
VPN returned F2F                    0    uni-directional viol                0
possible spoof viol                 0    TCP state viol                      0
out if not def/accl                 0    bridge, src=dst                     0
routing decision err                0    sanity checks failed                0
fwd to non-pivot                    0    broadcast/multicast                 0
cluster message                     0    cluster forward                     0
chain forwarding                    0    F2V conn match pkts                 0
general reason                      0    route changes                       0
```

Example: fwaccel stats -q

Example of statistics for notifications the SecureXL sent to the Firewall:

```
Notification           Packets           Notification           Packets
---------------------  --------------    ---------------------  --------------
ntSAAboutToExpire                   0    ntSAExpired                         0
ntMSPIError                         0    ntNoInboundSA                       0
ntNoOutboundSA                      0    ntDataIntegrityFailed               0
ntPossibleReplay                    0    ntReplay                            0
ntNextProtocolError                 0    ntCPIError                          0
ntClearTextPacket                   0    ntFragmentation                     0
ntUpdateUdpEncTable                 0    ntSASync                            0
ntReplayOutOfWindow                 0    ntVPNTrafficReport                  0
ntConnDeleted                       0    ntConnUpdate                        0
ntPacketDropped                     0    ntSendLog                           0
ntRefreshGTPTunnel                  0    ntMcastDrop                         0
ntAccounting                        0    ntAsyncIndex                        0
ntACkReordering                     0    ntAccelAckInfo                      0
ntMonitorPacket                     0    ntPacketCapture                     0
ntCpasPacketCapture                 0    ntPSLGlueUpdateReject               0
ntSeqVerifyDrop                     0    ntPacketForwardBefore               0
ntICMPMessage                       0    ntQoSReclassifyPacket               0
ntQoSResumePacket                   0    ntVPNEncHaLinkFailure               0
ntVPNEncLsLinkFailure               0    ntVPNEncRouteChange                 0
ntVPNDecVerRouteChang               0    ntVPNDecRouteChange                 0
ntMuxSimToFw                        0    ntPSLEventLog                       0
ntSendCPHWDStats                14871    ntPacketTaggingViolat               0
ntDosNotify                        28    ntSynatkNotify                      0
ntSynatkStats                       0    ntQoSEventLog                       0
ntPrintGetParam                     0
```

Example: fwaccel stats -x

Example of statistics for PXL:

```
PXL Release Context statistics:

 Name                     Value           Name                     Value
-----------------------  ------------    -----------------------  ------------
End Handler                         0    Post Sync                           0
Stop Stream                         0    kbuf fail                           0
Set field failure                   0    Notif set field fail                0
Non SYN seq fail                    0    Tmpl kbuf fail                      0
Tmpl set field fail                 0    Segment Injection                   0
Init app fail                       0    Expiration                          0
Newconn set field fail              0    Newconn fail                        0
CPHWD dec                           0    No PSL policy                       0

PXL Exception statistics:

 Name                     Value           Name                     Value
-----------------------  ------------    -----------------------  ------------
urgent packets                      0    invalid SYN retrans                 0
SYN seq not init                    0    old pkts out win                    0
old pkts out win trunc              0    old pkts out win strip              0
new pkts out win                    0    incorrect retrans                   0
TCP pkts with bad csum              0    ACK unprocessed data                0
old ACK out win                     0    Max segments reached                0
No resources                        0    Hold timeout                        0
```

fwaccel synatk

Description

The *fwaccel synatk* and *fwaccel6 synatk* commands control the Accelerated SYN Defender on the local Security Gateway, or Cluster Member.

Important - See sk120476 for information about the 'SYN Attack' protection in SmartConsole.

Syntax for IPv4

```
fwaccel synatk
      -a
      -c <options>
      -d
      -e
      -g
      -m
      -t <options>
      config
      monitor <options>
      state <options>
      whitelist <options>
```

Syntax for IPv6

```
fwaccel6 synatk
      -a
      -c <options>
      -d
      -e
      -g
      -m
      -t <options>
      config
      monitor <options>
      state <options>
      whitelist <options>
```

Parameters

Parameter	Description
No Parameters	Shows the applicable built-in usage.
`-a`	Applies the configuration from the default file. See *"fwaccel synatk -a" on page 119*.
`-c <options>`	Applies the configuration from the specified file. See *"fwaccel synatk -c <Configuration File>" on page 120*.
`-d`	Disables the Accelerated SYN Defender on all interfaces. See *"fwaccel synatk -d" on page 121*.
`-e`	Enables the Accelerated SYN Defender on interfaces with topology "External". Enables the Accelerated SYN Defender in Monitor (Detect only) mode on interfaces with topology "Internal". See *"fwaccel synatk -e" on page 122*.
`-g`	Enables the Accelerated SYN Defender on all interfaces. See *"fwaccel synatk -g" on page 123*.
`-m`	Enables the Accelerated SYN Defender in Monitor (Detect only) mode on all interfaces. In this state, the Accelerated SYN Defender only sends a log when it recognizes a TCP SYN Flood attack. See *"fwaccel synatk -m" on page 124*.
`-t <options>`	Configures the threshold numbers of half-opened TCP connections that trigger the Accelerated SYN Defender. See *"fwaccel synatk -t <Threshold>" on page 125*.
`config`	Shows the current Accelerated SYN Defender configuration. See *"fwaccel synatk config" on page 132*.
`monitor <options>`	Shows the Accelerated SYN Defender status. See *"fwaccel synatk monitor" on page 135*.
`state <options>`	Controls the Accelerated SYN Defender states. See *"fwaccel synatk state" on page 140*.
`whitelist <options>`	Controls the Accelerated SYN Defender whitelist. See *"fwaccel synatk allow" on page 127*.

fwaccel synatk -a

Description

The "*fwaccel synatk -a*" and "*fwaccel6 synatk -a*" commands apply the Accelerated SYN Defender configuration from the default `$FWDIR/conf/synatk.conf` file.

Notes:

- Both IPv4 and IPv6 use the same configuration file.
- Interface specific state settings that you define in the configuration file, override the settings that you define with these commands:
  - *"fwaccel synatk -d" on page 121*
  - *"fwaccel synatk -e" on page 122*
  - *"fwaccel synatk -g" on page 123*
  - *"fwaccel synatk -m" on page 124*

Syntax for IPv4

```
fwaccel synatk -a
```

Syntax for IPv6

```
fwaccel6 synatk -a
```

fwaccel synatk -c <Configuration File>

Description

The "*fwaccel synatk -c <Configuration File>*" and "*fwaccel6 synatk -c <Configuration File>*" commands apply the Accelerated SYN Defender configuration from the specified file.

Important - If you use this parameter, then it must be the first parameter in the syntax.

Notes:

- Both IPv4 and IPv6 use the same configuration file.
- The state settings of a specific interface that you define in the configuration file, override the settings that you define with these commands:
  - *"fwaccel synatk -d" on page 121*
  - *"fwaccel synatk -e" on page 122*
  - *"fwaccel synatk -g" on page 123*
  - *"fwaccel synatk -m" on page 124*

Syntax for IPv4

```
fwaccel synatk -c <Configuration File>
```

Syntax for IPv6

```
fwaccel6 synatk -c <Configuration File>
```

Parameters

Parameter	Description
`<Configuration File>`	Specifies the full path and the name of the file. For reference, see the default file: `$FWDIR/conf/synatk.conf`

fwaccel synatk -d

Description

The "*fwaccel synatk -d*" and "*fwaccel6 synatk -d*" commands disable the Accelerated SYN Defender on all interfaces.

Notes:

- This command:
  1. Modifies the default configuration file `$FWDIR/conf/synatk.conf`, or the configuration file specified with the "`-c`" parameter.
  2. Loads the modified file.
  3. Does not show any output.
- Output of the *"fwaccel synatk monitor" on page 135* command shows:
  - In the row "`Configuration`": `Disabled`
  - In the column "`Enforce`": `Disable`
  - In the column "`State (sec)`": `Disable`
- Output of the *"fwaccel synatk config" on page 132* command shows:
  - In the row "`enabled`": `0`
  - In the row "`enforce`": `0`

Syntax for IPv4

```
fwaccel synatk -d
```

Syntax for IPv6

```
fwaccel6 synatk -d
```

fwaccel synatk -e

Description

The "*fwaccel synatk -e*" and "*fwaccel6 synatk -e*" commands:

- Enable the Accelerated SYN Defender on interfaces with topology "External".
- Enable the Accelerated SYN Defender in Monitor (Detect only) mode on interfaces with topology "Internal".

Notes:

- This command:
  1. Modifies the default configuration file `$FWDIR/conf/synatk.conf`, or the configuration file specified with the "`-c`" parameter.
  2. Loads the modified file.
- Output of the *"fwaccel synatk monitor" on page 135* command shows for "External" interfaces:
  - Configuration: `Enforcing`
  - Enforce: `Prevent`
  - State: `Ready` (may change later depending on what the SYN Defender detects)
- Output of the *"fwaccel synatk monitor" on page 135* command shows for "Internal" interfaces:
  - Configuration: `Enforcing`
  - Enforce: `Detect`
  - State: `Monitor`
- Output of the *"fwaccel synatk config" on page 132* command shows:
  - `enabled 1`
  - `enforce 1`

Syntax for IPv4

```
fwaccel synatk -e
```

Syntax for IPv6

```
fwaccel6 synatk -e
```

fwaccel synatk -g

Description

The "*fwaccel synatk -g*" and "*fwaccel6 synatk -g*" commands enable the Accelerated SYN Defender on all interfaces.

Notes:

- This command:
  1. Modifies the default configuration file `$FWDIR/conf/synatk.conf`, or the configuration file specified with the "`-c`" parameter.
  2. Loads the modified file.
- Output of the *"fwaccel synatk monitor" on page 135* command shows for "External" interfaces:
  - Configuration: `Enforcing`
  - Enforce: `Prevent`
  - State: `Ready` (may change later depending on what the SYN Defender detects)
- Output of the *"fwaccel synatk monitor" on page 135* command shows for "Internal" interfaces:
  - Configuration: `Enforcing`
  - Enforce: `Detect`
  - State: `Monitor`
- Output of the *"fwaccel synatk config" on page 132* command shows:
  - `enabled 1`
  - `enforce 2`

Syntax for IPv4

```
fwaccel synatk -g
```

Syntax for IPv6

```
fwaccel6 synatk -g
```

fwaccel synatk -m

Description

The "*fwaccel synatk -m*" and "*fwaccel6 synatk -m*" commands enable the Accelerated SYN Defender in Monitor (Detect only) mode on all interfaces.

In this state, the Accelerated SYN Defender only sends a log when it recognizes a TCP SYN Flood attack.

Notes:

- This command:
  1. Modifies the default configuration file `$FWDIR/conf/synatk.conf`, or the configuration file specified with the "`-c`" parameter.
  2. Loads the modified file.
- Output of the *"fwaccel synatk monitor" on page 135* command shows:
  - Configuration: `Monitoring`
  - Enforce: `Detect`
  - State: `Monitor`
- Output of the *"fwaccel synatk config" on page 132* command shows:
  - `enabled 1`
  - `enforce 0`

Syntax for IPv4

```
fwaccel synatk -m
```

Syntax for IPv6

```
fwaccel6 synatk -m
```

fwaccel synatk -t <Threshold>

Description

The "*fwaccel synatk -t <Threshold>*" and "*fwaccel6 synatk -t <Threshold>*" commands configure the threshold numbers of half-opened TCP connections that trigger the Accelerated SYN Defender.

Notes:

- This command:
  1. Modifies the default configuration file `$FWDIR/conf/synatk.conf`, or the configuration file specified with the "`-c`" parameter.
  2. Loads the modified file.
- Threshold values are independent for IPv4 and IPv6.

Syntax for IPv4

```
fwaccel synatk -t <Threshold>
```

Syntax for IPv6

```
fwaccel6 synatk -t <Threshold>
```

Thresholds

- The **Global high attack threshold** number is configured to the specified value `<Threshold>`.

  This is the number of half-open TCP connections on all interfaces required for the Accelerated SYN Defender to engage.

  - Valid values: 100 and greater
  - Default: 10000

- The **High attack threshold** number is configured to 1/2 of the specified value `<Threshold>`.

  This is the high number of half-open TCP connections on an interface required for the Accelerated SYN Defender to engage.

  - Valid values: (Low attack threshold) < (High attack threshold) <= (Global high attack threshold)
  - Default: 5000

- The **Low attack threshold** number is configured to 1/10 of the specified value `<Threshold>`.

  This is the low number of half-open TCP connections on an interface required for the Accelerated SYN Defender to engage.

  - Valid values: 10 and greater
  - Default: 1000

fwaccel synatk allow

Description

The "*fwaccel synatk allow*" and "*fwaccel6 synatk allow*" commands control the Accelerated SYN Defender allow-list.

Notes:

- This allow-list overrides which packet the Accelerated SYN Defender drops. Before you use a 3rd-party or automatic blacklists, add trusted networks and hosts to the allow-list to avoid outages.
- Also, see the *"fwaccel dos allow" on page 52* command.

Important - In Cluster, you must configure the Rate Limiting in the same way on all the Cluster Members.

Syntax for IPv4

```
fwaccel synatk allow
      -a <IPv4 Address>[/<Subnet Prefix>]
      -d <IPv4 Address>[/<Subnet Prefix>]
      -F
      -l /<Path>/<Name of File>
      -L
      -s
```

Syntax for IPv6

```
fwaccel6 synatk allow
      -a <IPv6 Address>[/<Subnet Prefix>]
      -d <IPv6 Address>[/<Subnet Prefix>]
      -F
      -l /<Path>/<Name of File>
      -L
      -s
```

Parameters

Parameter	Description
No Parameters	Shows the applicable built-in usage.

Parameter	Description

`-a <IPv4 Address> [/<Subnet Prefix>]`	Adds the specified IPv4 address to the Accelerated SYN Defender allow-list.
	■ `<IPv4 Address>`
	Can be an IPv4 address of a network or a host.
	■ `<Subnet Prefix>`
	Must specify the length of the subnet mask in the format `/<bits>`.
	Optional for a host IPv4 address.
	Mandatory for a network IPv4 address.
	Range - from /1 to /32.
	Important - If you do not specify the subnet prefix explicitly, this command uses the subnet prefix /32.
	Examples:
	■ For a host:
	`192.168.20.30`
	`192.168.20.30/32`
	■ For a network:
	`192.168.20.0/24`

`-a <IPv6 Address> [/<Subnet Prefix>]`	Adds the specified IPv6 address to the Accelerated SYN Defender allow-list.
	■ `<IPv6 Address>`
	Can be an IPv6 address of a network or a host.
	■ `<Subnet Prefix>`
	Must specify the length of the subnet mask in the format `/<bits>`.
	Optional for a host IPv6 address.
	Mandatory for a network IPv6 address.
	Range - from /1 to /128.
	Important - If you do not specify the subnet prefix explicitly, this command uses the subnet prefix /128.
	Examples:
	■ For a host:
	`2001:0db8:85a3:0000:0000:8a2e:0370:7334`
	`2001:0db8:85a3:0000:0000:8a2e:0370:7334/128`
	■ For a network:
	`2001:cdba:9abc:5678::/64`

Parameter	Description
`-d <IPv4 Address> [/<Subnet Prefix>]`	Removes the specified IPv4 address from the Accelerated SYN Defender allow-list. ■ `<IPv4 Address>` Can be an IPv4 address of a network or a host. ■ `<Subnet Prefix>` Optional. Must specify the length of the subnet mask in the format `/<bits>`. Optional for a host IPv4 address. Mandatory for a network IPv4 address. Range - from /1 to /32. **Important** - If you do not specify the subnet prefix explicitly, this command uses the subnet prefix /32.
`-d <IPv6 Address> [/<Subnet Prefix>]`	Removes the specified IPv6 address from the Accelerated SYN Defender allow-list. ■ `<IPv6 Address>` Can be an IPv6 address of a network or a host. ■ `<Subnet Prefix>` Optional. Must specify the length of the subnet mask in the format `/<bits>`. Optional for a host IPv6 address. Mandatory for a network IPv6 address. Range - from /1 to /128. **Important** - If you do not specify the subnet prefix explicitly, this command uses the subnet prefix /128.
`-F`	Removes (flushes) all entries from the Accelerated SYN Defender allow-list.

Parameter	Description
`-l /<Path>/<Name of File>`	Loads the Accelerated SYN Defender allow-list entries from the specified plain-text file. **Note** - To replace the current allow-list with the contents of a new file, use both the `-F` and `-l` parameters on the same command line. **Important:** ■ You must manually create and configure this file with the `touch` or `vi` command. ■ You must assign at least the read permission to this file with the `chmod +x` command. ■ Each entry in this file must be on a separate line. ■ Each entry in this file must be in this format: `<IPv4 Address>[/<Subnet Prefix>]` ■ SecureXL ignores empty lines and lines that start with the # character in this file.
`-L`	Loads the Accelerated SYN Defender allow-list entries from the plain-text file with a predefined name: `$FWDIR/conf/synatk-allow-list-v4.conf` Security Gateway automatically runs these commands "`{fwaccel | fwaccel6} synatk allow -L`" during each boot. **Note** - To replace the current allow-list with the contents of a new file, use both the "`-F`" and "`-L`" parameters on the same command line. **Important:** ■ This file does not exist by default. ■ You must manually create and configure this file with the `touch` or `vi` command. ■ You must assign at least the read permission to this file with the `chmod +x` command.. ■ Each entry in this file must be on a separate line. ■ Each entry in this file must be in this format: `<IPv4 Address>[/<Subnet Prefix>]` ■ SecureXL ignores empty lines and lines that start with the # character in this file.
`-s`	Shows the current Accelerated SYN Defender allow-list entries.

Example

```
[Expert@MyGW:0]# fwaccel synatk allow -a 192.168.20.0/24
[Expert@MyGW:0]# fwaccel synatk allow -s
192.168.20.0/24
[Expert@MyGW:0]# fwaccel synatk allow -d 192.168.20.0/24
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel synatk allow -a 192.168.40.55
[Expert@MyGW:0]# fwaccel synatk allow -s
192.168.40.55/32
[Expert@MyGW:0]# fwaccel synatk allow -d 192.168.40.55
```

fwaccel synatk config

Description

The "*fwaccel synatk config*" and "*fwaccel6 synatk config*" commands show the current Accelerated SYN Defender configuration.

Syntax for IPv4

```
fwaccel synatk config
```

Syntax for IPv6

```
fwaccel6 synatk config
```

Example

```
[Expert@MyGW:0]# fwaccel synatk config
enabled 0
enforce 1
global_high_threshold 10000
periodic_updates 1
cookie_resolution_shift 6
min_frag_sz 80
high_threshold 5000
low_threshold 1000
score_alpha 100
monitor_log_interval (msec) 60000
grace_timeout (msec) 30000
min_time_in_active (msec) 60000
[Expert@MyGW:0]#
```

Description of Configuration Parameters

Parameter	Description
`enabled`	Shows if the Accelerated SYN Defender is enabled or disabled. ▪ Valid values: 0 (disabled), 1 (enabled) ▪ Default: 0
`enforce`	When the Accelerated SYN Defender is enabled, shows it enforces the protection. Valid values: ▪ 0 - The Accelerated SYN Defender is in Monitor (Detect only) mode on all interfaces. ▪ 1 - The Accelerated SYN Defender is engaged only on external interfaces when the number of half-open TCP connections exceeds the threshold. ▪ 2 - The Accelerated SYN Defender is engaged on both external and internal interfaces when the number of half-open TCP connections exceeds the threshold.
`global_high_threshold`	Global high attack threshold number. See the *"fwaccel synatk -t <Threshold>" on page 125* command.
`periodic_updates`	For internal Check Point use only. ▪ Valid values: 0 (disabled), 1 (enabled) ▪ Default: 1
`cookie_resolution_shift`	For internal Check Point use only. ▪ Valid values: 1-7 ▪ Default: 6
`min_frag_sz`	During the TCP SYN Flood attack, the Accelerated SYN Defender prevents TCP fragments smaller than this minimal size value. ▪ Valid values: 80 and greater ▪ Default: 80
`high_threshold`	High attack threshold number. See the *"fwaccel synatk -t <Threshold>" on page 125* command.
`low_threshold`	Low attack threshold number. See the *"fwaccel synatk -t <Threshold>" on page 125* command.

Parameter	Description
`score_alpha`	For internal Check Point use only. ■ Valid values: 1-127 ■ Default: 100
`monitor_log_interval (msec)`	Interval, in milliseconds, between successive warning logs in the Monitor (Detect only) mode. ■ Valid values: 1000 and greater ■ Default: 60000
`grace_timeout (msec)`	Maximal time, in milliseconds, to stay in the Grace state (which is a transitional state between Ready and Active). In the Grace state, the Accelerated SYN Defender stops challenging Clients for TCP SYN Cookie, but continues to validate TCP SYN Cookies it receives from Clients. ■ Valid values: 10000 and greater ■ Default: 30000
`min_time_in_active (msec)`	Minimal time, in milliseconds, to stay in the Active mode. In the Active mode, the Accelerated SYN Defender is actively challenging TPC SYN packets with SYN Cookies. ■ Valid values: 10000 and greater ■ Default: 60000

fwaccel synatk monitor

Description

The "*fwaccel synatk monitor*" and "*fwaccel6 synatk monitor*" commands show the Accelerated SYN Defender status.

Important - To enable the Accelerated SYN Defender in Monitor (Detect only) mode on all interfaces, you must run the *"fwaccel synatk -m" on page 124* command.

Syntax for IPv4

```
fwaccel synatk monitor
        [-p]
        [-p] -a
        [-p] -s
        [-p] -v
```

Syntax for IPv6

```
fwaccel6 synatk monitor
        [-p]
        [-p] -a
        [-p] -s
        [-p] -v
```

Parameters

Important - You can specify only one of these parameters: `-a`, `-s`, or `-v`.

Parameter	Description
`-p`	Shows the Accelerated SYN Defender status for each SecureXL instance ("`PPAK ID: 0`" is the Host Security Appliance).
`[-p] -a`	Shows the Accelerated SYN Defender statistics for all interfaces (for each SecureXL instance).
`[-p] -s`	Shows the attack state in short form (for each SecureXL instance).
`[-p] -v`	Shows the attack state in verbose form (for each SecureXL instance).

Examples

Example 1 - Default output before and after enabling the Accelerated SYN Defender

```
[Expert@MyGW:0]# fwaccel synatk monitor
+-----------------------------------------------------------------------------+
| SYN Defender status                                                         |
+-----------------------------------------------------------------------------+
| Configuration                                                      Disabled |
| Status                                                               Normal |
| Non established connections                                               0 |
| Global Threshold                                                      10000 |
| Interface Threshold                                                    5000 |
+-----------------------------------------------------------------------------+
| IF              | Topology | Enforce | State (sec)  | Non-established conns |
|                 |          |         |              | Peak      | Current   |
+-----------------------------------------------------------------------------+
| eth0            | External | Disable | Disable      | N/A       | N/A       |
| eth1            | Internal | Disable | Disable      | N/A       | N/A       |
+-----------------------------------------------------------------------------+
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel synatk -m
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel synatk monitor
+-----------------------------------------------------------------------------+
| SYN Defender status                                                         |
+-----------------------------------------------------------------------------+
| Configuration                                                    Monitoring |
| Status                                                               Normal |
| Non established connections                                               0 |
| Global Threshold                                                      10000 |
| Interface Threshold                                                    5000 |
+-----------------------------------------------------------------------------+
| IF              | Topology | Enforce | State (sec)  | Non-established conns |
|                 |          |         |              | Peak      | Current   |
+-----------------------------------------------------------------------------+
| eth0            | External | Detect  | Monitor      | 0         | 0         |
| eth1            | Internal | Detect  | Monitor      | 0         | 0         |
+-----------------------------------------------------------------------------+
[Expert@MyGW:0]#
```

Example 2 - Showing the Accelerated SYN Defender status for each SecureXL instance

```
[Expert@MyGW:0]# fwaccel synatk monitor -p
+-----------------------------------------------------------------------------+
| SYN Defender status                                                         |
+-----------------------------------------------------------------------------+
| Configuration                                                    Monitoring |
| Status                                                               Normal |
| Non established connections                                               0 |
| Global Threshold                                                      10000 |
| Interface Threshold                                                    5000 |
+-----------------------------------------------------------------------------+
| IF              | Topology | Enforce | State (sec)  | Non-established conns |
|                 |          |         |              | Peak      | Current   |
+-----------------------------------------------------------------------------+
| eth0            | External | Detect  | Monitor      | 0         | 0         |
| eth1            | Internal | Detect  | Monitor      | 0         | 0         |
+-----------------------------------------------------------------------------+

PPAK ID: 0
----------
+-----------------------------------------------------------------------------+
| SYN Defender status                                                         |
+-----------------------------------------------------------------------------+
| Configuration                                                    Monitoring |
| Status                                                               Normal |
| Non established connections                                               0 |
| Global Threshold                                                      10000 |
| Interface Threshold                                                    5000 |
+-----------------------------------------------------------------------------+
| IF              | Topology | Enforce | State (sec)  | Non-established conns |
|                 |          |         |              | Peak      | Current   |
+-----------------------------------------------------------------------------+
| eth0            | External | Detect  | Monitor      | 0         | 0         |
| eth1            | Internal | Detect  | Monitor      | 0         | 0         |
+-----------------------------------------------------------------------------+
[Expert@MyGW:0]#
```

Example 3 - Showing the Accelerated SYN Defender statistics for all interfaces and for each SecureXL instance.

```
[Expert@MyGW:0]# fwaccel synatk monitor -p -a
Global:
  status            attached
  nr_active                0

Firewall
----------
Per-interface:
                            eth0        eth1
                      ----------  ----------
topology                External    Internal
state                    Monitor     Monitor
syn ready                      0           0
syn active prev                0           0
syn active curr                0           0
active_score                   0           0
msec grace                     0           0
msec active                    0           0
sent cookies                   0           0
fail validations               0           0
succ validations               0           0
early packets                  0           0
no conn data                   0           0
bogus syn                      0           0
peak non-estab                 0           0
int sent cookies               0           0
int succ validations           0           0
msec interval                  0           0

PPAK ID: 0
----------
Per-interface:
                            eth0        eth1
                      ----------  ----------
topology                External    Internal
state                    Monitor     Monitor
syn ready                      0           0
syn active prev                0           0
syn active curr                0           0
active_score                   0           0
msec grace                     0           0
msec active                    0           0
sent cookies                   0           0
fail validations               0           0
succ validations               0           0
early packets                  0           0
no conn data                   0           0
bogus syn                      0           0
peak non-estab                 0           0
int sent cookies               0           0
int succ validations           0           0
msec interval                  0           0
[Expert@MyGW:0]#
```

Example 4 - Showing the attack state in short form (for each SecureXL instance)

```
[Expert@MyGW:0]# fwaccel synatk monitor -p -s
M,N,0,0

PPAK ID: 0
----------
M,N,0,0
[Expert@MyGW:0]#
```

Example 5 - Showing the attack state in verbose form (for each SecureXL instance)

```
[Expert@MyGW:0]# fwaccel synatk monitor -p -v
+-----------------------------------------------------------------------------+
| SYN Defender statistics                                                     |
+-----------------------------------------------------------------------------+
| Status                                                               Normal |
| Spoofed SYN/sec                                                           0 |
+-----------------------------------------------------------------------------+
 PPAK ID: 0
----------
+-----------------------------------------------------------------------------+
| SYN Defender statistics                                                     |
+-----------------------------------------------------------------------------+
| Status                                                               Normal |
| Spoofed SYN/sec                                                           0 |
+-----------------------------------------------------------------------------+
[Expert@MyGW:0]#
```

fwaccel synatk state

Description

The "*fwaccel synatk state*" and "*fwaccel6 synatk state*" commands control the Accelerated SYN Defender states.

The states are independent for IPv4 and IPv6.

Important - This command is **not** intended for end-user usage. Transitions between states (Ready, Grace, and Active) occur automatically. This command provides a way to force temporarily a state transition on an interface or group of interfaces.

Syntax for IPv4

```
fwaccel synatk state
      -h
      -a
      -d
      -g
      -i {all | external | internal | <Name of Interface>}
      -m
      -r
```

Syntax for IPv6

```
fwaccel6 synatk state
      -h
      -a
      -d
      -g
      -i {all | external | internal | <Name of Interface>}
      -m
      -r
```

Parameters

Important - You can specify only one of these parameters: `-a`, `-d`, `-g`, `-m`, or `-r`.

Parameter	Description
`-h`	Shows the applicable built-in usage.
`-a`	Sets the state to Active.
`-d`	Sets the state to Disabled.
`-g`	Sets the state to Grace.
`-i all`	Applies the change to all interfaces (this is the default).
`-i external`	Applies the change only to external interfaces.
`-i internal`	Applies the change only to internal interfaces.
`-i` *`<Name of Interface>`*	Applies the change to the specified interface.
`-m`	Sets the state to Monitor (Detect only) mode.
`-r`	Sets the state to Ready.

fwaccel tab

Description

The *fwaccel tab* and *fwaccel6 tab* commands show the contents of the specified SecureXL kernel table.

Notes:

- Dynamic tables, such as the `connections` table can change while this command prints their contents.
  This may cause some values to be missed or reported twice.
- For some tables, the command prints their contents on the screen.
- For some tables, the command prints their contents to the `/var/log/messages` file.
- Also, see the `fw tab` command.

Syntax for IPv4

```
fwaccel tab [-f] [-m <Number of Rows>] -t <Name of Kernel Table>
```

```
fwaccel tab -s -t <Name of Kernel Table>
```

Syntax for IPv6

```
fwaccel6 tab [-f] [-m <Number of Rows>] -t <Name of Kernel Table>
```

```
fwaccel6 tab -s -t <Name of Kernel Table>
```

Parameters

Parameter	Description
No Parameters	Shows the applicable built-in usage.
`-f`	Formats the output. We recommend to always use this parameter.
`-m <Number of Rows>`	Specifies how many rows to show from the kernel table. Note - The command counts from the top of the table. Default : 1000
`-s`	Shows summary information only.

Parameter	Description
`-t <Name of Kernel Table>`	Specifies the kernel table. This command supports only these kernel tables: ■ `connections` ■ `dos_ip_blacklists` ■ `dos_pbox` ■ `dos_pbox_violating_ips` ■ `dos_rate_matches` ■ `dos_rate_track_src` ■ `dos_rate_track_src_svc` ■ `drop_templates` ■ `frag_table` ■ `gtp_apns` ■ `gtp_tunnels` ■ `if_by_name` ■ `inbound_SAs` ■ `invalid_replay_counter` ■ `ipsec_mtu_icmp` ■ `mcast_drop_conns` ■ `outbound_SAs` ■ `PMTU_table` ■ `<Profile>` ■ `reset_table` ■ `vpn_link_selection` ■ `vpn_trusted_ifs`

Examples

```
[Expert@MyGW:0]# fwaccel tab -f -m 200 -t connections
Table connections is empty
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t inbound_SAs
Table contents written to /var/log/messages.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t outbound_SAs
Table contents written to /var/log/messages.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t vpn_link_selection
Table contents written to /var/log/messages.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t drop_templates
Table drop_templates is empty
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t vpn_trusted_ifs
Table contents written to /var/log/messages.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t profile
Table contents written to /var/log/messages.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t mcast_drop_conns
Table contents written to /var/log/messages.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t invalid_replay_counter
Table contents written to /var/log/messages.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t ipsec_mtu_icmp
Table contents written to /var/log/messages.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t gtp_tunnels
Table contents written to /var/log/messages.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t gtp_apns
Table contents written to /var/log/messages.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t if_by_name
Table contents written to /var/log/messages.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t PMTU_table
Table PMTU_table is empty
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t frag_table
Table frag_table is empty
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t reset_table
Table reset_table is empty
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t dos_ip_blacklists
Table dos_ip_blacklists is not active for SecureXL device 0.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t dos_pbox
Table dos_pbox is not active for SecureXL device 0.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t dos_rate_matches
Table dos_rate_matches is not active for SecureXL device 0.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t dos_rate_track_src
Table dos_rate_track_src is not active for SecureXL device 0.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t dos_rate_track_src_svc
Table dos_rate_track_src_svc is not active for SecureXL device 0.
[Expert@MyGW:0]#
```

```
[Expert@MyGW:0]# fwaccel tab -t dos_pbox_violating_ips
Table dos_pbox_violating_ips is not active for SecureXL device 0.
[Expert@MyGW:0]#
```

fwaccel templates

Description

The *fwaccel templates* and *fwaccel6 templates* commands show the contents of the SecureXL templates tables:

- Accept Templates
- Drop Templates

  Important - By default, the Drop Templates are disabled.
  To enable the Drop Templates:

  1. In SmartConsole, open the Security Gateway / Cluster object.
  2. In the left tree, click the **Optimizations** pane.
  3. Select **Enable drop optimization**.
  4. Click **OK**.
  5. Install the Access Control policy.

Important - Based on the number of current templates, these commands can consume memory at very high level.

Syntax for IPv4

```
fwaccel templates
      [-h]
      [-d]
      [-m <Number of Rows>]
      [-s]
      [-S]
```

Syntax for IPv6

```
fwaccel6 templates
      [-h]
      [-d]
      [-m <Number of Rows>]
      [-s]
      [-S]
```

Parameters

Parameter	Description
No Parameters	Shows the contents of the SecureXL Accept Templates table (Table Name - `cphwd_tmpl`, Table ID - 8111).
`-h`	Shows the applicable built-in usage.
`-d`	Shows the contents of the SecureXL Drop Templates table.
`-m <Number of Rows>`	Specifies how many rows to show from the templates table. Note - The command counts from the top of the table. Default : 1000
`-s`	Shows the summary of SecureXL Connections Templates (number of templates)
`-S`	Shows statistics for the SecureXL Connections Templates.

Accept Templates flags

One or more of these flags appears in the output:

Flag	Description
A	Connection is accounted (SecureXL counts the number of packets and bytes).
B	Connection is created for a rule that contains an Identity Awareness object, or for a rule below that rule.
E	Connection is created for a NAT rule that contains an Identity Awareness object.
I	Identity Awareness (NAC) is enabled for this connection.
M	Connection is created for a rule that contains a Domain object, or for a rule below that rule.
N	Connection undergoes NAT.
O	Connection is created for a rule that contains a Dynamic object, or for a rule below that rule.
Q	QoS is enabled for this connection.
R	Connection is created for a rule that contains a Traceroute object, or for a rule below that rule.
S	PXL (combination of SecureXL and PSL (Passive Streaming Library)) is enabled for this connection.
T	Connection is created for a rule that contains a Time object, or for a rule below that rule.
U	Connection is unidirectional.
X	Connection is created for a NAT rule that contains a translated Dynamic object.
Z	Connection is created for a rule that contains a Security Zone object, or for a rule below that rule.

Drop Templates flags

One or more of these flags appears in the output:

Flag	Description
D	Drop template exists for this connection.
L	Log and Drop action for this connection.

Examples

Example 1 - Default output

```
[Expert@MyGW:0]# fwaccel templates
Source          SPort Destination     DPort PR Flags         LCT  DLY C2S i/f S2C i/f
--------------- ----- --------------- ----- -- ------------  ---- --- ------- -------
192.168.10.20       * 192.168.10.50      80  6            0     0   0 eth5/eth1 eth1/eth5
[Expert@MyGW:0]#
```

Example 2 - Drop Templates

```
[Expert@MyGW:0]# fwaccel templates -d
The SecureXL drop templates table is empty
[Expert@MyGW:0]#
```

Example 3 - Summary of SecureXL Connections Templates

```
[Expert@MyGW:0]# fwaccel templates -s
Total number of templates: 1
[Expert@MyGW:0]#
```

Example 4 - Templates statistics

```
[Expert@MyGW:0]# fwaccel templates -S

Templates stats:

 Name                     Value           Name                     Value
--------------------     ------------    --------------------     ------------
C templates                         0    conns from templates                0
nat templates                       0    conns from nat tmpl                 0
C CPASXL templates                  0    C PSLXL templates                   0
C used templates                    0    cpasxl tmpl conns                   0
pslxl tmpl conns                    0    C conns from tmpl                   0

[Expert@MyGW:0]#
```

fwaccel ver

Description

Shows this information:

- Firewall Version and Build
- Accelerator Version
- Firewall API version
- Accelerator API version

Syntax

```
fwaccel ver
```

Example

```
Expert@MyGW:0]# fwaccel ver
Firewall version: R81 - Build 123
Acceleration Device: Performance Pack
Accelerator Version 2.1
Firewall API version: 3.0NG (19/11/2015)
Accelerator API version: 3.0NG (19/11/2015)
[Expert@MyGW:0]#
```

fw monitor

Description

Firewall Monitor is the Check Point traffic capture tool.

In a Security Gateway, traffic passes through different inspection points - Chain Modules in the Inbound direction and then in the Outbound direction (see the "`fw ctl chain`" command.

The FW Monitor tool captures the traffic at each Chain Module in both directions.

You can later analyze the captured traffic with the same FW Monitor tool, or with special tools like Wireshark.

Notes:

- Only one instance of "`fw monitor`" can run at a time.
- You can stop the "`fw monitor`" instance in one of these ways:
  - In the shell, in which the "`fw monitor`" instance runs, press **CTRL + C** keys
  - In another shell, run this command: `fw monitor -U`
- Each time you run the FW Monitor, it compiles its temporary policy files (`$FWDIR/tmp/monitorfilter.*`).
- From R80.20, the FW Monitor is able to show the traffic accelerated with SecureXL.
- For more information, see sk30583 and How to use FW Monitor.

Syntax for IPv4

```
fw monitor {-h | -help}
```

```
fw monitor [-d] [-D] [-ci <Number of Inbound Packets>] [-co
<Number of Outbound Packets>] [-e <INSPECT Expression> | -f
{<INSPECT Filter File> | -}] [-F "<Source IP>,<Source Port>,<Dest
IP>,<Dest Port>,<Protocol Number>"] [-i] [-l <Length>] [-m
{i,I,o,O,e,E}] [-o <Output File> [-w]] [[-pi <Position>] [-pI
<Position>] [-po <Position>] [-pO <Position>] | -p all [-a]] [-T]
[-u | -s] [-U] [-v <VSID>] [-x <Offset>[,<Length>] [-w]]
```

Syntax for IPv6

```
fw6 monitor {-h | -help}
```

```
fw6 monitor [-d] [-D] [-ci <Number of Inbound Packets>] [-co
<Number of Outbound Packets>] [-e <INSPECT Expression> | -f
{<INSPECT Filter File> | -}] [-F "<Source IP>,<Source Port>,<Dest
IP>,<Dest Port>,<Protocol Number>"] [-i] [-l <Length>] [-m
{i,I,o,O,e,E}] [-o <Output File> [-w]] [[-pi <Position>] [-pI
<Position>] [-po <Position>] [-pO <Position>] | -p all [-a]] [-T]
[-u | -s] [-U] [-v <VSID>] [-x <Offset>[,<Length>] [-w]]
```

Parameters

Parameter	Description
`{-h | -help}`	Shows the built-in usage.
`-d` `-D`	Runs the command in debug mode and shows some information about how the FW Monitor starts and compiles the specified INSPECT filter: ■ `-d` Simple debug output. ■ `-D` Verbose output. **Note** - You can specify both parameters to show more information.

`-ci` *`<Number of Inbound Packets>`*
`-co` *`<Number of Outbound Packets>`*
	Specifies how many packets to capture.
	The FW Monitor stops the traffic capture if it counted the specified number of packets.
	■ `-ci`
	Specifies the number of inbound packets to count.
	■ `-co`
	Specifies the number of inbound packets to count
	Best Practice - You can use the "`-ci`" and the "`-co`" parameters together. This is especially useful during large volumes of traffic. In such scenarios, FW Monitor may bind so many resources (for writing to the console, or to a file) that recognizing the break sequence (**CTRL+C**) might take a very long time.

Parameter	Description

`-e <INSPECT Expression>`
or
`-f {<INSPECT Filter File> | -}`
	Captures only specific packets of non-accelerated traffic:
	■ `"-e <INSPECT Expression>"`
	Defines the INSPECT filter expression on the command line.
	■ `"-f <INSPECT Filter File>"`
	Reads the INSPECT filter expression from the specified file. You must enter the full path and name of the plain-text file that contains the INSPECT filter expression.
	■ `"-f -"`
	Reads the INSPECT filter expression from the standard input. After you enter the INSPECT filter expression, you must enter the `^D` (**CTRL+D**) as the EOF (End Of File) character.
	Warning - These INSPECT filters do **not** apply to the accelerated traffic.
	Important - Make sure to enclose the INSPECT filter expression correctly in single quotes (ASCII value 39) or double quotes (ASCII value 34).
	Notes:
	■ Refer to the `$FWDIR/lib/fwmonitor.def` file for useful macro definitions.
	■ See syntax examples below (*"Examples for the "-e" parameter" on page 167*).

`-F "<Source IP>,<Source Port>,<Dest IP>,<Dest Port>,<Protocol Number>"`	Specifies the capture filter (for both accelerated and non-accelerated traffic): ■ `<Source IP>` - Specifies the source IP address ■ `<Source Port>` - Specifies the source Port Number (see *IANA Service Name and Port Number Registry*) ■ `<Dest IP>` - Specifies the destination IP address ■ `<Dest Port>` - Specifies the destination Port Number (see *IANA Service Name and Port Number Registry*) ■ `<Protocol Number>` - Specifies the Protocol Number (see *IANA Protocol Numbers*)

Parameter	Description

	Notes:
	■ See syntax examples below (*"Examples for the "-F" parameter" on page 181*).
	■ The "`-F`" parameter uses these Kernel Debug Filters. For more information, see *"Kernel Debug Filters" on page 427*.
	• For the Source IP address:
	`simple_debug_filter_saddr_<N> "<IP Address>"`
	• For the Source Ports:
	`simple_debug_filter_sport_<N> <1-65535>`
	• For the Destination IP address:
	`simple_debug_filter_daddr_<N> "<IP Address>"`
	• For the Destination Ports:
	`simple_debug_filter_dport_<N> <1-65535>`
	• For the Protocol Number:
	`simple_debug_filter_proto_<N> <0-254>`
	■ Value 0 means "`any`".
	■ This parameter supports up to 5 capture filters (up to 5 instances of the "`-F`" parameter in the syntax).
	The FW Monitor performs the logical "OR" between all specified simple capture filters.

`-H`	Creates an IP address filter. For more information, see *"Kernel Debug Filters" on page 427*. This parameter supports up to 3 capture filters (up to 3 instances of the "`-H`" parameter in the syntax). Example - Capture only HTTP traffic to and from the Host 1.1.1.1: `fw ctl debug -H "1.1.1.1"`

Parameter	Description
`-i`	Flushes the standard output. **Note** - This parameter is valid only with the "`-v <VSID>`" parameter. **Best Practice** - Use this parameter to make sure FW Monitor immediately writes the captured data for each packet to the standard output. This is especially useful if you want to kill a running FW Monitor process, and want to be sure that FW Monitor writes all the data to the specified file.
`-l <Length>`	Specifies the maximal length of the captured packets. FW Monitor reads only the specified number of bytes from each packet. **Notes:** ▪ This parameter is optional. ▪ With this parameter you can capture only the headers from each packet (for example, IP and TCP) and omit the payload. This decreases the size of the output file. This also helps the internal FW Monitor buffer not to fill too fast. ▪ Make sure to capture the minimal required number of bytes, to capture the Layer 3 IP header and Layer 4 Transport header.

Parameter	Description
`-m {i, I, o, O, e, E}`	Specifies the capture mask (inspection point) in relation to Chain Modules, in which the FW Monitor captures the traffic. These are the inspection points, through which each packet passes on a Security Gateway. ▪ `-m i` Pre-Inbound only (before the packet enters a Chain Module in the inbound direction) ▪ `-m I` Post-Inbound only (after the packet passes a Chain Module in the inbound direction) ▪ `-m o` Pre-Outbound only (before the packet enters a Chain Module in the outbound direction) ▪ `-m O` Post-Outbound only (after the packet passes through a Chain Module in the outbound direction) ▪ `-m e` Pre-Outbound VPN only (before the packet enters a VPN Chain Module in the outbound direction) ▪ `-m E` Post-Outbound VPN only (after the packet passes through a VPN Chain Module in the outbound direction)

Parameter	Description
	Notes: ■ You can specify several capture masks (for example, to see NAT on the egress packets, enter "`... -m o O ...`"). ■ You can use this capture mask parameter "`-m {i, I, o, O, e, E}`" together with the chain module position parameter "`-p{i | I | o | O}`". ■ In the inbound direction: • All chain positions *before* the FireWall Virtual Machine module are Pre-Inbound (the "`fw ctl chain`" command shows this module as "`fw VM inbound`"). • All chain modules *after* the FireWall Virtual Machine module are Post-Inbound. ■ In the outbound direction: • All chain position *before* the FireWall Virtual Machine module are Pre-Outbound. • All chain modules *after* the FireWall Virtual Machine module are Post-Outbound. ■ By default, the FW Monitor captures the traffic only in the FireWall Virtual Machine module. ■ The packet direction relates to each specific packet, and not to the connection's direction. ■ The letters "`q`" and "`Q`" after the inspection point mean that the QoS policy is applied to the interface.

	Example packet flows:
	■ From a Client to a Server through the FireWall Virtual Machine module:
	`[Client] --> ("i") {FW VM attached to eth1} ("I") [Security Gateway] ("o") {FW VM attached to eth2} ("O") --> [Server]`
	■ From a Server to a Client through the FireWall Virtual Machine module:
	`[Client] <-- ("O") {FW VM attached to eth1} ("o") [Security Gateway] ("I") {FW VM attached to eth2} ("i") <-- [Server]`

Parameter	Description
`-o <Output File>`	Specifies the output file, to which FW Monitor writes the captured raw data. **Important** - If you do not specify the path explicitly, FW Monitor creates this output file in the current working directory. Because this output file can grow very fast to very large size, we always recommend to specify the full path to the largest partition `/var/log/`. The format of this output file is the same format used by tools like `snoop` (refer to RFC 1761). You can later analyze the captured traffic with the same FW Monitor tool, or with special tools like Wireshark.

`-pi <Position>`
`-pI <Position>`
`-po <Position>`
`-pO <Position>`
or
`-p all [-a]`
	Inserts the FW Monitor Chain Module at the specified position between the kernel Chain Modules (see the "`fw ctl chain`" command).
	If the FW Monitor writes the captured data to the specified output file (with the parameter "`-o <Output File>`"), it also writes the position of the FW Monitor chain module as one of the fields.
	You can insert the FW Monitor Chain Module in these positions only:
	■ `-pi <Position>`
	Inserts the FW Monitor Chain Module in the specified Pre-Inbound position.
	■ `-pI <Position>`
	Inserts the FW Monitor Chain Module in the specified Post-Inbound position.
	■ `-po <Position>`
	Inserts the FW Monitor Chain Module in the specified Pre-Outbound position.
	■ `-pO <Position>`
	Inserts the FW Monitor Chain Module in the specified Post-Outbound position
	■ `-p all [-a]`
	Inserts the FW Monitor Chain Module at all positions (both Inbound and Outbound).
	Warning - This parameter causes very high load on the CPU, but provides the most complete traffic capture.
	The "`-a`" parameter specifies to use absolute chain positions. This parameter changes the chain ID from a relative value (which only makes sense with the matching output from the "`fw ctl chain`" command) to an absolute value.

Parameter	Description
	Notes: ■ *<Position>* can be one of these: • A relative position number In the output of the "`fw ctl chain`" command, refer to the numbers in the leftmost column (for example, 0, 5, 14). • A relative position alias In the output of the "`fw ctl chain`" command, refer to the internal chain module names in the rightmost column in the parentheses (for example, `sxl_in`, `fw`, `cpas`). • An absolute position In the output of the "`fw ctl chain`" command, refer to the numbers in the second column from the left (for example, -7fffffff, -1fffff8, 7f730000). In the syntax, you must write these numbers in the hexadecimal format (for example, -0x7fffffff, -0x1fffff8, 0x7f730000). ■ You can use this chain module position parameter "`-p{i | I| o | O} ...`" together with the capture mask parameter "`-m {i, I, o, O, e, E}`". ■ In the inbound direction: • All chain positions *before* the FireWall Virtual Machine module are Pre-Inbound (the "`fw ctl chain`" command shows this module as "`fw VM inbound`"). • All chain modules *after* the FireWall Virtual Machine module are Post-Inbound. ■ In the outbound direction: • All chain position *before* the FireWall Virtual Machine module are Pre-Outbound. • All chain modules *after* the FireWall Virtual Machine module are Post-Outbound. ■ By default, the FW Monitor captures the traffic only in the FireWall Virtual Machine module. ■ The chain module position parameters "`-p{i | I| o | O} ...`" parameters do **not** apply to the accelerated traffic, which is still monitored at the default inbound and outbound positions. ■ For more information about the inspection points, see the applicable table below.

Parameter	Description
`-T`	Shows the timestamp for each packet: `DDMMMYYYY HH:MM:SS.mmmmmm` **Best Practice** - Use this parameter if you do not save the output to a file, but print it on the screen.

`-u`
or
`-s`
	Shows UUID for each packet (it is only possible to print either the UUID, or the SUUID - not both):
	- `-u`
	Prints connection's Universal-Unique-ID (UUID) for each packet
	- `-s`
	Prints connection's Session UUID (SUUID) for each packet

`-U`	Removes the simple capture filters specified with this parameter: `-F "<Source IP>,<Source Port>,<Dest IP>,<Dest Port>,<Protocol Number>"`
`-v <VSID>`	On a VSX Gateway or VSX Cluster Member, captures the packets on the specified Virtual System or Virtual Router. By default, FW Monitor captures the packets on all Virtual Systems and Virtual Routers. Example: `fw monitor -v 4 -e "accept;" -o /var/log/fw_mon.cap`
`-w`	Captures the entire packet, instead of only the header. Must be used together with one of these parameters: - `-o <Output File>` - `-x <Offset>[,<Length>]`

Parameter	Description
`-x <Offset> [,<Length>]`	Specifies the position in each packet, where the FW Monitor starts to capture the data from each packet. Optionally, it is also possible to limit the amount of data the FW Monitor captures. ■ `<Offset>` Specifies how many bytes to skip from the beginning of each packet. FW Monitor starts to capture the data from each packet only after the specified number of bytes. ■ `<Length>` Specifies the maximal length of the captured packets. FW Monitor reads only the specified number of bytes from each packet. For example, to skip over the IP header and TCP header, enter "`-x 52,96`"

Inspection points in Security Gateway and in the FW Monitor output

Note - The Inbound and Outbound traffic direction relates to each specific packet, and not to the connection.

- *Inbound*

Name of inspection point	Relation to the FireWall Virtual Machine	Notion of inspection point in the FW Monitor output
Pre-Inbound	Before the inbound FireWall VM	i (for example, `eth4:i`)
Post-Inbound	After the inbound FireWall VM	I (for example, `eth4:I`)
Pre-Inbound VPN	Inbound before decrypt	id (for example, `eth4:id`)
Post-Inbound VPN	Inbound after decrypt	ID (for example, `eth4:ID`)
Pre-Inbound QoS	Inbound before QoS	iq (for example, `eth4:iq`)
Post-Inbound QoS	Inbound after QoS	IQ (for example, `eth4:IQ`)

- *Outbound*

Name of inspection point	Relation to the FireWall Virtual Machine	Notion of inspection point in the FW Monitor output
Pre-Outbound	Before the outbound FireWall VM	o (for example, `eth4:o`)
Post-Outbound	After the outbound FireWall VM	O (for example, `eth4:O`)
Pre-Outbound VPN	Outbound before encrypt	e (for example, `eth4:e`)
Post-Outbound VPN	Outbound after encrypt	E (for example, `eth4:E`)
Pre-Outbound QoS	Outbound before QoS	oq (for example, `eth4:oq`)
Post-Outbound QoS	Outbound after QoS	OQ (for example, `eth4:OQ`)

Generic Examples

Example 1 - Default syntax

```
[Expert@MyGW:0]# fw monitor
 monitor: getting filter (from command line)
 monitor: compiling
monitorfilter:
Compiled OK.
 monitor: loading
 monitor: monitoring (control-C to stop)
[vs_0][fw_1] eth0:i[40]: 192.168.204.1 -> 192.168.204.40 (TCP) len=40 id=31789
TCP: 53901 -> 22 ....A. seq=761113cd ack=f92e2a13
[vs_0][fw_1] eth0:I[40]: 192.168.204.1 -> 192.168.204.40 (TCP) len=40 id=31789
TCP: 53901 -> 22 ....A. seq=761113cd ack=f92e2a13
[vs_0][fw_1] eth0:i[40]: 192.168.204.1 -> 192.168.204.40 (TCP) len=40 id=31790
TCP: 53901 -> 22 ....A. seq=761113cd ack=f92e2a47
... ... ...
 monitor: caught sig 2
 monitor: unloading
[Expert@MyGW:0]#
```

Example 2 - Showing timestamps in the output for each packet

```
[Expert@MyGW:0]# fw monitor -T
 monitor: getting filter (from command line)
 monitor: compiling
monitorfilter:
Compiled OK.
 monitor: loading
 monitor: monitoring (control-C to stop)
[vs_0][fw_1] 12Sep2018 19:08:05.453947 eth0:oq[124]: 192.168.3.53 -> 172.20.168.16 (TCP)
len=124 id=38414
TCP: 22 -> 64424 ...PA. seq=1c23924a ack=3c951092
[vs_0][fw_1] 12Sep2018 19:08:05.453960 eth0:OQ[124]: 192.168.3.53 -> 172.20.168.16 (TCP)
len=124 id=38414
TCP: 22 -> 64424 ...PA. seq=1c23924a ack=3c951092
[vs_0][fw_1] 12Sep2018 19:08:05.454059 eth0:oq[252]: 192.168.3.53 -> 172.20.168.16 (TCP)
len=252 id=38415
TCP: 22 -> 64424 ...PA. seq=1c23929e ack=3c951092
[vs_0][fw_1] 12Sep2018 19:08:05.454064 eth0:OQ[252]: 192.168.3.53 -> 172.20.168.16 (TCP)
len=252 id=38415
TCP: 22 -> 64424 ...PA. seq=1c23929e ack=3c951092
[vs_0][fw_1] 12Sep2018 19:08:05.454072 eth0:oq[252]: 192.168.3.53 -> 172.20.168.16 (TCP)
len=252 id=38416
TCP: 22 -> 64424 ...PA. seq=1c239372 ack=3c951092
[vs_0][fw_1] 12Sep2018 19:08:05.454074 eth0:OQ[252]: 192.168.3.53 -> 172.20.168.16 (TCP)
len=252 id=38416
TCP: 22 -> 64424 ...PA. seq=1c239372 ack=3c951092
[vs_0][fw_1] 12Sep2018 19:08:05.463165 eth0:iq[40]: 172.20.168.16 -> 192.168.3.53 (TCP)
len=40 id=17398
TCP: 64424 -> 22 ....A. seq=3c951092 ack=1c239446
[vs_0][fw_1] 12Sep2018 19:08:05.463177 eth0:IQ[40]: 172.20.168.16 -> 192.168.3.53 (TCP)
len=40 id=17398
TCP: 64424 -> 22 ....A. seq=3c951092 ack=1c239446
 monitor: unloading
[Expert@MyGW:0]#
```

Example 3 - Capturing only three Pre-Inbound packets at the FireWall Virtual Machine module

```
[Expert@MyGW:0]# fw monitor -m i -ci 3
 monitor: getting filter (from command line)
 monitor: compiling
monitorfilter:
Compiled OK.
 monitor: loading
 monitor: monitoring (control-C to stop)
[vs_0][fw_1] eth0:i[40]: 192.168.204.1 -> 192.168.204.40 (TCP) len=40 id=31905
TCP: 53901 -> 22 ....A. seq=76111bb5 ack=f92e683b
[vs_0][fw_1] eth0:i[40]: 192.168.204.1 -> 192.168.204.40 (TCP) len=40 id=31906
TCP: 53901 -> 22 ....A. seq=76111bb5 ack=f92e68ef
[vs_0][fw_1] eth0:i[40]: 192.168.204.1 -> 192.168.204.40 (TCP) len=40 id=31907
TCP: 53901 -> 22 ....A. seq=76111bb5 ack=f92e69a3
 monitor: unloading
Read 3 inbound packets and 0 outbound packets
[Expert@MyGW:0]#
```

Example 4 - Inserting the FW Monitor chain is before the chain #2 and capture only three Pre-Inbound packets

```
[Expert@MyGW:0]# fw ctl chain
in chain (15):
        0: -7fffffff (0000000000000000) (00000000) SecureXL inbound (sxl_in)
        1: -7ffffffe (0000000000000000) (00000000) SecureXL inbound CT (sxl_ct)
        2: -7f800000 (ffffffff8b6718c0) (ffffffff) IP Options Strip (in) (ipopt_strip)
        3: - 1fffff8 (ffffffff8b66f6f0) (00000001) Stateless verifications (in) (asm)
        4: - 1fffff7 (ffffffff8b66f210) (00000001) fw multik misc proto forwarding
        5:         0 (ffffffff8b8506a0) (00000001) fw VM inbound  (fw)
        6:         2 (ffffffff8b671d10) (00000001) fw SCV inbound (scv)
        7:         4 (ffffffff8b061ed0) (00000003) QoS inbound offload chain module
        8:         5 (ffffffff8b564d30) (00000003) fw offload inbound (offload_in)
        9:        10 (ffffffff8b842710) (00000001) fw post VM inbound  (post_vm)
        10:    100000 (ffffffff8b7fd6c0) (00000001) fw accounting inbound (acct)
        11:  22000000 (ffffffff8b0638d0) (00000003) QoS slowpath inbound chain mod (fg_sched)
        12:  7f730000 (ffffffff8b3c40b0) (00000001) passive streaming (in) (pass_str)
        13:  7f750000 (ffffffff8b0e5b40) (00000001) TCP streaming (in) (cpas)
        14:  7f800000 (ffffffff8b671870) (ffffffff) IP Options Restore (in) (ipopt_res)
out chain (14):
        0: -7f800000 (ffffffff8b6718c0) (ffffffff) IP Options Strip (out) (ipopt_strip)
        1: - 1ffffff0 (ffffffff8b0d0190) (00000001) TCP streaming (out) (cpas)
        2: - 1ffff50 (ffffffff8b3c40b0) (00000001) passive streaming (out) (pass_str)
        3: - 1f00000 (ffffffff8b66f6f0) (00000001) Stateless verifications (out) (asm)
        4: -     1ff (ffffffff8aeec0a0) (00000001) NAC Packet Outbound (nac_tag)
        5:         0 (ffffffff8b8506a0) (00000001) fw VM outbound (fw)
        6:        10 (ffffffff8b842710) (00000001) fw post VM outbound  (post_vm)
        7:  15000000 (ffffffff8b062540) (00000003) QoS outbound offload chain modul (fg_pol)
        8:  21000000 (ffffffff8b0638d0) (00000003) QoS slowpath outbound chain mod (fg_sched)
        9:  7f000000 (ffffffff8b7fd6c0) (00000001) fw accounting outbound (acct)
        10:  7f700000 (ffffffff8b0e4660) (00000001) TCP streaming post VM (cpas)
        11:  7f800000 (ffffffff8b671870) (ffffffff) IP Options Restore (out) (ipopt_res)
        12:  7f900000 (0000000000000000) (00000000) SecureXL outbound (sxl_out)
        13:  7fa00000 (0000000000000000) (00000000) SecureXL deliver (sxl_deliver)
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw monitor -pi 2 -ci 3
 monitor: getting filter (from command line)
 monitor: compiling
monitorfilter:
Compiled OK.
 monitor: loading
in chain (17):
        0: -7fffffff (0000000000000000) (00000000) SecureXL inbound (sxl_in)
        1: -7ffffffe (0000000000000000) (00000000) SecureXL inbound CT (sxl_ct)
        2: -7f800001 (ffffffff8b6774d0) (ffffffff) fwmonitor (i/f side)
        3: -7f800000 (ffffffff8b6718c0) (ffffffff) IP Options Strip (in) (ipopt_strip)
        4: - 1fffff8 (ffffffff8b66f6f0) (00000001) Stateless verifications (in) (asm)
        5: - 1fffff7 (ffffffff8b66f210) (00000001) fw multik misc proto forwarding
        6:         0 (ffffffff8b8506a0) (00000001) fw VM inbound  (fw)
        7:         2 (ffffffff8b671d10) (00000001) fw SCV inbound (scv)
        8:         4 (ffffffff8b061ed0) (00000003) QoS inbound offload chain module
        9:         5 (ffffffff8b564d30) (00000003) fw offload inbound (offload_in)
        10:        10 (ffffffff8b842710) (00000001) fw post VM inbound  (post_vm)
        11:    100000 (ffffffff8b7fd6c0) (00000001) fw accounting inbound (acct)
        12:  22000000 (ffffffff8b0638d0) (00000003) QoS slowpath inbound chain mod (fg_sched)
        13:  70000000 (ffffffff8b6774d0) (ffffffff) fwmonitor (IP  side)
        14:  7f730000 (ffffffff8b3c40b0) (00000001) passive streaming (in) (pass_str)
        15:  7f750000 (ffffffff8b0e5b40) (00000001) TCP streaming (in) (cpas)
        16:  7f800000 (ffffffff8b671870) (ffffffff) IP Options Restore (in) (ipopt_res)
out chain (16):
        0: -7f800000 (ffffffff8b6718c0) (ffffffff) IP Options Strip (out) (ipopt_strip)
        1: -70000000 (ffffffff8b6774d0) (ffffffff) fwmonitor (i/f side)
        2: - 1ffffff0 (ffffffff8b0d0190) (00000001) TCP streaming (out) (cpas)
        3: - 1ffff50 (ffffffff8b3c40b0) (00000001) passive streaming (out) (pass_str)
        4: - 1f00000 (ffffffff8b66f6f0) (00000001) Stateless verifications (out) (asm)
        5: -     1ff (ffffffff8aeec0a0) (00000001) NAC Packet Outbound (nac_tag)
        6:         0 (ffffffff8b8506a0) (00000001) fw VM outbound (fw)
        7:        10 (ffffffff8b842710) (00000001) fw post VM outbound  (post_vm)
        8:  15000000 (ffffffff8b062540) (00000003) QoS outbound offload chain modul (fg_pol)
        9:  21000000 (ffffffff8b0638d0) (00000003) QoS slowpath outbound chain mod (fg_sched)
        10:  70000000 (ffffffff8b6774d0) (ffffffff) fwmonitor (IP side)
        11:  7f000000 (ffffffff8b7fd6c0) (00000001) fw accounting outbound (acct)
        12:  7f700000 (ffffffff8b0e4660) (00000001) TCP streaming post VM (cpas)
        13:  7f800000 (ffffffff8b671870) (ffffffff) IP Options Restore (out) (ipopt_res)
        14:  7f900000 (0000000000000000) (00000000) SecureXL outbound (sxl_out)
        15:  7fa00000 (0000000000000000) (00000000) SecureXL deliver (sxl_deliver)
 monitor: monitoring (control-C to stop)
[vs_0][fw_1] eth0:oq1 (TCP streaming (out))[1228]: 192.168.204.40 -> 192.168.204.1 (TCP)
len=1228 id=37575
TCP: 22 -> 51702 ...PA. seq=34e2af31 ack=e6c995ce
[vs_0][fw_1] eth0:OQ10 (TCP streaming post VM)[1228]: 192.168.204.40 -> 192.168.204.1 (TCP)
len=1228 id=37575
TCP: 22 -> 51702 ...PA. seq=34e2af31 ack=e6c995ce
[vs_0][fw_1] eth0:iq2 (IP Options Strip (in))[40]: 192.168.204.1 -> 192.168.204.40 (TCP)
len=40 id=32022
TCP: 51702 -> 22 ....A. seq=e6c995ce ack=34e2af31
[vs_0][fw_1] eth0:IQ13 (TCP streaming (in))[40]: 192.168.204.1 -> 192.168.204.40 (TCP) len=40
id=32022
TCP: 51702 -> 22 ....A. seq=e6c995ce ack=34e2af31
[vs_0][fw_1] eth0:oq1 (TCP streaming (out))[1356]: 192.168.204.40 -> 192.168.204.1 (TCP)
len=1356 id=37576
TCP: 22 -> 51702 ...PA. seq=34e2b3d5 ack=e6c995ce
[vs_0][fw_1] eth0:OQ10 (TCP streaming post VM)[1356]: 192.168.204.40 -> 192.168.204.1 (TCP)
len=1356 id=37576
TCP: 22 -> 51702 ...PA. seq=34e2b3d5 ack=e6c995ce
[vs_0][fw_1] eth0:iq2 (IP Options Strip (in))[40]: 192.168.204.1 -> 192.168.204.40 (TCP)
len=40 id=32023
TCP: 51702 -> 22 ....A. seq=e6c995ce ack=34e2b8f9
[vs_0][fw_1] eth0:IQ13 (TCP streaming (in))[40]: 192.168.204.1 -> 192.168.204.40 (TCP) len=40
id=32023
TCP: 51702 -> 22 ....A. seq=e6c995ce ack=34e2b8f9
[vs_0][fw_1] eth0:oq1 (TCP streaming (out))[1356]: 192.168.204.40 -> 192.168.204.1 (TCP)
len=1356 id=37577
TCP: 22 -> 51702 ...PA. seq=34e2b8f9 ack=e6c995ce
[vs_0][fw_1] eth0:OQ10 (TCP streaming post VM)[1356]: 192.168.204.40 -> 192.168.204.1 (TCP)
len=1356 id=37577
TCP: 22 -> 51702 ...PA. seq=34e2b8f9 ack=e6c995ce
[vs_0][fw_1] eth0:oq1 (TCP streaming (out))[412]: 192.168.204.40 -> 192.168.204.1 (TCP)
len=412 id=37578
TCP: 22 -> 51702 ...PA. seq=34e2be1d ack=e6c995ce
[vs_0][fw_1] eth0:OQ10 (TCP streaming post VM)[412]: 192.168.204.40 -> 192.168.204.1 (TCP)
len=412 id=37578
TCP: 22 -> 51702 ...PA. seq=34e2be1d ack=e6c995ce
[vs_0][fw_1] eth0:iq2 (IP Options Strip (in))[40]: 192.168.204.1 -> 192.168.204.40 (TCP)
len=40 id=32024
TCP: 51702 -> 22 ....A. seq=e6c995ce ack=34e2bf91
[vs_0][fw_1] eth0:IQ13 (TCP streaming (in))[40]: 192.168.204.1 -> 192.168.204.40 (TCP) len=40
id=32024
TCP: 51702 -> 22 ....A. seq=e6c995ce ack=34e2bf91
[vs_0][fw_1] eth0:oq1 (TCP streaming (out))[716]: 192.168.204.40 -> 192.168.204.1 (TCP)
len=716 id=37579
TCP: 22 -> 51702 ...PA. seq=34e2bf91 ack=e6c995ce
[vs_0][fw_1] eth0:OQ10 (TCP streaming post VM)[716]: 192.168.204.40 -> 192.168.204.1 (TCP)
len=716 id=37579
TCP: 22 -> 51702 ...PA. seq=34e2bf91 ack=e6c995ce
 monitor: unloading
Read 3 inbound packets and 5 outbound packets
[Expert@MyGW:0]#
```

Example 5 - Showing list of Chain Modules with the FW Monitor, when you do not change the default capture positions

```
[Expert@MyGW:0]# fw ctl chain
in chain (17):
        0: -7fffffff (0000000000000000) (00000000) SecureXL inbound (sxl_in)
        1: -7ffffffe (0000000000000000) (00000000) SecureXL inbound CT (sxl_ct)
        2: -7f800000 (ffffffff8b6718c0) (ffffffff) IP Options Strip (in) (ipopt_strip)
        3: -70000000 (ffffffff8b6774d0) (ffffffff) fwmonitor (i/f side)
        4: - 1fffff8 (ffffffff8b66f6f0) (00000001) Stateless verifications (in) (asm)
        5: - 1fffff7 (ffffffff8b66f210) (00000001) fw multik misc proto forwarding
        6:         0 (ffffffff8b8506a0) (00000001) fw VM inbound  (fw)
        7:         2 (ffffffff8b671d10) (00000001) fw SCV inbound (scv)
        8:         4 (ffffffff8b061ed0) (00000003) QoS inbound offload chain module
        9:         5 (ffffffff8b564d30) (00000003) fw offload inbound (offload_in)
        10:        10 (ffffffff8b842710) (00000001) fw post VM inbound  (post_vm)
        11:    100000 (ffffffff8b7fd6c0) (00000001) fw accounting inbound (acct)
        12:  22000000 (ffffffff8b0638d0) (00000003) QoS slowpath inbound chain mod (fg_sched)
        13:  70000000 (ffffffff8b6774d0) (ffffffff) fwmonitor (IP  side)
        14:  7f730000 (ffffffff8b3c40b0) (00000001) passive streaming (in) (pass_str)
        15:  7f750000 (ffffffff8b0e5b40) (00000001) TCP streaming (in) (cpas)
        16:  7f800000 (ffffffff8b671870) (ffffffff) IP Options Restore (in) (ipopt_res)
out chain (16):
        0: -7f800000 (ffffffff8b6718c0) (ffffffff) IP Options Strip (out) (ipopt_strip)
        1: -70000000 (ffffffff8b6774d0) (ffffffff) fwmonitor (i/f side)
        2: - 1fffff0 (ffffffff8b0d0190) (00000001) TCP streaming (out) (cpas)
        3: - 1ffff50 (ffffffff8b3c40b0) (00000001) passive streaming (out) (pass_str)
        4: - 1f00000 (ffffffff8b66f6f0) (00000001) Stateless verifications (out) (asm)
        5: -     1ff (ffffffff8aeec0a0) (00000001) NAC Packet Outbound (nac_tag)
        6:         0 (ffffffff8b8506a0) (00000001) fw VM outbound (fw)
        7:        10 (ffffffff8b842710) (00000001) fw post VM outbound  (post_vm)
        8:  15000000 (ffffffff8b062540) (00000003) QoS outbound offload chain modul (fg_pol)
        9:  21000000 (ffffffff8b0638d0) (00000003) QoS slowpath outbound chain mod (fg_sched)
        10:  70000000 (ffffffff8b6774d0) (ffffffff) fwmonitor (IP side)
        11:  7f000000 (ffffffff8b7fd6c0) (00000001) fw accounting outbound (acct)
        12:  7f700000 (ffffffff8b0e4660) (00000001) TCP streaming post VM (cpas)
        13:  7f800000 (ffffffff8b671870) (ffffffff) IP Options Restore (out) (ipopt_res)
        14:  7f900000 (0000000000000000) (00000000) SecureXL outbound (sxl_out)
        15:  7fa00000 (0000000000000000) (00000000) SecureXL deliver (sxl_deliver)
[Expert@MyGW:0]#
```

Examples for the "-e" parameter

Example 1 - Capture everything

```
[Expert@HostName]# fw monitor -e "accept;" -o /var/log/fw_
mon.cap
```

Example 2 - Capture traffic to / from specific hosts

To specify a host, you can use one of these expressions:

- Use `"host(`*`<IP_Address_in_Doted_Decimal_format>`*`)"`, which applies to both Source IP address and Destination IP address
- Use a specific Source IP address `"src=`*`<IP_Address_in_Doted_Decimal_format>`*`"` and a specific Destination IP address `"dst=`*`<IP_Address_in_Doted_Decimal_format>`*`"`

Example filters:

- Capture everything between host X and host Y:

```
[Expert@HostName]# fw monitor -e "host(x.x.x.x) and host
(y.y.y.y), accept;" -o /var/log/fw_mon.cap
```

```
[Expert@HostName]# fw monitor -e "((src=x.x.x.x ,
dst=y.y.y.y) or (src=y.y.y.y , dst=x.x.x.x)), accept;" -o
/var/log/fw_mon.cap
```

- Capture everything between hosts X,Z and hosts Y,Z in *all* Firewall kernel chains:

```
[Expert@HostName]# fw monitor -p all -e "((src=x.x.x.x or
dst=z.z.z.z) and (src=y.y.y.y or dst=z.z.z.z)), accept ;" -o
/var/log/fw_mon.cap
```

- Capture everything to/from host X or to/from host Y or to/from host Z:

```
[Expert@HostName]# fw monitor -e "host(x.x.x.x) or host
(y.y.y.y) or host(z.z.z.z), accept;" -o /var/log/fw_mon.cap
```

```
[Expert@HostName]# fw monitor -e "((src=x.x.x.x or
dst=x.x.x.x) or (src=y.y.y.y or dst=y.y.y.y) or (src=z.z.z.z
or dst=z.z.z.z)), accept;" -o /var/log/fw_mon.cap
```

Example 3 - Capture traffic to / from specific ports

Note - You must specify port numbers in Decimal format. Refer to the `/etc/services` file on the Security Gateway, or to *IANA Service Name and Port Number Registry*.

To specify a port, you can use one of these expressions:

- Use "`port(<IANA_Port_Number>)`", which applies to both Source Port and Destination Port
- Use a specific Source Port "`sport=<IANA_Port_Number>`" and a specific Destination Port "`dport=<IANA_Port_Number>`"
- In addition:
  - For specific TCP port, you can use "`tcpport(<IANA_Port_Number>)`", which applies to both Source TCP Port and Destination TCP Port
  - For specific UDP port, you can use "`udpport(<IANA_Port_Number>)`", which applies to both Source UDP Port and Destination UDP Port

Example filters:

- Capture everything to/from port X:

```
[Expert@HostName]# fw monitor -e "port(x), accept;" -o
/var/log/fw_mon.cap
```

```
[Expert@HostName]# fw monitor -e "(sport=x or dport=x),
accept;" -o /var/log/fw_mon.cap
```

- Capture everything except port X:

```
[Expert@HostName]# fw monitor -e "((sport=!x) or
(dport=!x)), accept;" -o /var/log/fw_mon.cap
```

```
[Expert@HostName]# fw monitor -e "not (sport=x or dport=x),
accept;" -o /var/log/fw_mon.cap
```

- Capture everything except SSH:

```
[Expert@HostName]# fw monitor -e "((sport!=22) or
(dport!=22)), accept;" -o /var/log/fw_mon.cap
```

```
[Expert@HostName]# fw monitor -e "not (sport=22 or
dport=22), accept;" -o /var/log/fw_mon.cap
```

```
[Expert@HostName]# fw monitor -e "not tcpport(22), accept;"
-o /var/log/fw_mon.cap
```

- Capture everything to/from host X except SSH:

```
[Expert@HostName]# fw monitor -e "(host(x.x.x.x) and
(sport!=22 or dport!=22)), accept;" -o /var/log/fw_mon.cap
```

```
[Expert@HostName]# fw monitor -e "((src=x.x.x.x or
dst=x.x.x.x) and (not (sport=22 or dport=22))), accept;" -o
/var/log/fw_mon.cap
```

```
[Expert@HostName]# fw monitor -e "(host(x.x.x.x) and not
tcpport(22)), accept;" -o /var/log/fw_mon.cap
```

- Capture everything except NTP:

```
[Expert@HostName]# fw monitor -e "not udpport(123), accept;"
-o /var/log/fw_mon.cap
```

Example 4 - Capture traffic over specific protocol

Note - You must specify protocol numbers in Decimal format. Refer to the `/etc/protocols` file on the Security Gateway, or to *IANA Protocol Numbers*.

To specify a protocol, you can use one of these expressions:

- Use `"ip_p=<IANA_Protocol_Number>"`

  Examples:

  - To specify TCP protocol with byte offset, use `"ip_p=6"`
  - To specify UDP protocol with byte offset, use `"ip_p=11"`
  - To specify ICMP protocol with byte offset, use `"ip_p=1"`

- Use `"accept [9:1]=<IANA_Protocol_Number>"`

  Examples:

  - To specify TCP protocol with byte offset, use `"accept [9:1]=6"`
  - To specify UDP protocol with byte offset, use `"accept [9:1]=11"`
  - To specify ICMP protocol with byte offset, use `"accept [9:1]=1"`

- In addition, you can explicitly use these expressions to specify protocols:

  Summary Table

Which protocol to specify	On which port(s) traffic is captured	Expression
TCP	N / A	`"tcp, accept;"`
UDP	N / A	`"udp, accept;"`

ICMPv4	N / A	`"icmp, accept;"`
		or
		`"icmp4, accept;"`

ICMPv6	N / A	`"icmp6, accept;"`
HTTP	TCP 80	`"http, accept;"`
HTTPS	TCP 443	`"https, accept;"`
PROXY	TCP 8080	`"proxy, accept;"`
DNS	UDP 53	`"dns, accept;"`
IKE	UDP 500	`"ike, accept;"`
NAT-T	UDP 4500	`"natt, accept;"`
ESP and IKE	IP proto 50 and UDP 500	`"vpn, accept;"`

Which protocol to specify	On which port(s) traffic is captured	Expression
All VPN-related data: a. ESP b. IPsec over UDP c. IKE d. NAT-T e. CRL f. RDP g. Tunnel Test h. Topology i. L2TP j. SCV k. Multi-Portal l. and so on	a. IP proto 50 b. UDP 2746 c. UDP 500 d. UDP 4500 e. TCP 18264 f. UDP 259 g. UDP 18234 h. TCP 264 i. TCP 1701 j. UDP 18233 k. TCP 443 + TCP 444 l. and so on	`"vpnall, accept;"`
Multi-Portal connections	TCP 443 and TCP 444	`"multi, accept;"`
SSH	TCP 22	`"ssh, accept;"`
FTP	TCP 20 and TCP 21	`"ftp, accept;"`
Telnet	TCP 23	`"telnet, accept;"`
SMTP	TCP 25	`"smtp, accept;"`
POP3	TCP 110	`"pop3, accept;"`

Example filters:

- Filter to capture everything on protocol X:

```
[Expert@HostName]# fw monitor -e "ip_p=X, accept;" -o
/var/log/fw_mon.cap
```

- Filter to capture rverything on protocol X and port Z on protocol Y:

```
[Expert@HostName]# fw monitor -e "(ip_p=X) or (ip_p=Y, port
(Z)), accept;" -o /var/log/fw_mon.cap
```

- Filter to capture capture everything TCP between host X and host Y:

```
[Expert@HostName]# fw monitor -e "ip_p=6, host(x.x.x.x) or
host(y.y.y.y), accept;" -o /var/log/fw_mon.cap
```

```
[Expert@HostName]# fw monitor -e "tcp, host(x.x.x.x) or host
(y.y.y.y), accept;" -o /var/log/fw_mon.cap
```

```
[Expert@HostName]# fw monitor -e "accept [9:1]=6 ,
((src=x.x.x.x , dst=y.y.y.y) or (src=y.y.y.y ,
dst=x.x.x.x));"
```

```
[Expert@HostName]# fw monitor -e "ip_p=6, ((src=x.x.x.x ,
dst=y.y.y.y) or (src=y.y.y.y , dst=x.x.x.x)), accept;" -o
/var/log/fw_mon.cap
```

Example 5 - Capture traffic with specific protocol options

Note - Refer to the `$FWDIR/lib/tcpip.def` file on Security Gateway.

Summary Table for IPv4

Option Description	Expression	Example
Source IPv4 address of the IPv4 packet	`ip_src = <IPv4_Address>`	`fw monitor -e "ip_src = 192.168.22.33, accept;"`
Destination IPv4 address of the IPv4 packet	`ip_dst = <IPv4_Address>`	`fw monitor -e "ip_dst = 192.168.22.33, accept;"`
Time To Live of the IPv4 packet	`ip_ttl = <Number>`	`fw monitor -e "ip_ttl = 255, accept;"`
Total Length of the IPv4 packet in bytes	`ip_len = <Length_in_Bytes>`	`fw monitor -e "ip_len = 64, accept;"`

Option Description	Expression	Example
TOS field of the IPv4 packet	`ip_tos = <Number>`	`fw monitor -e "ip_tos = 0, accept;"`

IANA Protocol Number (either in Dec or in Hex) encapsulated in the IPv4 packet	`ip_p = <IANA_Protocol_Number>`	Example for TCP:
		`fw monitor -e "ip_p = 6, accept;"`
		Examples for UDP:
		`fw monitor -e "ip_p = 17, accept;"`
		`fw monitor -e "ip_p = 0x11, accept;"`
		Example for ICMPv4:
		`fw monitor -e "ip_p = 1, accept;"`

Summary Table for IPv6

Option Description	Expression	Example
Source IPv6 address of the IPv6 packet	ip_src6p = <IPv6_Address>	`fw monitor -e "ip_src6p = 0:0:0:0:0:ffff:c0a8:1621, accept;"`
Destination IPv6 address of the IPv6 packet	ip_dst6p = <IPv6_Address>	`fw monitor -e "ip_dst6p = 0:0:0:0:0:ffff:c0a8:1621, accept;"`
Payload Length of the IPv6 packet in bytes	ip_len6 = <Length_in_Bytes>	`fw monitor -e "ip_len6 = 1000, accept;"`
Hop Limit ("Time To Live") of the IPv6 packet	ip_ttl6 = <Number>	`fw monitor -e "ip_ttl6 = 255, accept;"`
Next Header of the IPv6 packet - encapsulated IANA Protocol Number	ip_p6 = <IANA_Protocol_Number>	`fw monitor -e "ip_p6 = 6, accept;"`

Summary Table for TCP

Option Description	Expression	Example
SYN flag is set in TCP packet	`syn`	`fw monitor -e "ip_p = 6, syn, accept;"`
ACK flag is set in TCP packet	`ack`	`fw monitor -e "ip_p = 6, ack, accept;"`
RST flag is set in TCP packet	`rst`	`fw monitor -e "ip_p = 6, rst, accept;"`
FIN flag is set in TCP packet	`fin`	`fw monitor -e "ip_p = 6, fin, accept;"`
First packet of TCP connection (SYN flag is set, but ACK flag is not set in TCP packet)	`first`	`fw monitor -e "ip_p = 6, first, accept;"`
Not the first packet of TCP connection (SYN flag is not set in TCP packet)	`not_first`	`fw monitor -e "ip_p = 6, not_first, accept;"`
Established TCP connection (either ACK flag is set, or SYN flag is not set in TCP packet)	`established`	`fw monitor -e "ip_p = 6, established, accept;"`
Last packet of TCP connection (both ACK flag and FIN flag are set in TCP packet)	`last`	`fw monitor -e "ip_p = 6, last, accept;"`
End of TCP connection (either RST flag is set, or FIN flag is set in TCP packet)	`tcpdone`	`fw monitor -e "ip_p = 6, tcpdone, accept;"`

Option Description	Expression	Example	
		TCP Flag	Example
General way to match the flags inside in TCP packets	`th_flags = <Sum_of_Flags_Hex_Values>`	SYN (0x2)	`fw monitor -e "th_flags = 0x2, accept;"`
		ACK (0x10)	`fw monitor -e "th_flags = 0x10, accept;"`
		PSH (0x8)	`fw monitor -e "th_flags = 0x8, accept;"`
		FIN (0x1)	`fw monitor -e "th_flags = 0x1, accept;"`
		RST (0x4)	`fw monitor -e "th_flags = 0x4, accept;"`
		URG (0x20)	`fw monitor -e "th_flags = 0x20, accept;"`

<table>
<tr><th>Option Description</th><th>Expression</th><th>Example</th></tr>
<tr><td></td><td></td><td><table>
<tr><th>TCP Flag</th><th>Example</th></tr>
<tr><td>SYN + ACK</td><td><code>fw monitor -e "th_flags = 0x12, accept;"</code></td></tr>
<tr><td>PSH + ACK</td><td><code>fw monitor -e "th_flags = 0x18, accept;"</code></td></tr>
<tr><td>FIN + ACK</td><td><code>fw monitor -e "th_flags = 0x11, accept;"</code></td></tr>
<tr><td>RST + ACK</td><td><code>fw monitor -e "th_flags = 0x14, accept;"</code></td></tr>
</table></td></tr>
<tr><td>TCP source port</td><td><code>th_sport = &lt;<i>Port_Number</i>&gt;</code></td><td><code>fw monitor -e "th_sport = 59259, accept;"</code></td></tr>
<tr><td>TCP destination port</td><td><code>th_dport = &lt;<i>Port_Number</i>&gt;</code></td><td><code>fw monitor -e "th_dport = 22, accept;"</code></td></tr>
<tr><td>TCP sequence number (either in Dec or in Hex)</td><td><code>th_seq = &lt;<i>Number</i>&gt;</code></td><td>Example for Dec format:
<code>fw monitor -e "th_seq = 3937833514, accept;"</code>
Example for Hex format:
<code>fw monitor -e "th_seq = 0xeab6922a, accept;"</code></td></tr>
</table>

Option Description	Expression	Example

TCP acknowledged number (either in Dec or in Hex)	`th_ack = <Number>`	Example for Dec format:
		`fw monitor -e "th_ack = 509054325, accept;"`
		Example for Hex format:
		`fw monitor -e "th_ack = 0x1e578d75, accept;"`

Summary Table for UDP

Option Description	Expression	Example
UDP source port	`uh_sport = <Port_Number>`	`fw monitor -e "uh_sport = 53, accept;"`
UDP destination port	`uh_dport = <Port_Number>`	`fw monitor -e "uh_dport = 53, accept;"`

Summary Table for ICMPv4

Option Description	Expression	Example
ICMPv4 packets with specified Type	`icmp_type = <Number>`	`fw monitor -e "icmp_type = 0, accept;"`
ICMPv4 packets with specified Code	`icmp_code = <Number>`	`fw monitor -e "icmp_code = 0, accept;"`
ICMPv4 packets with specified Identifier	`icmp_id = <Number>`	`fw monitor -e "icmp_id = 20583, accept;"`
ICMPv4 packets with specified Sequence number	`icmp_seq = <Number>`	`fw monitor -e "icmp_seq = 1, accept;"`
ICMPv4 Echo Request packets (Type 8, Code 0)	`echo_req`	`fw monitor -e "echo_req, accept;"`
ICMPv4 Echo Reply packets (Type 0, Code 0)	`echo_reply`	`fw monitor -e "echo_reply, accept;"`
ICMPv4 Echo Request and ICMPv4 Echo Reply packets	`ping`	`fw monitor -e "ping, accept;"`

Option Description	Expression	Example
Traceroute packets as implemented in Unix OS (UDP packets on ports above 30000 and with TTL<30; or ICMP Time exceeded packets)	`traceroute`	`fw monitor -e "traceroute, accept;"`
Traceroute packets as implemented in Windows OS (ICMP Request packets with TTL<30; or ICMP Time exceeded packets)	`tracert`	`fw monitor -e "tracert, accept;"`
Length of ICMPv4 packets	`icmp_ip_len = <length>`	`fw monitor -e "icmp_ip_len = 84, accept;"`

Summary Table for ICMPv6

Option Description	Expression	Example
ICMPv6 packets with specified Type	`icmp6_type = <Number>`	`fw monitor -e "icmp6_type = 1, accept;"`
ICMPv6 packets with specified Code	`icmp6_code = <Number>`	`fw monitor -e "icmp6_code = 3, accept;"`

Example 6 - Capture specific bytes in packets

Syntax:

```
fw monitor -e "accept [ <Offset> : <Length> , <Byte Order> ]
<Relational-Operator> <Value>;"
```

Parameters:

Parameter	Explanation
`<Offset>`	Specifies the offset relative to the beginning of the IP packet from where the value should be read.

Parameter	Explanation
<Length>	Specifies the number of bytes: - `1` = byte - `2` = word - `4` = dword If length is not specified, FW Monitor assumes 4 (dword).
<Byte Order>	Specifies the byte order: - `b` = big endian, or network order - `l` = little endian, or host order If order is not specified, FW Monitor assumes little endian byte order.
<Relational-Operator	Relational operator to express the relation between the packet data and the value: - `<` - less than - `>` - greater than - `<=` - less than or equal to - `>=` - greater than - `=` or `is` - equal to - `!=` or `is not` - not equal to
<Value>	One of the data types known to INSPECT (for example, an IP address, or an integer).

Explanations:

- The IP-based protocols are stored in the IP packet as a byte at offset 9.
    - To filter based on a Protocol encapsulated into IP, use this syntax:

      ```
      [Expert@HostName]# fw monitor -e "accept [9:1]=<IANA_
      Protocol_Number>;"
      ```

- The Layer 3 IP Addresses are stored in the IP packet as double words at offset 12 (Source address) and at offset 16 (Destination address).

  - To filter based on a Source IP address, use this syntax:

```
[Expert@HostName]# fw monitor -e "accept [12:4,b]=<IP_
Address_in_Doted_Decimal_format>;"
```

  - To filter based on a Destination IP address, use this syntax:

```
[Expert@HostName]# fw monitor -e "accept [16:4,b]=<IP_
Address_in_Doted_Decimal_format>;"
```

- The Layer 4 Ports are stored in the IP packet as a word at offset 20 (Source port) and at offset 22 (Destination port).

  - To filter based on a Source port, use this syntax:

```
[Expert@HostName]# fw monitor -e "accept
[20:2,b]=<Port_Number_in_Decimal_format>;"
```

  - To filter based on a Destination port, use this syntax:

```
[Expert@HostName]# fw monitor -e "accept
[22:2,b]=<Port_Number_in_Decimal_format>;"
```

Example filters:

- Capture everything between host X and host Y:

```
[Expert@HostName]# fw monitor -e "accept (([12:4,b]=x.x.x.x
, [16:4,b]=y.y.y.y) or ([12:4,b]=y.y.y.y ,
[16:4,b]=x.x.x.x));"
```

- Capture everything on port X:

```
[Expert@HostName]# fw monitor -e "accept [20:2,b]=x or
[22:2,b]=x;" -o /var/log/fw_mon.cap
```

Example 7 - Capture traffic to/from specific network

You must specify the *network address* and *length of network mask* (number of bits).

There are 3 options:

Traffic direction	Expression
To or From a network	`"net(<Network_IP_Address>, <Mask_Length>), accept;"`

Traffic direction	Expression
To a network	`"to_net(<Network_IP_Address>, <Mask_Length>), accept;"`
From a network	`"from_net(<Network_IP_Address>, <Mask_Length>), accept;"`

Example filters:

- Capture everything to/from network 192.168.33.0 / 24:

  ```
  [Expert@HostName]# fw monitor -e "net(192.168.33.0, 24), accept;"
  ```

- Capture everything sent to network 192.168.33.0 / 24:

  ```
  [Expert@HostName]# fw monitor -e "to_net(192.168.33.0, 24), accept;"
  ```

- Capture everything sent from network 192.168.33.0 / 24:

  ```
  [Expert@HostName]# fw monitor -e "from_net(192.168.33.0, 24), accept;"
  ```

Example 8 - Filter out irrelevant "noise"

Filter in only TCP protocol, and HTTP and HTTPS ports

Filter out the SSH and FW Logs

```
[Expert@HostName]# fw monitor -e "accept (ip_p=6) and (not (sport=22 or dport=22)) and (not (sport=257 or dport=257)) and ((dport=80 or dport=443) or (sport=80 or sport=443);" -o /var/log/fw_mon.cap
```

Examples for the "-F" parameter

You can specify up to 5 capture filters with this parameter (up to 5 instances of the "`-F`" parameter in the syntax).

The FW Monitor performs the logical "OR" between all specified simple capture filters.

Value 0 is used as "`any`".

Example 1 - Capture everything

```
[Expert@HostName]# fw monitor -F "0,0,0,0,0" -o /var/log/fw_
mon.cap
```

Example 2 - Capture traffic to / from specific hosts

- Capture all traffic from Source IP x.x.x.x (any port) to Destination IP y.y.y.y (any port), over all protocols:

  ```
  [Expert@HostName]# fw monitor -F "x.x.x.x,0,y.y.y.y,0,0" -o
  /var/log/fw_mon.cap
  ```

- Capture all traffic between Host x.x.x.x (any port) and Host y.y.y.y (any port), over all protocols:

  ```
  [Expert@HostName]# fw monitor -F "x.x.x.x,0,y.y.y.y,0,0" -F
  "y.y.y.y,0, x.x.x.x ,0,0" -o /var/log/fw_mon.cap
  ```

Example 3 - Capture traffic to / from specific ports

- Capture traffic from any Source IP from Source Port X to any Destination IP to Destination Port Y, over all protocols:

  ```
  [Expert@HostName]# fw monitor -F "0,x,0,y,0" -o /var/log/fw_
  mon.cap
  ```

- Capture traffic between all hosts, between Port X and Port Y, over all protocols:

  ```
  [Expert@HostName]# fw monitor -F "0,x,0,y,0" -F "0,y,0,x,0"
  -o /var/log/fw_mon.cap
  ```

Example 4 - Capture traffic over specific protocol

- Capture traffic between all hosts, between all ports, over a Protocol with assigned number X:

  ```
  [Expert@HostName]# fw monitor -F "0,0,0,0,x" -o /var/log/fw_
  mon.cap
  ```

Example 5 - Capture traffic between specific hosts between specific ports over specific protocol

```
[Expert@HostName]# fw monitor -F "a.a.a.a,b,c.c.c.c,d,e" -F
"c.c.c.c,d,a.a.a.a,b,e" -o /var/log/fw_mon.cap
```

To capture only HTTP traffic between the Client 1.1.1.1 and the Server 2.2.2.2:

```
fw montior –F "1.1.1.1,0,2.2.2.2,80,6" –F
"2.2.2.2,80,1.1.1.1,0,6" -o /var/log/fw_mon.cap
```

fw sam_policy

Description

Manages the Suspicious Activity Policy editor that works with these types of rules:

- Suspicious Activity Monitoring (SAM) rules.

  See sk112061: How to create and view Suspicious Activity Monitoring (SAM) Rules.

- Rate Limiting rules.

  See sk112454: How to configure Rate Limiting rules for DoS Mitigation.

Also, see these commands:

- `sam_alert` (see the *R81 CLI Reference Guide*)

Notes:

- These commands are interchangeable:
  - For IPv4: "`fw sam_policy`" and "`fw samp`".
  - For IPv6: "`fw6 sam_policy`" and "`fw6 samp`".
- You can run these commands in Gaia Clish, or Expert mode.
- Security Gateway stores the SAM Policy rules in the `$FWDIR/database/sam_policy.db` file.
- Security Gateway stores the SAM Policy management settings in the `$FWDIR/database/sam_policy.mng` file.

Important:

- Configuration you make with these commands, survives reboot.
- VSX mode does **not** support Suspicious Activity Policy configured in SmartView Monitor. See sk79700.
- In VSX mode, you must go to the context of an applicable Virtual System.
  - In Gaia Clish, run: `set virtual-system <VSID>`
  - In the Expert mode, run: `vsenv <VSID>`
- In a Cluster, you must configure all the Cluster Members in the same way.

Best Practice - The SAM Policy rules consume some CPU resources on Security Gateway. Set an expiration for rules that gives you time to investigate, but does not affect performance. Keep only the required SAM Policy rules. If you confirm that an activity is risky, edit the Security Policy, educate users, or otherwise handle the risk.

Syntax for IPv4

```
fw [-d] sam_policy
      add <options>
      batch
      del <options>
      get <options>
```

```
fw [-d] samp
      add <options>
      batch
      del <options>
      get <options>
```

Syntax for IPv6

```
fw6 [-d] sam_policy
      add <options>
      batch
      del <options>
      get <options>
```

```
fw6 [-d] samp
      add <options>
      batch
      del <options>
      get <options>
```

Parameters

Parameter	Description
`-d`	Runs the command in debug mode. Use only if you troubleshoot the command itself. ★ **Best Practice** - If you use this parameter, then redirect the output to a file, or use the `script` command to save the entire CLI session.
`add <options>`	Adds one Rate Limiting rule one at a time. See *"fw sam_policy add" on page 187*.
`batch`	Adds or deletes many Rate Limiting rules at a time. See *"fw sam_policy batch" on page 200*.
`del <options>`	Deletes one configured Rate Limiting rule one at a time. See *"fw sam_policy del" on page 202*.
`get <options>`	Shows all the configured Rate Limiting rules. See *"fw sam_policy get" on page 206*.

fw sam_policy add

Description

The "*fw sam_policy add*" and "*fw6 sam_policy add*" commands:

- Add one Suspicious Activity Monitoring (SAM) rule at a time.
- Add one Rate Limiting rule at a time.

Notes:

- These commands are interchangeable:
  - For IPv4: "`fw sam_policy`" and "`fw samp`".
  - For IPv6: "`fw6 sam_policy`" and "`fw6 samp`".
- You can run these commands in Gaia Clish, or Expert mode.
- Security Gateway stores the SAM Policy rules in the `$FWDIR/database/sam_policy.db` file.
- Security Gateway stores the SAM Policy management settings in the `$FWDIR/database/sam_policy.mng` file.

Important:

- Configuration you make with these commands, survives reboot.
- VSX mode does **not** support Suspicious Activity Policy configured in SmartView Monitor. See sk79700.
- In VSX mode, you must go to the context of an applicable Virtual System.
  - In Gaia Clish, run: `set virtual-system <VSID>`
  - In the Expert mode, run: `vsenv <VSID>`
- In a Cluster, you must configure all the Cluster Members in the same way.

Best Practice - The SAM Policy rules consume some CPU resources on Security Gateway. Set an expiration for rules that gives you time to investigate, but does not affect performance. Keep only the required SAM Policy rules. If you confirm that an activity is risky, edit the Security Policy, educate users, or otherwise handle the risk.

Syntax to configure a Suspicious Activity Monitoring (SAM) rule for IPv4

```
fw [-d] sam_policy add [-u] -a {d|n|b} [-l {r|a}] [-t <Timeout>]
[-f <Target>] [-n <"Rule Name">] [-c <"Rule Comment">] [-o <"Rule
Originator">] [-z "<Zone>"] ip <IP Filter Arguments>
```

Syntax to configure a Suspicious Activity Monitoring (SAM) rule for IPv6

```
fw6 [-d] sam_policy add [-u] -a {d|n|b} [-l {r|a}] [-t <Timeout>]
[-f <Target>] [-n <"Rule Name">] [-c <"Rule Comment">] [-o <"Rule
Originator">] [-z "<Zone>"] ip <IP Filter Arguments>
```

Syntax to configure a Rate Limiting rule for IPv4

```
fw [-d] sam_policy add [-u] -a {d|n|b} [-l {r|a}] [-t <Timeout>]
[-f <Target>] [-n <"Rule Name">] [-c <"Rule Comment">] [-o <"Rule
Originator">] [-z "<Zone>"] quota <Quota Filter Arguments>
```

Syntax to configure a Rate Limiting rule for IPv6

```
fw6 [-d] sam_policy add [-u] -a {d|n|b} [-l {r|a}] [-t <Timeout>]
[-f <Target>] [-n <"Rule Name">] [-c <"Rule Comment">] [-o <"Rule
Originator">] [-z "<Zone>"] quota <Quota Filter Arguments
```

Parameters

Parameter	Description
`-d`	Runs the command in debug mode. Use only if you troubleshoot the command itself. **Best Practice** - If you use this parameter, then redirect the output to a file, or use the `script` command to save the entire CLI session.
`-u`	Optional. Specifies that the rule category is `User-defined`. Default rule category is `Auto`.
`-a {d | n | b}`	Mandatory. Specifies the rule action if the traffic matches the rule conditions: ■ `d` - Drop the connection. ■ `n` - Notify (generate a log) about the connection and let it through. ■ `b` - Bypass the connection - let it through without checking it against the policy rules. **Note** - Rules with action set to *Bypass* cannot have a log or limit specification. Bypassed packets and connections do not count towards overall number of packets and connection for limit enforcement of type ratio.
`-l {r | a}`	Optional. Specifies which type of log to generate for this rule for all traffic that matches: ■ `-r` - Generate a regular log ■ `-a` - Generate an alert log

Parameter	Description
`-t <Timeout>`	Optional. Specifies the time period (in seconds), during which the rule will be enforced. Default timeout is indefinite.
`-f <Target>`	Optional. Specifies the target Security Gateways, on which to enforce the Rate Limiting rule. `<Target>` can be one of these: ■ `all` - This is the default option. Specifies that the rule should be enforced on all managed Security Gateways. ■ Name of the Security Gateway or Cluster object - Specifies that the rule should be enforced only on this Security Gateway or Cluster object (the object name must be as defined in the SmartConsole). ■ Name of the Group object - Specifies that the rule should be enforced on all Security Gateways that are members of this Group object (the object name must be as defined in the SmartConsole).
`-n "<Rule Name>"`	Optional. Specifies the name (label) for this rule. **Notes:** ■ You must enclose this string in double quotes. ■ The length of this string is limited to 128 characters. ■ Before each space or a backslash character in this string, you must write a backslash (\) character. Example: `"This\ is\ a\ rule\ name\ with\ a\ backslash\ \\"`
`-c "<Rule Comment>"`	Optional. Specifies the comment for this rule. **Notes:** ■ You must enclose this string in double quotes. ■ The length of this string is limited to 128 characters. ■ Before each space or a backslash character in this string, you must write a backslash (\) character. Example: `"This\ is\ a\ comment\ with\ a\ backslash\ \\"`

Parameter	Description
`-o "<Rule Originator>"`	Optional. Specifies the name of the originator for this rule. **Notes:** - You must enclose this string in double quotes. - The length of this string is limited to 128 characters. - Before each space or a backslash character in this string, you must write a backslash (\) character. Example: `"Created\ by\ John\ Doe"`
`-z "<Zone>"`	Optional. Specifies the name of the Security Zone for this rule. **Notes:** - You must enclose this string in double quotes. - The length of this string is limited to 128 characters.
`ip <IP Filter Arguments>`	Mandatory (use this `ip` parameter, or the `quota` parameter). Configures the *Suspicious Activity Monitoring (SAM)* rule. Specifies the IP Filter Arguments for the SAM rule (you must use at least one of these options): `[-C] [-s <Source IP>] [-m <Source Mask>] [-d <Destination IP>] [-M <Destination Mask>] [-p <Port>] [-r <Protocol>]` See the explanations below.

Parameter	Description
`quota <Quota Filter Arguments>`	Mandatory (use this `quota` parameter, or the `ip` parameter). Configures the *Rate Limiting* rule. Specifies the Quota Filter Arguments for the Rate Limiting rule (see the explanations below): ■ `[flush true]` ■ `[source-negated {true | false}] source <Source>` ■ `[destination-negated {true | false}] destination <Destination>` ■ `[service-negated {true | false}] service <Protocol and Port numbers>` ■ `[<Limit1 Name> <Limit1 Value>] [<Limit2 Name> <Limit2 Value>] ...[<LimitN Name> <LimitN Value>]` ■ `[track <Track>]` **Important:** ■ The Quota rules are not applied immediately to the Security Gateway. They are only registered in the Suspicious Activity Monitoring (SAM) policy database. To apply all the rules from the SAM policy database immediately, add "`flush true`" in the `fw samp add` command syntax. ■ Explanation: For new connections rate (and for any rate limiting in general), when a rule's limit is violated, the Security Gateway also drops all packets that match the rule. The Security Gateway computes new connection rates on a per-second basis. At the start of the 1-second timer, the Security Gateway allows all packets, including packets for existing connections. If, at some point, during that 1 second period, there are too many new connections, then the Security Gateway blocks all remaining packets for the remainder of that 1-second interval. At the start of the next 1-second interval, the counters are reset, and the process starts over - the Security Gateway allows packets to pass again up to the point, where the rule’s limit is violated.

Explanation for the *IP Filter Arguments* syntax for Suspicious Activity Monitoring (SAM) rules

Argument	Description
`-C`	Specifies that open connections should be closed.
`-s` *`<Source IP>`*	Specifies the Source IP address.
`-m` *`<Source Mask>`*	Specifies the Source subnet mask (in dotted decimal format - x.y.z.w).
`-d` *`<Destination IP>`*	Specifies the Destination IP address.
`-M` *`<Destination Mask>`*	Specifies the Destination subnet mask (in dotted decimal format - x.y.z.w).
`-p` *`<Port>`*	Specifies the port number (see *IANA Service Name and Port Number Registry*).
`-r` *`<Protocol>`*	Specifies the protocol number (see *IANA Protocol Numbers*).

Explanation for the *Quota Filter Arguments* syntax for Rate Limiting rules

Argument	Description
`flush true`	Specifies to compile and load the quota rule to the SecureXL immediately.
`[source-negated {true | false}] source <Source>`	Specifies the source type and its value: ■ `any` The rule is applied to packets sent from all sources. ■ `range:<IP Address>` or `range:<IP Address Start>-<IP Address End>` The rule is applied to packets sent from: • Specified IPv4 addresses (x.y.z.w) • Specified IPv6 addresses (xxxx:yyyy:...:zzzz) ■ `cidr:<IP Address>/<Prefix>` The rule is applied to packets sent from: • IPv4 address with Prefix from 0 to 32 • IPv6 address with Prefix from 0 to 128 ■ `cc:<Country Code>` The rule matches the country code to the source IP addresses assigned to this country, based on the Geo IP database. The two-letter codes are defined in ISO 3166-1 alpha-2. ■ `asn:<Autonomous System Number>` The rule matches the AS number of the organization to the source IP addresses that are assigned to this organization, based on the Geo IP database. The valid syntax is *ASnnnn*, where *nnnn* is a number unique to the specific organization. **Notes:** ■ Default is: `source-negated false` ■ The `source-negated true` processes all source types, *except* the specified type.

Argument	Description
`[destination-negated {true | false}] destination <Destination>`	Specifies the destination type and its value: - `any` The rule is applied to packets sent to all destinations. - `range:<IP Address>` or `range:<IP Address Start>-<IP Address End>` The rule is applied to packets sent to: • Specified IPv4 addresses (x.y.z.w) • Specified IPv6 addresses (xxxx:yyyy:...:zzzz) - `cidr:<IP Address>/<Prefix>` The rule is applied to packets sent to: • IPv4 address with Prefix from 0 to 32 • IPv6 address with Prefix from 0 to 128 - `cc:<Country Code>` The rule matches the country code to the destination IP addresses assigned to this country, based on the Geo IP database. The two-letter codes are defined in ISO 3166-1 alpha-2. - `asn:<Autonomous System Number>` The rule matches the AS number of the organization to the destination IP addresses that are assigned to this organization, based on the Geo IP database. The valid syntax is *ASnnnn*, where *nnnn* is a number unique to the specific organization. **Notes:** - Default is: `destination-negated false` - The `destination-negated true` will process all destination types except the specified type

Argument	Description
`[service-negated {true | false}] service <Protocol and Port numbers>`	Specifies the Protocol number (see *IANA Protocol Numbers*) and Port number (see *IANA Service Name and Port Number Registry*): - `<Protocol>` IP protocol number in the range 1-255 - `<Protocol Start>-<Protocol End>` Range of IP protocol numbers - `<Protocol>/<Port>` IP protocol number in the range 1-255 and TCP/UDP port number in the range 1-65535 - `<Protocol>/<Port Start>-<Port End>` IP protocol number and range of TCP/UDP port numbers from 1 to 65535 **Notes:** - Default is: `service-negated false` - The `service-negated true` will process all traffic except the traffic with the specified protocols and ports

Argument	Description
`[<Limit 1 Name> <Limit 1 Value>] [<Limit 2 Name> <Limit 2 Value>] ... [<Limit N Name> <Limit N Value>]`	Specifies quota limits and their values. **Note** - Separate multiple quota limits with spaces. ■ `concurrent-conns <Value>` Specifies the maximal number of concurrent active connections that match this rule. ■ `concurrent-conns-ratio <Value>` Specifies the maximal ratio of the *concurrent-conns* value to the total number of active connections through the Security Gateway, expressed in parts per 65536 (formula: `N / 65536`). ■ `pkt-rate <Value>` Specifies the maximum number of packets per second that match this rule. ■ `pkt-rate-ratio <Value>` Specifies the maximal ratio of the *pkt-rate* value to the rate of all connections through the Security Gateway, expressed in parts per 65536 (formula: `N / 65536`). ■ `byte-rate <Value>` Specifies the maximal total number of bytes per second in packets that match this rule. ■ `byte-rate-ratio <Value>` Specifies the maximal ratio of the *byte-rate* value to the bytes per second rate of all connections through the Security Gateway, expressed in parts per 65536 (formula: `N / 65536`). ■ `new-conn-rate <Value>` Specifies the maximal number of connections per second that match the rule. ■ `new-conn-rate-ratio <Value>` Specifies the maximal ratio of the *new-conn-rate* value to the rate of all connections per second through the Security Gateway, expressed in parts per 65536 (formula: `N / 65536`).

Argument	Description
`[track <Track>]`	Specifies the tracking option: ■ `source` Counts connections, packets, and bytes for specific source IP address, and not cumulatively for this rule. ■ `source-service` Counts connections, packets, and bytes for specific source IP address, and for specific IP protocol and destination port, and not cumulatively for this rule.

Examples

Example 1 - Rate Limiting rule with a range

```
fw sam_policy add -a d -l r -t 3600 quota service any source range:172.16.7.11-172.16.7.13 new-conn-rate 5 flush true
```

Explanations:

- This rule drops packets for all connections (`-a d`) that exceed the quota set by this rule, including packets for existing connections.
- This rule logs packets (`-l r`) that exceed the quota set by this rule.
- This rule will expire in 3600 seconds (`-t 3600`).
- This rule limits the rate of creation of new connections to 5 connections per second (`new-conn-rate 5`) for any traffic (`service any`) from the source IP addresses in the range 172.16.7.11 - 172.16.7.13 (`source range:172.16.7.11-172.16.7.13`).

  Note - The limit of the total number of log entries per second is configured with the *fwaccel dos config set -n <rate>* command.
- This rule will be compiled and loaded on the SecureXL, together with other rules in the Suspicious Activity Monitoring (SAM) policy database immediately, because this rule includes the "`flush true`" parameter.

Example 2 - Rate Limiting rule with a service specification

```
fw sam_policy add -a n -l r quota service 1,50-51,6/443,17/53 service-negated true source cc:QQ byte-rate 0
```

Explanations:

- This rule logs and lets through all packets (`-a n`) that exceed the quota set by this rule.
- This rule does not expire (the `timeout` parameter is not specified). To cancel it, you must delete it explicitly.
- This rule applies to all packets except (`service-negated true`) the packets with IP protocol number 1, 50-51, 6 port 443 and 17 port 53 (`service 1,50-51,6/443,17/53`).
- This rule applies to all packets from source IP addresses that are assigned to the country with specified country code (`cc:QQ`).
- This rule does not let any traffic through (`byte-rate 0`) except the packets with IP protocol number 1, 50-51, 6 port 443 and 17 port 53.
- This rule will not be compiled and installed on the SecureXL immediately, because it does not include the "`flush true`" parameter.

Example 3 - Rate Limiting rule with ASN

```
fw sam_policy -a d quota source asn:AS64500,cidr:[::FFFF:C0A8:1100]/120 service any pkt-rate 0
```

Explanations:

- This rule drops (`-a d`) all packets that match this rule.
- This rule does not expire (the `timeout` parameter is not specified). To cancel it, you must delete it explicitly.
- This rule applies to packets from the Autonomous System number 64500 (`asn:AS64500`).
- This rule applies to packets from source IPv6 addresses FFFF:C0A8:1100/120 (`cidr:[::FFFF:C0A8:1100]/120`).
- This rule applies to all traffic (`service any`).
- This rule does not let any traffic through (`pkt-rate 0`).
- This rule will not be compiled and installed on the SecureXL immediately, because it does not include the "`flush true`" parameter.

Example 4 - Rate Limiting rule with whitelist

```
fw sam_policy add -a b quota source range:172.16.8.17-172.16.9.121 service 6/80
```

Explanations:

- This rule bypasses (`-a b`) all packets that match this rule.

  Note - The Access Control Policy and other types of security policy rules still apply.
- This rule does not expire (the `timeout` parameter is not specified). To cancel it, you must delete it explicitly.
- This rule applies to packets from the source IP addresses in the range 172.16.8.17 - 172.16.9.121 (`range:172.16.8.17-172.16.9.121`).
- This rule applies to packets sent to TCP port 80 (`service 6/80`).
- This rule will not be compiled and installed on the SecureXL immediately, because it does not include the "`flush true`" parameter.

Example 5 - Rate Limiting rule with tracking

```
fw sam_policy add -a d quota service any source-negated true source cc:QQ concurrent-conns-ratio 655 track source
```

Explanations:

- This rule drops (`-a d`) all packets that match this rule.
- This rule does not log any packets (the `-l r` parameter is not specified).
- This rule does not expire (the `timeout` parameter is not specified). To cancel it, you must delete it explicitly.
- This rule applies to all traffic (`service any`).
- This rule applies to all sources except (`source-negated true`) the source IP addresses that are assigned to the country with specified country code (`cc:QQ`).
- This rule limits the maximal number of concurrent active connections to 655/65536=~1% (`concurrent-conns-ratio 655`) for any traffic (`service any`) except (`service-negated true`) the connections from the source IP addresses that are assigned to the country with specified country code (`cc:QQ`).
- This rule counts connections, packets, and bytes for traffic only from sources that match this rule, and not cumulatively for this rule.
- This rule will not be compiled and installed on the SecureXL immediately, because it does not include the "`flush true`" parameter.

fw sam_policy batch

Description

The "*fw sam_policy batch*" and "*fw6 sam_policy batch*" commands:

- Add and delete many Suspicious Activity Monitoring (SAM) rules at a time.
- Add and delete many Rate Limiting rules at a time.

Notes:

- These commands are interchangeable:
  - For IPv4: "`fw sam_policy`" and "`fw samp`".
  - For IPv6: "`fw6 sam_policy`" and "`fw6 samp`".
- You can run these commands in Gaia Clish, or Expert mode.
- Security Gateway stores the SAM Policy rules in the `$FWDIR/database/sam_policy.db` file.
- Security Gateway stores the SAM Policy management settings in the `$FWDIR/database/sam_policy.mng` file.

Important:

- Configuration you make with these commands, survives reboot.
- VSX mode does **not** support Suspicious Activity Policy configured in SmartView Monitor. See sk79700.
- In VSX mode, you must go to the context of an applicable Virtual System.
  - In Gaia Clish, run: `set virtual-system <VSID>`
  - In the Expert mode, run: `vsenv <VSID>`
- In a Cluster, you must configure all the Cluster Members in the same way.

Best Practice - The SAM Policy rules consume some CPU resources on Security Gateway. Set an expiration for rules that gives you time to investigate, but does not affect performance. Keep only the required SAM Policy rules. If you confirm that an activity is risky, edit the Security Policy, educate users, or otherwise handle the risk.

Procedure

1. **Start the batch mode**

   - For IPv4, run:

     ```
     fw sam_policy batch << EOF
     ```

   - For IPv6, run:

     ```
     fw6 sam_policy batch << EOF
     ```

2. **Enter the applicable commands**

    - Enter one "`add`" or "`del`" command on each line, on as many lines as necessary.

      Start each line with only "`add`" or "`del`" parameter (not with "`fw samp`").

    - Use the same set of parameters and values as described in these commands:

        - *"fw sam_policy add" on page 187*
        - *"fw sam_policy del" on page 202*

    - Terminate each line with a Return (ASCII 10 - Line Feed) character (press Enter).

3. **End the batch mode**

    Type `EOF` and press Enter.

Example of a Rate Limiting rule for IPv4

```
[Expert@HostName]# fw samp batch <<EOF

add -a d -l r -t 3600 -c "Limit\ conn\ rate\ to\ 5\ conn/sec from\ these\ sources" quota service
any source range:172.16.7.13-172.16.7.13 new-conn-rate 5

del <501f6ef0,00000000,cb38a8c0,0a0afffe>

add -a b quota source range:172.16.8.17-172.16.9.121 service 6/80

EOF
[Expert@HostName]#
```

fw sam_policy del

Description

The "*fw sam_policy del*" and "*fw6 sam_policy del*" commands:

- Delete one configured Suspicious Activity Monitoring (SAM) rule at a time.
- Delete one configured Rate Limiting rule at a time.

Notes:

- These commands are interchangeable:
  - For IPv4: "`fw sam_policy`" and "`fw samp`".
  - For IPv6: "`fw6 sam_policy`" and "`fw6 samp`".
- You can run these commands in Gaia Clish, or Expert mode.
- Security Gateway stores the SAM Policy rules in the `$FWDIR/database/sam_policy.db` file.
- Security Gateway stores the SAM Policy management settings in the `$FWDIR/database/sam_policy.mng` file.

Important:

- Configuration you make with these commands, survives reboot.
- VSX mode does **not** support Suspicious Activity Policy configured in SmartView Monitor. See sk79700.
- In VSX mode, you must go to the context of an applicable Virtual System.
  - In Gaia Clish, run: `set virtual-system <VSID>`
  - In the Expert mode, run: `vsenv <VSID>`
- In a Cluster, you must configure all the Cluster Members in the same way.

Best Practice - The SAM Policy rules consume some CPU resources on Security Gateway. Set an expiration for rules that gives you time to investigate, but does not affect performance. Keep only the required SAM Policy rules. If you confirm that an activity is risky, edit the Security Policy, educate users, or otherwise handle the risk.

Syntax for IPv4

```
fw [-d] sam_policy del '<Rule UID>'
```

Syntax for IPv6

```
fw6 [-d] sam_policy del '<Rule UID>'
```

Parameters

Parameter	Description
`-d`	Runs the command in debug mode. Use only if you troubleshoot the command itself. **Best Practice** - If you use this parameter, then redirect the output to a file, or use the `script` command to save the entire CLI session.
`'<Rule UID>'`	Specifies the UID of the rule you wish to delete. **Important:** ■ The quote marks and angle brackets (`'<...>'`) are mandatory. ■ To see the Rule UID, run the *"fw sam_policy get" on page 206* command.

Procedure

1. **List all the existing rules in the Suspicious Activity Monitoring policy database**

   List all the existing rules in the Suspicious Activity Monitoring policy database.

   - For IPv4, run:

     ```
     fw sam_policy get
     ```

   - For IPv6, run:

     ```
     fw6 sam_policy get
     ```

   The rules show in this format:

   ```
   operation=add uid=<Value1,Value2,Value3,Value4> target=...
   timeout=... action=... log= ... name= ... comment=...
   originator= ... src_ip_addr=... req_tpe=...
   ```

   Example for IPv4:

   ```
   operation=add uid=<5ac3965f,00000000,3403a8c0,0000264a>
   target=all timeout=300 action=notify log=log name=Test\ Rule
   comment=Notify\ about\ traffic\ from\ 1.1.1.1
   originator=John\ Doe src_ip_addr=1.1.1.1 req_tpe=ip
   ```

2. **Delete a rule from the list by its UID**

   - For IPv4, run:

     ```
     fw [-d] sam_policy del '<Rule UID>'
     ```

   - For IPv6, run:

     ```
     fw6 [-d] sam_policy del '<Rule UID>'
     ```

   Example for IPv4:

   ```
   fw samp del '<5ac3965f,00000000,3403a8c0,0000264a>'
   ```

3. **Add the flush-only rule**

   - For IPv4, run:

     ```
     fw samp add -t 2 quota flush true
     ```

   - For IPv6, run:

     ```
     fw6 samp add -t 2 quota flush true
     ```

   Explanation:

   The "`fw samp del`" and "`fw6 samp del`" commands only remove a rule from the persistent database. The Security Gateway continues to enforce the deleted rule until the next time you compiled and load a policy. To force the rule deletion immediately, you must enter a flush-only rule right after the "`fw samp del`" and "`fw6 samp del`" command. This flush-only rule immediately deletes the rule you specified in the previous step, and times out in 2 seconds.

   Best Practice - Specify a short timeout period for the flush-only rules. This prevents accumulation of rules that are obsolete in the database.

fw sam_policy get

Description

The "*fw sam_policy get*" and "*fw6 sam_policy get*" commands:

- Show all the configured Suspicious Activity Monitoring (SAM) rules.
- Show all the configured Rate Limiting rules.

Notes:

- These commands are interchangeable:
  - For IPv4: "`fw sam_policy`" and "`fw samp`".
  - For IPv6: "`fw6 sam_policy`" and "`fw6 samp`".
- You can run these commands in Gaia Clish, or Expert mode.
- Security Gateway stores the SAM Policy rules in the `$FWDIR/database/sam_policy.db` file.
- Security Gateway stores the SAM Policy management settings in the `$FWDIR/database/sam_policy.mng` file.

Important:

- Configuration you make with these commands, survives reboot.
- VSX mode does **not** support Suspicious Activity Policy configured in SmartView Monitor. See sk79700.
- In VSX mode, you must go to the context of an applicable Virtual System.
  - In Gaia Clish, run: `set virtual-system <VSID>`
  - In the Expert mode, run: `vsenv <VSID>`
- In a Cluster, you must configure all the Cluster Members in the same way.

Best Practice - The SAM Policy rules consume some CPU resources on Security Gateway. Set an expiration for rules that gives you time to investigate, but does not affect performance. Keep only the required SAM Policy rules. If you confirm that an activity is risky, edit the Security Policy, educate users, or otherwise handle the risk.

Syntax for IPv4

```
fw [-d] sam_policy get [-l] [-u '<Rule UID>'] [-k '<Key>' -t
<Type> [+{-v '<Value>'}] [-n]]
```

Syntax for IPv6

```
fw6 [-d] sam_policy get [-l] [-u '<Rule UID>'] [-k '<Key>' -t
<Type> [+{-v '<Value>'}] [-n]]
```

Parameters

Note - All these parameters are optional.

Parameter	Description
`-d`	Runs the command in debug mode. Use only if you troubleshoot the command itself. **Best Practice** - If you use this parameter, then redirect the output to a file, or use the `script` command to save the entire CLI session.
`-l`	Controls how to print the rules: ■ In the default format (without "`-l`"), the output shows each rule on a separate line. ■ In the list format (with "`-l`"), the output shows each parameter of a rule on a separate line. ■ See *"fw sam_policy add" on page 187*.
`-u '<Rule UID>'`	Prints the rule specified by its Rule UID or its zero-based rule index. The quote marks and angle brackets ('<...>') are mandatory.
`-k '<Key>'`	Prints the rules with the specified predicate key. The quote marks are mandatory.
`-t <Type>`	Prints the rules with the specified predicate type. For Rate Limiting rules, you must always use "`-t in`".
`+{-v '<Value>'}`	Prints the rules with the specified predicate values. The quote marks are mandatory.
`-n`	Negates the condition specified by these predicate parameters: ■ `-k` ■ `-t` ■ `+-v`

Examples

Example 1 - Output in the default format

```
[Expert@HostName:0]# fw samp get

operation=add uid=<5ac3965f,00000000,3403a8c0,0000264a> target=all timeout=300 action=notify
log=log name=Test\ Rule comment=Notify\ about\ traffic\ from\ 1.1.1.1 originator=John\ Doe
src_ip_addr=1.1.1.1 req_tpe=ip
```

Example 2 - Output in the list format

```
[Expert@HostName:0]# fw samp get -l

uid
<5ac3965f,00000000,3403a8c0,0000264a>
target
all
timeout
2147483647
action
notify
log
log
name
Test\ Rule
comment
Notify\ about\ traffic\ from\ 1.1.1.1
originator
John\ Doe
src_ip_addr
1.1.1.1
req_type
ip
```

Example 3 - Printing a rule by its Rule UID

```
[Expert@HostName:0]# fw samp get -u '<5ac3965f,00000000,3403a8c0,0000264a>'
0
operation=add uid=<5ac3965f,00000000,3403a8c0,0000264a> target=all timeout=300 action=notify
log=log name=Test\ Rule comment=Notify\ about\ traffic\ from\ 1.1.1.1 originator=John\ Doe
src_ip_addr=1.1.1.1 req_tpe=ip
```

Example 4 - Printing rules that match the specified filters

```
[Expert@HostName:0]# fw samp get
no corresponding SAM policy requests
[Expert@HostName:0]#
[Expert@HostName:0]# fw samp add -a d -l r -t 3600 quota service any source
range:172.16.7.11-172.16.7.13 new-conn-rate 5 flush true
[Expert@HostName:0]#
[Expert@HostName:0]# fw samp add -a n -l r quota service 1,50-51,6/443,17/53 service-negated
true source cc:QQ byte-rate 0
[Expert@HostName:0]#
[Expert@HostName:0]# fw samp add -a b quota source range:172.16.8.17-172.16.9.121 service
6/80
[Expert@HostName:0]#
[Expert@HostName:0]# fw samp add -a d quota service any source-negated true source cc:QQ
concurrent-conns-ratio 655 track source
[Expert@HostName:0]#
[Expert@HostName:0]# fw samp get
operation=add uid=<5bab3acf,00000000,3503a8c0,00003ddc> target=all timeout=indefinite
action=drop service=any source-negated=true source=cc:QQ concurrent-conns-ratio=655
track=source req_type=quota
operation=add uid=<5bab3ac6,00000000,3503a8c0,00003dbf> target=all timeout=3586 action=drop
log=log service=any source=range:172.16.7.11-172.16.7.13 new-conn-rate=5 flush=true req_
type=quota
operation=add uid=<5bab3acc,00000000,3503a8c0,00003dd7> target=all timeout=indefinite
action=bypass source=range:172.16.8.17-172.16.9.121 service=6/80 req_type=quota
operation=add uid=<5bab3ac9,00000000,3503a8c0,00003dd5> target=all timeout=indefinite
action=notify log=log service=1,50-51,6/443,17/53 service-negated=true source=cc:QQ byte-
rate=0 req_type=quota
[Expert@HostName:0]#
[Expert@HostName:0]# fw samp get -k 'service' -t in -v '6/80'
operation=add uid=<5bab3acc,00000000,3503a8c0,00003dd7> target=all timeout=indefinite
action=bypass source=range:172.16.8.17-172.16.9.121 service=6/80 req_type=quota
[Expert@HostName:0]#
[Expert@HostName:0]# fw samp get -k 'service-negated' -t in -v 'true'
operation=add uid=<5bab3ac9,00000000,3503a8c0,00003dd5> target=all timeout=indefinite
action=notify log=log service=1,50-51,6/443,17/53 service-negated=true source=cc:QQ byte-
rate=0 req_type=quota
[Expert@HostName:0]#
[Expert@HostName:0]# fw samp get -k 'source' -t in -v 'cc:QQ'
operation=add uid=<5bab3acf,00000000,3503a8c0,00003ddc> target=all timeout=indefinite
action=drop service=any source-negated=true source=cc:QQ concurrent-conns-ratio=655
track=source req_type=quota
operation=add uid=<5bab3ac9,00000000,3503a8c0,00003dd5> target=all timeout=indefinite
action=notify log=log service=1,50-51,6/443,17/53 service-negated=true source=cc:QQ byte-
rate=0 req_type=quota
[Expert@HostName:0]#
[Expert@HostName:0]# fw samp get -k source -t in -v 'cc:QQ' -n
operation=add uid=<5bab3ac6,00000000,3503a8c0,00003dbf> target=all timeout=3291 action=drop
log=log service=any source=range:172.16.7.11-172.16.7.13 new-conn-rate=5 flush=true req_
type=quota
operation=add uid=<5bab3acc,00000000,3503a8c0,00003dd7> target=all timeout=indefinite
action=bypass source=range:172.16.8.17-172.16.9.121 service=6/80 req_type=quota
[Expert@HostName:0]#
[Expert@HostName:0]# fw samp get -k 'source-negated' -t in -v 'true'
operation=add uid=<5baa94e0,00000000,860318ac,00003016> target=all timeout=indefinite
action=drop service=any source-negated=true source=cc:QQ concurrent-conns-ratio=655
track=source req_type=quota
[Expert@HostName:0]#
[Expert@HostName:0]# fw samp get -k 'byte-rate' -t in -v '0'
operation=add uid=<5baa9431,00000000,860318ac,00002efd> target=all timeout=indefinite
action=notify log=log service=1,50-51,6/443,17/53 service-negated=true source=cc:QQ byte-
rate=0 req_type=quota
[Expert@HostName:0]#
[Expert@HostName:0]# fw samp get -k 'flush' -t in -v 'true'
operation=add uid=<5baa9422,00000000,860318ac,00002eea> target=all timeout=2841 action=drop
log=log service=any source=range:172.16.7.11-172.16.7.13 new-conn-rate=5 flush=true req_
type=quota
[Expert@HostName:0]#
[Expert@HostName:0]# fw samp get -k 'concurrent-conns-ratio' -t in -v '655'
operation=add uid=<5baa94e0,00000000,860318ac,00003016> target=all timeout=indefinite
```

```
action=drop service=any source-negated=true source=cc:QQ concurrent-conns-ratio=655
track=source req_type=quota
[Expert@HostName:0]#
```

The /proc/ppk/ and /proc/ppk6/ entries

Description

SecureXL supports Linux **/proc** entries. The read-only entries in the **/proc/ppk/** and **/proc/ppk6/** contain various data about SecureXL.

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/<Name of File>
```

```
[Expert@MyGW:0]# cat /proc/ppk/<SecureXL Instance ID>/<Name of
File>
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/<Name of File>
```

```
[Expert@MyGW:0]# cat /proc/ppk6/<SecureXL Instance ID>/<Name of
File>
```

Files

File	Description
`affinity`	Contains status and the thresholds for SecureXL New Affinity mechanism. See *"/proc/ppk/affinity" on page 214*.
`conf`	Contains the SecureXL configuration and basic statistics. See *"/proc/ppk/conf" on page 215*.
`conns`	Contains the list of the SecureXL connections. See *"/proc/ppk/conns" on page 216*.
`cpls`	Contains SecureXL configuration for ClusterXL Load Sharing (CPLS). See *"/proc/ppk/cpls" on page 217*.
`cqstats`	Contains statistics for SecureXL connections queue. See *"/proc/ppk/cqstats" on page 218*.
`drop_statistics`	Contains SecureXL statistics for dropped packets. See *"/proc/ppk/drop_statistics" on page 219*.

File	Description
`ifs`	Contains the list of interfaces that SecureXL uses. See *"/proc/ppk/ifs" on page 220*.
`mcast_statistics`	Contains SecureXL statistics for multicast traffic. See *"/proc/ppk/mcast_statistics" on page 225*.
`nac`	Contains SecureXL statistics for Identity Awareness Network Access Control (NAC) traffic. See *"/proc/ppk/nac" on page 226*.
`notify_statistics`	Contains SecureXL statistics for notifications SecureXL sent to Firewall about accelerated connections. See *"/proc/ppk/notify_statistics" on page 227*.
`profile_cpu_stat`	Contains IDs of the CPU cores and status of Traffic Profiling See *"/proc/ppk/profile_cpu_stat" on page 229*.
`rlc`	Contains SecureXL statistics for drops due to Rate Limiting for DoS Mitigation. See *"/proc/ppk/rlc" on page 230*.
`statistics`	Contains SecureXL overall statistics. See *"/proc/ppk/statistics" on page 231*.
`stats`	Contains the IRQ numbers and names of interfaces the SecureXL uses. See *"/proc/ppk/stats" on page 233*.
`viol_statistics`	Contains SecureXL statistics for violations - packets SecureXL forwarded (F2F) to the Firewall. See *"/proc/ppk/viol_statistics" on page 234*.

/proc/ppk/affinity

Description

Contains the number of accelerated packets per second and rate of encrypted bytes.

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/affinity
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/affinity
```

Example for IPv4

```
[Expert@MyGW:0]# cat /proc/ppk/affinity
Current accelerated PPS        : 0
Current enc. bytes rate        : 0
[Expert@MyGW:0]#
```

/proc/ppk/conf

Description

Contains the SecureXL configuration and basic statistics.

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/conf
```

```
[Expert@MyGW:0]# cat /proc/ppk/<SecureXL Instance ID>/conf
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/conf
```

```
[Expert@MyGW:0]# cat /proc/ppk6/<SecureXL Instance ID>/conf
```

Example for IPv4

```
[Expert@MyGW:0]# cat /proc/ppk/conf
Flags                           : 0x00000592
Accounting Update Interval      : 3600
Conn Refresh Interval           : 512
SA Sync Notification Interval   : 200000
UDP Encapsulation Port          : 2746
Min TCP MSS                     : 0
TCP End Timeout                 : 5
Connection Limit                : 18446744073709551615

Total Number of conns           : 0
Number of Crypt conns           : 0
Number of TCP conns             : 0
Number of Non-TCP conns         : 0
Total Number of corrs           : 0

Debug flags :
 0     : 0x1
 1     : 0x1
 2     : 0x1
 3     : 0x1
 4     : 0x1
 5     : 0x1
 6     : 0x1
 7     : 0x1
 8     : 0x100
 9     : 0x8
 10     : 0x1
 11     : 0x10
[Expert@MyGW:0]#
```

/proc/ppk/conns

Description

Contains the list of the SecureXL connections.

Important - This file is for future use. Refer to the *"fwaccel conns" on page 41* command.

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
[Expert@MyGW:0]# cat /proc/ppk/conns
[Expert@MyGW:0]# cat /proc/ppk/<SecureXL Instance ID>/conns
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
[Expert@MyGW:0]# cat /proc/ppk6/conns
[Expert@MyGW:0]# cat /proc/ppk6/<SecureXL Instance ID>/conns
```

/proc/ppk/cpls

Description

Contains SecureXL configuration for ClusterXL Load Sharing (CPLS).

Important - This file is for future use. Refer to the "`fwaccel cfg -h`" command (see *"fwaccel cfg" on page 38*).

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/cpls
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/cpls
```

Example for IPv4

```
[Expert@MyGW:0]# cat /proc/ppk/cpls
fwha_conf_flags: 638
fwha_df_type: 0
fwha_member_id: 0
fwha_port: 8116
FWHAP MAC magic: 0
Forwarding MAC magic: 0
My state: ACTIVE
udp_enc_port: 0
selection table size: 0
[Expert@MyGW:0]#
```

/proc/ppk/cqstats

Description

Contains statistics for SecureXL connections queue.

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/cqstats
```

```
[Expert@MyGW:0]# cat /proc/ppk/<SecureXL Instance ID>/cqstats
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/cqstats
```

```
[Expert@MyGW:0]# cat /proc/ppk6/<SecureXL Instance ID>/cqstats
```

Example for IPv4

```
[Expert@MyGW:0]# cat /proc/ppk/cqstats
Name                  Value              Name                  Value
--------------------  ---------------    --------------------  ---------------
Queued pkts                         0    Queue fail                          0
Dequeue & f2f                       0    Dequeue & drop                      0
Dequeue & resume                    0    Async index req                     0
Err Async index req                 0    Async index cb                      0
Err Async index cb                  0    Queue alloc fail                    0
Queue empty err                     0
[Expert@MyGW:0]#
```

/proc/ppk/drop_statistics

Description

Contains SecureXL statistics for dropped packets.

Note - This is the same information that the "`fwaccel stats -d`" command shows (see *"fwaccel stats" on page 98*).

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/drop_statistics
```

```
[Expert@MyGW:0]# cat /proc/ppk/<SecureXL Instance ID>/drop_
statistics
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/drop_statistics
```

```
[Expert@MyGW:0]# cat /proc/ppk6/<SecureXL Instance ID>/drop_
statistics
```

Example for IPv4

```
[Expert@MyGW:0]# cat /proc/ppk/drop_statistics
Reason                Packets            Reason                Packets
--------------------  ---------------    --------------------  ---------------
general reason                      0    CPASXL decision                     0
PSLXL decision                      0    clr pkt on vpn                      0
encrypt failed                      0    drop template                       0
decrypt failed                      0    interface down                      0
cluster error                       0    XMT error                           0
anti spoofing                       0    local spoofing                      0
sanity error                        0    monitored spoofed                   0
QOS decision                        0    C2S violation                       0
S2C violation                       0    Loop prevention                     0
DOS Fragments                       0    DOS IP Options                      0
DOS Blacklists                      0    DOS Penalty Box                     0
DOS Rate Limiting                   0    Syn Attack                          0
Reorder                             0    Defrag timeout                      0
[Expert@MyGW:0]#
```

/proc/ppk/ifs

Description

Contains the list of interfaces that SecureXL uses.

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/ifs
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/ifs
```

Example for IPv4

```
[Expert@MyGW:0]# cat /proc/ppk/ifs
 No | Interface | Address         | IRQ | F   | SIM F | Dev                | Output Func
| Features
-------------------------------------------------------------------------------------------------
-------------
  2 | eth0      |    192.168.3.52 |  67 |   1 |   480 | 0xffff81023e5df000 | 0x000013a0
  3 | eth1      |     10.20.30.52 |  83 |   1 |   488 | 0xffff81023dd0c000 | 0x000013a0
  4 | eth2      |     40.50.60.52 |  59 |   1 |   480 | 0xffff810237f88000 | 0x000013a0
  5 | eth3      |         0.0.0.0 |  67 |   1 |    80 | 0xffff810239b3d000 | 0x000013a0
  6 | eth4      |         0.0.0.0 |  91 |   1 |    80 | 0xffff81023841f000 | 0x000013a0
  7 | eth5      |         0.0.0.0 |  83 |   1 |   480 | 0xffff8102396fe000 | 0x000013a0
  8 | eth6      |         0.0.0.0 |  59 |   1 |   480 | 0xffff810239a4d000 | 0x000013a0
 10 | bond0     |     70.80.90.52 |   0 |   1 |   280 | 0xffff8101f1a0e000 | 0x000013a0
[Expert@MyGW:0]#
```

Example for IPv6

```
[Expert@MyGW:0]# cat /proc/ppk6/ifs
  No | Interface | Address         | IRQ | F  | SIM F | Dev               | Output Func
| Features
-------------------------------------------------------------------------------------------------
-------------
  2 | eth0      |           fe80:0:0:0:250:56ff:fea3:1807 |  67 |   1 |   480 |
0xffff81023e5df000 | 0x000013a0
  3 | eth1      |           fe80:0:0:0:250:56ff:fea3:15a4 |  83 |   1 |   480 |
0xffff81023dd0c000 | 0x000013a0
  4 | eth2      |           fe80:0:0:0:250:56ff:fea3:2f50 |  59 |   1 |   480 |
0xffff810237f88000 | 0x000013a0
  5 | eth3      |                         0:0:0:0:0:0:0:0 |  67 |   1 |    80 |
0xffff810239b3d000 | 0x000013a0
  6 | eth4      |                         0:0:0:0:0:0:0:0 |  91 |   1 |    80 |
0xffff81023841f000 | 0x000013a0
  7 | eth5      |           fe80:0:0:0:250:56ff:fea3:75a9 |  83 |   1 |   480 |
0xffff8102396fe000 | 0x000013a0
  8 | eth6      |           fe80:0:0:0:250:56ff:fea3:5d4c |  59 |   1 |   480 |
0xffff810239a4d000 | 0x000013a0
 10 | bond0     |           fe80:0:0:0:250:56ff:fea3:287b |   0 |   1 |   280 |
0xffff8101f1a0e000 | 0x000013a0
[Expert@MyGW:0]#
```

Explanation about the configuration flags in the "F" and "SIM F" columns

The "**F**" column shows the internal configuration flags that Firewall set on these interfaces.

The "**SIM F**" column shows the internal configuration flags that SecureXL set on these interfaces.

Flag	Description
0x001	If this flag is set, the SecureXL drops the packet at the end of the inbound inspection, if the packet is a "cut-through" packet. In outbound, SecureXL forwards all the packets to the network.
0x002	If this flag is set, the SecureXL sends an applicable notification when a TCP state change occurs (connection is established or torn down).
0x004	If this flag is set, the SecureXL it sets the UDP header's checksum field correctly when the SecureXL encapsulates an encrypted packet (UDP encapsulation). If this flag is not set, SecureXL sets the UDP header's checksum field to zero. It is safe to ignore this flag, if it is set to 0 (SecureXL continues to calculate the UDP packet's checksum).
0x008	If this flag is set, the SecureXL does not create new connections that match a template, and SecureXL drops the packet that matches the template, when the number of entries in the Connections Table reaches the specified limit. If this flag is not set, the SecureXL forwards the packet to the Firewall.

Flag	Description
0x010	If this flag is set, the SecureXL forwards fragments to the Firewall.
0x020	If this flag is set, the SecureXL does not create connections from TCP templates anymore. The Firewall offloads connections to SecureXL when necessary. This flag only disables the creation of TCP templates.
0x040	If this flag is set, the SecureXL notifies the Firewall at intervals, so it refreshes the accelerated connections in the Firewall kernel tables.
0x080	If this flag is set, the SecureXL does not create connections from non-TCP templates anymore. The Firewall offloads connections to SecureXL when necessary. This flag only disables the creation of non-TCP templates.
0x100	If this flag is set, the SecureXL allows sequence verification violations for connections that did not complete the TCP 3-way handshake process. If this flag is not set, SecureXL must forward the violating packets to the Firewall.
0x200	If this flag is set, the SecureXL allows sequence verification violations for connections that completed the TCP 3-way handshake process. If this flag is not set, SecureXL must forward the violating packets to the Firewall.
0x400	If this flag is set, the SecureXL forwards TCP [RST] packets to the Firewall.
0x0001	If this flag is set, the SecureXL notifies the Firewall about HitCount data.
0x0002	If this flag is set, the VSX Virtual System works as a junction, rather than a regular Virtual System (only the local Virtual System flag is applicable).
0x0004	If this flag is set, the SecureXL disables the reply counter of inbound encrypted traffic. At a result, SecureXL kernel module works in the same way as the VPN kernel module.
0x0008	If this flag is set, the SecureXL enables the MSS Clamping. Refer to the kernel parameters "`fw_clamp_tcp_mss`" and "`fw_clamp_vpn_mss`" in sk101219.
0x0010	If this flag is set, the SecureXL disables the "No Match Ranges" (NMR) Templates (see sk117755).

Flag	Description
0x0020	If this flag is set, the SecureXL disables the "No Match Time" (NMT) Templates (see sk117755).
0x0040	If this flag is set, the SecureXL does not send Drop Templates notifications about dropped packets to the Firewall (to update the drop counters). For example, if you set the value of the kernel parameter "`activate_optimize_drops_support_now`" to 1, it disables the Drop Templates notifications.
0x0080	If this flag is set, the SecureXL enables the MultiCore support for IPsec VPN (see sk118097).
0x0100	If this flag is set, the SecureXL enables the support for CoreXL Dynamic Dispatcher (see sk105261).
0x0800	If this flag is set, the SecureXL does not enforce the Path MTU Discovery for IP multicast packets.
0x1000	If this flag is set, the SecureXL disables the SIM "drop_templates" feature.
0x2000	If this flag is set, it indicates that an administrator enabled the Link Selection Load Sharing feature.
0x4000	If this flag is set, the SecureXL disables the asynchronous notification feature.
0x8000	If this flag is set, it indicates that the capacity of the Firewall Connections Table is unlimited.

Examples:

Value	Description
0x039	Means the sum of these flags: ■ 0x001 ■ 0x008 ■ 0x010 ■ 0x020
0x00008a16	Means the sum of these flags: ■ 0x0002 ■ 0x0004 ■ 0x0010 ■ 0x0200 ■ 0x0800 ■ 0x8000
0x00009a16	Means the sum of these flags: ■ 0x0002 ■ 0x0004 ■ 0x0010 ■ 0x0200 ■ 0x0800 ■ 0x1000 ■ 0x8000

/proc/ppk/mcast_statistics

Description

Contains SecureXL statistics for multicast traffic.

Note - This is the same information that the "`fwaccel stats -m`" command shows (see *"fwaccel stats" on page 98*).

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/mcast_statistics
```

```
[Expert@MyGW:0]# cat /proc/ppk/<SecureXL Instance ID>/mcast_
statistics
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/mcast_statistics
```

```
[Expert@MyGW:0]# cat /proc/ppk6/<SecureXL Instance ID>/mcast_
statistics
```

Example for IPv4

```
[Expert@MyGW:0]# cat /proc/ppk/mcast_statistics
Name                  Value              Name                  Value
--------------------  ---------------    --------------------  ---------------
in packets                      10100    out packets                         0
if restricted                       0    conns with down if                  0
f2f packets                         0    f2f bytes                           0
dropped packets                     0    dropped bytes                       0
accel packets                       0    accel bytes                         0
mcast conns                         0
[Expert@MyGW:0]#
```

/proc/ppk/nac

Description

Contains SecureXL statistics for Identity Awareness Network Access Control (NAC) traffic.

Note - This is the same information that the "`fwaccel stats -n`" command shows (see *"fwaccel stats" on page 98*).

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/nac
```

```
[Expert@MyGW:0]# cat /proc/ppk/<SecureXL Instance ID>/nac
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/nac
```

```
[Expert@MyGW:0]# cat /proc/ppk6/<SecureXL Instance ID>/nac
```

Example for IPv4

```
[Expert@MyGW:0]# cat /proc/ppk/nac
Name                  Value                Name                  Value
--------------------  ---------------      --------------------  ---------------
NAC packets                         0      NAC bytes                           0
NAC connections                     0      complience failure                  0
[Expert@MyGW:0]#
```

/proc/ppk/notify_statistics

Description

Contains SecureXL statistics for notifications SecureXL sent to Firewall about accelerated connections.

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/notify_statistics
```

```
[Expert@MyGW:0]# cat /proc/ppk/<SecureXL Instance ID>/notify_
statistics
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/notify_statistics
```

```
[Expert@MyGW:0]# cat /proc/ppk6/<SecureXL Instance ID>/notify_
statistics
```

Example for IPv4

```
[Expert@MyGW:0]# cat /proc/ppk/notify_statistics
Notification           Packets           Notification           Packets
---------------------  --------------    ---------------------  --------------
ntSAAboutToExpire                   0    ntSAExpired                         0
ntMSPIError                         0    ntNoInboundSA                       0
ntNoOutboundSA                      0    ntDataIntegrityFailed               0
ntPossibleReplay                    0    ntReplay                            0
ntNextProtocolError                 0    ntCPIError                          0
ntClearTextPacket                   0    ntFragmentation                     0
ntUpdateUdpEncTable                 0    ntSASync                            0
ntReplayOutOfWindow                 0    ntVPNTrafficReport                  0
ntConnDeleted                       0    ntConnUpdate                        0
ntPacketDropped                     0    ntSendLog                           0
ntRefreshGTPTunnel                  0    ntMcastDrop                         0
ntAccounting                        0    ntAsyncIndex                        0
ntACkReordering                     0    ntAccelAckInfo                      0
ntMonitorPacket                     0    ntPacketCapture                     0
ntCpasPacketCapture                 0    ntPSLGlueUpdateReject               0
ntSeqVerifyDrop                     0    ntPacketForwardBefore               0
ntICMPMessage                       0    ntQoSReclassifyPacket               0
ntQoSResumePacket                   0    ntVPNEncHaLinkFailure               0
ntVPNEncLsLinkFailure               0    ntVPNEncRouteChange                 0
ntVPNDecVerRouteChang               0    ntVPNDecRouteChange                 0
ntMuxSimToFw                        0    ntPSLEventLog                       0
ntSendCPHWDStats                39375    ntPacketTaggingViolat               0
ntDosNotify                         0    ntSynatkNotify                      0
ntSynatkStats                       0    ntQoSEventLog                       0
ntPrintGetParam                     0
[Expert@MyGW:0]#
```

/proc/ppk/profile_cpu_stat

Description

This file is for Check Point use only.

Contains IDs of the CPU cores and status of Traffic Profiling:

- The first column shows the IDs of the CPU cores.
- The second column shows the status of Traffic Profiling for the applicable CPU core.

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/profile_cpu_stat
```

```
[Expert@MyGW:0]# cat /proc/ppk/<SecureXL Instance ID>/profile_cpu_stat
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/profile_cpu_stat
```

```
[Expert@MyGW:0]# cat /proc/ppk6/<SecureXL Instance ID>/profile_cpu_stat
```

Example for IPv4 from a Security Gateway with 4 CPU cores

```
[Expert@MyGW:0]# cat /proc/ppk/profile_cpu_stat
0 0
1 0
2 0
3 0
[Expert@MyGW:0]#
```

/proc/ppk/rlc

Description

Contains SecureXL statistics for drops due to Rate Limiting for DoS Mitigation.

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/rlc
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/rlc
```

Example for IPv4

```
[Expert@MyGW:0]# cat /proc/ppk/rlc
Total drop packets : 0
Total drop bytes : 0
[Expert@MyGW:0]#
```

/proc/ppk/statistics

Description

Contains SecureXL overall statistics.

To see these statistics in a better way, run the *"fwaccel stats" on page 98* command.

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/statistics
```

```
[Expert@MyGW:0]# cat /proc/ppk/<SecureXL Instance ID>/statistics
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/statistics
```

```
[Expert@MyGW:0]# cat /proc/ppk6/<SecureXL Instance ID>/statistics
```

Example for IPv4

```
[Expert@MyGW:0]# cat /proc/ppk/statistics
Name                  Value                 Name                  Value
--------------------  ---------------       --------------------  ---------------
accel packets                       0       accel bytes                         0
outbound packets                    0       outbound bytes                      0
conns created                       0       conns deleted                       0
current total conns                 0       TCP conns                           0
non TCP conns                       0       nat conns                           0
dropped packets                   728       dropped bytes                  107978
fragments received                  0       fragments transmit                  0
fragments dropped                   0       fragments expired                   0
IP options stripped                 0       IP options restored                 0
IP options dropped                  0       corrs created                       0
corrs deleted                       0       C corrections                       0
corrected packets                   0       corrected bytes                     0
crypt conns                         0       enc bytes                           0
dec bytes                           0       ESP enc pkts                        0
ESP enc err                         0       ESP dec pkts                        0
ESP dec err                         0       ESP other err                       0
espudp enc pkts                     0       espudp enc err                      0
espudp dec pkts                     0       espudp dec err                      0
espudp other err                    0       acct update interval             3600
CPASXL packets                      0       PSLXL packets                       0
CPASXL async packets                0       PSLXL async packets                 0
CPASXL bytes                        0       PSLXL bytes                         0
CPASXL conns                        0       PSLXL conns                         0
CPASXL conns created                0       PSLXL conns created                 0
PXL FF conns                        0       PXL FF packets                      0
PXL FF bytes                        0       PXL FF acks                         0
PXL no conn drops                   0       PSL Inline packets                  0
PSL Inline bytes                    0       CPAS Inline packets                 0
CPAS Inline bytes                   0       Total QoS conns                     0
CLASSIFY                            0       CLASSIFY_FLOW                       0
RECLASSIFY_POLICY                   0       Enq-IN FW pkts                      0
Enq-OUT FW pkts                     0       Deq-IN FW pkts                      0
Deq-OUT FW pkts                     0       Enq-IN FW bytes                     0
Enq-OUT FW bytes                    0       Deq-IN FW bytes                     0
Deq-OUT FW bytes                    0       Enq-IN AXL pkts                     0
Enq-OUT AXL pkts                    0       Deq-IN AXL pkts                     0
Deq-OUT AXL pkts                    0       Enq-IN AXL bytes                    0
Enq-OUT AXL bytes                   0       Deq-IN AXL bytes                    0
Deq-OUT AXL bytes                   0       F2F packets                         0
F2F bytes                           0       TCP violations                      0
F2V conn match pkts                 0       F2V packets                         0
F2V bytes                           0       gtp tunnels created                 0
gtp tunnels                         0       gtp accel pkts                      0
gtp f2f pkts                        0       gtp spoofed pkts                    0
gtp in gtp pkts                     0       gtp signaling pkts                  0
gtp tcpopt pkts                     0       gtp apn err pkts                    0
memory used                  38799384       C tcp handshake conn                0
C tcp estab. conns                  0       C tcp closed conns                  0
C tcp pxl hnshk conn                0       C tcp pxl est. conn                 0
C tcp pxl closed                    0       ob cpasxl packets                   0
ob pslxl packets                    0       ob cpasxl bytes                     0
ob pslxl bytes                      0       DNS DoR stats                       0
trimmed pkts
[Expert@MyGW:0]#
```

/proc/ppk/stats

Description

Contains the IRQ numbers and names of interfaces the SecureXL uses.

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/stats
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/stats
```

Example for IPv4

```
[Expert@MyGW:0]# cat /proc/ppk/stats
IRQ | Interface
---------------------------
 18   eth0
 16   eth1
 17   eth2
 18   eth3
 19   eth4
[Expert@MyGW:0]#
```

/proc/ppk/viol_statistics

Description

Contains SecureXL statistics for violations - packets SecureXL forwarded (F2F) to the Firewall.

Note - This is the same information that the "`fwaccel stats -p`" command shows (see *"fwaccel stats" on page 98*).

Syntax for IPv4

```
[Expert@MyGW:0]# ls -lR /proc/ppk/
```

```
[Expert@MyGW:0]# cat /proc/ppk/viol_statistics
```

Syntax for IPv6

```
[Expert@MyGW:0]# ls -lR /proc/ppk6/
```

```
[Expert@MyGW:0]# cat /proc/ppk6/viol_statistics
```

Example for IPv4

```
[Expert@MyGW:0]# cat /proc/ppk/viol_statistics
Violation             Packets            Violation             Packets
--------------------  ---------------    --------------------  ---------------
pkt has IP options                  0    ICMP miss conn                      4
TCP-SYN miss conn                 356    TCP-other miss conn           1386954
UDP miss conn                  943355    other miss conn                     0
VPN returned F2F                    0    uni-directional viol                0
possible spoof viol                 0    TCP state viol                      0
out if not def/accl                 0    bridge, src=dst                     0
routing decision err                0    sanity checks failed                0
fwd to non-pivot                    0    broadcast/multicast                 0
cluster message             250859051    cluster forward                     0
chain forwarding                    0    F2V conn match pkts                 0
general reason                      0    route changes                       0

[Expert@MyGW:0]#
```

SecureXL Debug

To understand how SecureXL processes the traffic, enable the SecureXL debug while the traffic passes through the Security Gateway.

❗ **Warning** - Debug increases the load on Security Gateway's CPU. We recommend you schedule a maintenance window to debug the SecureXL.

In addition, see *"Kernel Debug on Security Gateway" on page 417*.

fwaccel dbg

Description

The *fwaccel dbg* command controls the SecureXL debug. See *"SecureXL Debug Procedure" on page 242*.

Important - In a Cluster, you must configure all the Cluster Members in the same way.

Syntax in Gaia Clish or the Expert mode on a Security Gateway / ClusterXL:

```
fwaccel dbg
      -h
      -m <Name of SecureXL Debug Module>
      all
      + <Debug Flags>
      - <Debug Flags>
      reset
      -f {"<5-Tuple Debug Filter>" | reset}
      list
      resetall
```

Parameters

Parameter	Description
`-h`	Shows the applicable built-in help.
`-m <Name of SecureXL Debug Module>`	Specifies the name of the SecureXL debug module. To see the list of available debug modules, run: `fwaccel dbg`
`all`	Enables all debug flags for the specified debug module.
`+ <Debug Flags>`	Enables the specified debug flags for the specified debug module: Syntax: `+ Flag1 [Flag2 Flag3 ... FlagN]` **Note** - You must press the space bar key after the plus (+) character.

Parameter	Description
`- <Debug Flags>`	Disables all debug flags for the specified debug module. Syntax: `- Flag1 [Flag2 Flag3 ... FlagN]` **Note** - You must press the space bar key after the minus (-) character.
`reset`	Resets all debug flags for the specified debug module to their default state.
`-f "<5-Tuple Debug Filter>"`	Configures the debug filter to show only debug messages that contain the specified connection. The filter is a string of five numbers separated with commas: `"<Source IP Address>,<Source Port>,<Destination IP Address>,<Destination Port>,<Protocol Number>"` **Notes:** - You can configure only one debug filter at one time. - You can use the asterisk "`*`" as a wildcard for an IP Address, Port number, or Protocol number. - For more information, see *IANA Service Name and Port Number Registry* and *IANA Protocol Numbers*.
`-f reset`	Resets the current debug filter.
`list`	Shows all enabled debug flags in all debug modules.
`resetall`	Reset all debug flags for all debug modules to their default state.

Examples

Example 1 - Default output

```
[Expert@MyGW:0]# fwaccel dbg
Usage: fwaccel dbg [-m <...>] [resetall | reset | list | all | +/- <flags>]
   -m <module>            - module of debugging
   -h                     - this help message
   resetall               - reset all debug flags for all modules
   reset                  - reset all debug flags for module
   all                    - set all debug flags for module
   list                   - list all debug flags for all modules
   -f reset | "<5-tuple>"  - filter debug messages
   + <flags>              - set the given debug flags
   - <flags>              - unset the given debug flags

List of available modules and flags:

Module: default (default)
err init drv tag lock cpdrv routing kdrv gtp tcp_sv gtp_pkt svm iter conn htab del update
acct conf stat queue ioctl corr util rngs relations ant conn_app rngs_print infra_ids offload
nat

Module: db
err get save del tmpl tmo init ant profile nmr nmt

Module: api
err init add update del acct conf stat vpn notif tmpl sv pxl qos gtp infra tmpl_info upd_conf
upd_if_inf add_sa del_sa del_all_sas misc get_features get_tab get_stat reset_stat tag long_
ver del_all_tmpl get_state upd_link_sel

Module: pkt
err f2f frag spoof acct notif tcp_state tcp_state_pkt sv cpls routing drop pxl qos user
deliver vlan pkt nat wrp corr caf

Module: infras
err reorder pm

Module: tmpl
err dtmpl_get dtmpl_notif tmpl

Module: vpn
err vpnpkt linksel routing vpn

Module: nac
err db db_get pkt pkt_ex signature offload idnt ioctl nac

Module: cpaq
init client server exp cbuf opreg transport transport_utils error

Module: synatk
init conf conn err log pkt proxy state msg

Module: adp
err rt nh eth heth wrp inf mbs bpl bplinf mbeinf if drop bond xmode ipsctl xnp

Module: dos
fw1-cfg fw1-pkt sim-cfg sim-pkt err detailed drop

[Expert@MyGW:0]#
```

Example 2 - Enabling and disabling of debug flags

```
[Expert@MyGW:0]# fwaccel dbg -m default + err conn
Debug flags updated.
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dbg list

Module: default (2001)
err conn

Module: db (1)
err

Module: api (1)
err

Module: pkt (1)
err

Module: infras (1)
err

Module: tmpl (1)
err

Module: vpn (1)
err

Module: nac (1)
err

Module: cpaq (100)
error

Module: synatk (0)


Module: adp (1)
err

Module: dos (10)
err

Debug filter not set.
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dbg -m default - conn
Debug flags updated.
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dbg list

Module: default (1)
err

Module: db (1)
err

Module: api (1)
err

Module: pkt (1)
err

Module: infras (1)
err

Module: tmpl (1)
```

```
err

Module: vpn (1)
err

Module: nac (1)
err

Module: cpaq (100)
error

Module: synatk (0)


Module: adp (1)
err

Module: dos (10)
err

Debug filter not set.
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dbg -m default reset
Debug flags updated.
[Expert@MyGW:0]#
```

Example 3 - Resetting all debug flags in all debug modules

```
[Expert@MyGW:0]# fwaccel dbg resetall
Debug state was reset to default.
[Expert@MyGW:0]#
```

Example 4 - Configuring debug filter for an SSH connection from 192.168.20.30 to 172.16.40.50

```
[Expert@MyGW:0]# fwaccel dbg -f 192.168.20.30,*,172.16.40.50,22,6
Debug filter was set.
[Expert@MyGW:0]#
[Expert@MyGW:0]# fwaccel dbg list

... ...

Debug filter: "<*,*,*,*,*>"
[Expert@MyGW:0]#
```

SecureXL Debug Procedure

By default, SecureXL writes the output debug information to the `/var/log/messages` file.

To collect the applicable SecureXL debug and to make its analysis easier, follow the steps below.

Note - For more information, see the [R81 Quantum Security Gateway Guide](#) - Chapter *Kernel Debug on Security Gateway*.

Important:

- We strongly recommend to schedule a full maintenance window to minimize the impact on your production traffic.
- We strongly recommend to connect over serial console to your Security Gateway.
  This is to avoid a possible issue when you cannot work with the CLI because of a high load on the CPU.
- In cluster, you must collect this debug from all Cluster Members in the same way.
- Debug the specific SecureXL instance only when you are sure that only that SecureXL instance processes the traffic.

Procedure

1. **Connect to the command line on your Security Gateway**

   Use an SSH or a console connection.

   Best Practice - Use a console connection.

2. **Log in to the Expert mode**

   If the default shell is Gaia Clish, then run:

   ```
   expert
   ```

3. **Reset all kernel debug flags in all kernel debug modules**

   Run:

   ```
   fw ctl debug 0
   ```

4. **Reset all the SecureXL debug flags in all SecureXL debug modules**

- For all SecureXL instances, run:

  ```
  fwaccel dbg resetall
  ```

- For a specific SecureXL instance, run:

  ```
  fwaccel -i <SecureXL ID> dbg resetall
  ```

5. **Allocate the kernel debug buffer**

   Run:

   ```
   fw ctl debug -buf 8200 [-v {"<List of VSIDs>" | all}]
   ```

   Note - The optional part "`-v {"<List of VSIDs>" | all}`" is to specify the applicable Virtual Systems on a VSX Gateway or VSX Cluster Member.

6. **Make sure the Security Gateway allocated the kernel debug buffer**

   Run:

   ```
   fw ctl debug | grep buffer
   ```

7. **Configure the applicable kernel debug modules and kernel debug flags**

   Run:

   ```
   fw ctl debug -m <Name of Kernel Debug Module> {all | +
   <Kernel Debug Flags>}
   ```

8. **Configure the applicable SecureXL debug modules and SecureXL debug flags**

   - For all SecureXL instances, run:

     ```
     fwaccel dbg -m <Name of SecureXL Debug Module> {all | +
     <SecureXL Debug Flags>}
     ```

   - For a specific SecureXL instance, run:

     ```
     fwaccel -i <SecureXL ID> dbg -m <Name of SecureXL Debug
     Module> {all | + <SecureXL Debug Flags>}
     ```

   See *"SecureXL Debug Modules and Debug Flags" on page 246*.

9. **Examine the kernel debug configuration for kernel debug modules**

   Run:

```
fw ctl debug
```

10. **Examine the SecureXL debug configuration for SecureXL debug modules**

    - For all SecureXL instances, run:

      ```
      fwaccel dbg list
      ```

    - For a specific SecureXL instance, run:

      ```
      fwaccel -i <SecureXL ID> dbg list
      ```

11. **Remove all entries from both the Firewall Connections table and SecureXL Connections table**

    Run:

    ```
    fw tab -t connections -x -y
    ```

    Important:

    - This step makes sure that you collect the debug of the real issue that is not affected by the existing connections.
    - **This command deletes all existing connections. This interrupts all connections, including the SSH.**
      Run this command only if you are connected over a serial console to your Security Gateway.

12. **Remove all entries from the Firewall Templates table**

    Run:

    ```
    fw tab -t cphwd_tmpl -x -y
    ```

    Note - This command does **not** interrupt the existing connections. This step makes sure that you collect the debug of the real issue that is not affected by the existing connection templates.

13. **Start the kernel debug**

    Run:

    ```
    fw ctl kdebug -T -f > /var/log/kernel_debug.txt
    ```

14. **Replicate the issue, or wait for the issue to occur**

    Perform the steps that cause the issue to occur, or wait for it to occur.

15. **Stop the kernel debug**

    Press **CTRL+C**.

16. **Reset all kernel debug flags in all kernel debug modules**

    Run:

    ```
    fw ctl debug 0
    ```

17. **Reset all the SecureXL debug flags in all SecureXL debug modules**

    - For all SecureXL instances, run:

      ```
      fwaccel dbg resetall
      ```

    - For a specific SecureXL instance, run:

      ```
      fwaccel -i <SecureXL ID> dbg resetall
      ```

18. **Examine the kernel debug configuration to make sure it returned to the default**

    Run:

    ```
    fw ctl debug
    ```

19. **Examine the SecureXL debug configuration to make sure it returned to the default**

    - For all SecureXL instances, run:

      ```
      fwaccel dbg list
      ```

    - For a specific SecureXL instance, run:

      ```
      fwaccel -i <SecureXL ID> dbg list
      ```

20. **Collect and analyze the debug output file**

    Path to the debug output file:

    ```
    /var/log/kernel_debug.txt
    ```

    Best Practice - Compress this file with the "`tar -zxvf`" command and transfer it from the Security Gateway to your computer. If you transfer to an FTP server, do so in the binary mode.

SecureXL Debug Modules and Debug Flags

To see the available SecureXL debug modules and their debug flags, run the *"fwaccel dbg" on page 236* command.

Module "default"

Flag	Description
`acct`	Connection accounting information
`ant`	Anticipated connections
`conf`	Configuration of the SecureXL (for example, interfaces)
`conn`	Processing of connections
`conn_app`	Processing of connections
`corr`	Correction layer
`cpdrv`	*Currently not in use*
`del`	Deletion of connections
`drv`	Driver information
`err`	General errors
`gtp`	Processing of GTP tunnel connections
`gtp_pkt`	Processing of GTP tunnel packets
`htab`	Hash table
`infra_ids`	Allocating IDs for a given range in Identity Awareness
`init`	Initialization
`ioctl`	Changes in the configuration, which were initiated from the user space
`iter`	Connection table iterator
`kdrv`	Driver information
`lock`	Lock initializing and finalizing
`nat`	Processing of NAT connections
`offload`	Offloading of connections from the Firewall to the SecureXL

Flag	Description
`queue`	Connections queue
`relations`	Related connections (such as FTP data connections)
`rngs`	Handling of SecureXL ranges
`rngs_print`	Printing of SecureXL ranges
`routing`	Handling of SecureXL routing
`stat`	Handling of SecureXL statistics
`svm`	Registering templates or connections for System Counters in Security Gateway object in SmartConsole
`tag`	Tags that were added to the packets by the SecureXL before forwarding them to the Firewall
`tcp_sv`	Verification of sequence in TCP packets
`update`	Updates of connections
`util`	Utilization

Module "pkt" (Packet)

Flag	Description
`acct`	Connection accounting information
`caf`	Mirror and Decrypt feature - Mirror only of all traffic
`corr`	Correction layer
`cpls`	ClusterXL Load Sharing
`deliver`	Packet delivery
`drop`	Packets dropped by SecureXL
`err`	General errors
`f2f`	Reason for forwarding a packet to the Firewall
`frag`	Processing of fragments

Flag	Description
nat	Processing of NAT connections
notif	Notifications sent to the Firewall
pkt	Processing of packets
pxl	PXL (PacketXL) handling - API between the SecureXL and PSL (Packet Streaming Layer), which is a TCP Streaming engine that parses TCP streams
qos	QoS acceleration
routing	Handling of SecureXL routing
spoof	Handling of SecureXL Anti-Spoofing
sv	Validation of sequence in TCP packets
tcp_state	Validation of TCP state in TCP packets
tcp_state_pkt	Validation of TCP packets
<*Username*>	*Currently not in use*
vlan	Handling of VLAN tags
wrp	Handling of WRP interfaces in VSX

Module "db" (Database)

Flag	Description
ant	Anticipated connections
del	Deleting of data from the SecureXL database
err	General errors
get	Retrieving of data from the SecureXL database
init	Initializing and finalizing of SecureXL database
nmr	"No Match Ranges" templates, which allow SecureXL Accept Templates for rules that contain Dynamic objects or Domain objects (or for rules located below such rules)

Flag	Description
`nmt`	"No Match Time" templates, which allow SecureXL Accept Templates for rules that contain Time objects (or for rules located below such rules)

`<`
`Profile>`
	Operations on profile table

`save`	Saving of data to the SecureXL database
`tmo`	Handling of timeouts for SecureXL database entries
`tmpl`	Handling of SecureXL templates database

Module "api" (Application Programmable Interface)

Flag	Description
`acct`	Connection accounting information
`add`	Adding of connections
`add_sa`	Offloading of VPN SA to SecureXL
`conf`	Configuration of the SecureXL (for example, interfaces)
`del`	Deletion of connections
`del_all_sas`	Deletion of all VPN SAs from SecureXL
`del_all_tmpl`	Deletion of the SecureXL Templates
`del_sa`	Deletion of VPN SA from SecureXL
`err`	General errors
`get_features`	Getting features buffer (in SecureXL initialization)
`get_stat`	Retrieving of SecureXL statistics
`get_state`	Getting the connection state from SecureXL
`get_tab`	Some extra printouts when processing SecureXL tables
`gtp`	Processing of GTP tunnel connections

Flag	Description
`infra`	SecureXL infrastructure
`init`	Enabling and disabling of SecureXL
`long_ver`	Prints additional verbose information about connections
`misc`	Prints additional information about SecureXL internals
`notif`	Notifications sent to the Firewall
`pxl`	PXL (PacketXL) handling - API between the SecureXL and PSL (Packet Streaming Layer), which is a TCP Streaming engine that parses TCP streams
`qos`	QoS acceleration
`reset_stat`	Prints statistics IDs that are reset
`stat`	Handling of SecureXL statistics
`sv`	Validation of sequence in TCP packets
`tag`	Tags that were added to the packets by the SecureXL before forwarding them to the Firewall
`tmpl`	Handling of SecureXL Templates
`tmpl_info`	Information about SecureXL Templates
`upd_conf`	Update of SecureXL in ClusterXL Load Sharing
`upd_if_inf`	Prints some text that shows if SecureXL updated information about interfaces
`upd_link_sel`	Updates of VPN Link Selection
`update`	Updates of connections
`vpn`	Processing of VPN connection

Module "adp"

Reserved for future use.

Module "infras" (Identity Awareness - Identities Infrastructure)

Flag	Description
`err`	General errors
`pm`	Pattern Matcher
`reorder`	Reordering of packets in queue

Module "nac" (Identity Awareness - Network Access Control)

Flag	Description
`db`	Updating, adding, deleting of identities
`db_get`	Updating, fetching, searching of identities
`err`	General errors
`idnt`	Identity Tags
`ioctl`	Changes in the configuration, which were initiated from the user space
`nac`	Network Access Control
`offload`	Offloading of connections from the Firewall to the SecureXL
`pkt`	Forwarding of connections to Firewall (when identity is not found or revoked, or NAC packet tagging verification failed)
`pkt_ex`	NAC packet-tagging verification
`signature`	Signing of packets

Module "vpn" (VPN)

Flag	Description
`err`	General errors
`linksel`	VPN Link Selection
`routing`	VPN Encryption routing information
`vpn`	Processing of VPN connections
`vpnpkt`	Processing of VPN packets

Module "cpaq" (Internal Asynchronous Queue)

Flag	Description
`cbuf`	Information about queue buffers
`client`	Information about queue clients
`error`	General errors
`exp`	Information about expiration of queue items
`init`	Initializing of queue
`opreg`	*Currently not in use*
`<`*`Mgmt Server`*`>`	Information about queue servers
`transport`	Information about sending messages in queue
`transport_utils`	Additional information about sending messages in queue

Module "dos" (Denial of Service Defender)

Flag	Description
`detailed`	Detailed tracing of DoS Rate Limiting logic in the packet flow. **Important** - This debug flag is not suitable for large traffic volumes because it prints a large number of messages. This causes high load on the CPU.
`drop`	Dropped packets
`err`	General errors
`fw1-cfg`	Information about DoS Rate Limiting configuration in the Firewall kernel module
`fw1-pkt`	Information about DoS Rate Limiting packet flow in the Firewall kernel module
`sim-cfg`	Information about DoS Rate Limiting configuration in the SecureXL kernel module
`sim-pkt`	Information about DoS Rate Limiting packet flow in the SecureXL kernel module

Module "synatk" (Accelerated SYN Defender)

Flag	Description
`conf`	Receiving and updating of Accelerated SYN Defender module's configuration
`conn`	Handling of TCP connections
`err`	General errors
`init`	Initializing of the Accelerated SYN Defender module
`log`	Prints time of the last sent monitor log and interval between the monitor logs
`msg`	Information about internal messages in the Accelerated SYN Defender module
`pkt`	Handling of TCP packets
`proxy`	*Currently not in use*
`state`	Information about states of the Accelerated SYN Defender module

Module "tmpl" (Drop Templates)

Flag	Description
`err`	General errors
`dtmpl_get`	Getting of Drop Templates
`dtmpl_notif`	Notifications about Drop Templates
`tmpl`	Information about Drop Templates

CoreXL

CoreXL is a performance-enhancing technology for Security Gateways on multi-core platforms.

CoreXL makes it possible for the CPU cores to perform multiple tasks concurrently. This enhances the Security Gateway performance.

CoreXL provides almost linear scalability of performance, according to the number of processing cores on a single machine. The increase in performance does not require changes to management or to network topology.

On a Security Gateway with CoreXL enabled, the Firewall kernel is replicated multiple times.

Each replicated copy of the Firewall kernel, or CoreXL Firewall instance, runs on one CPU core.

These CoreXL Firewall instances handle traffic concurrently, and each CoreXL Firewall instance is a complete and independent Firewall inspection kernel. When CoreXL is enabled, all the Firewall kernel instances in the Security Gateway process traffic through the same interfaces and apply the same security policy.

CoreXL Firewall instances work with SecureXL instances.

Enabling and Disabling CoreXL

Important Notes for Cluster:

- You must configure the CoreXL in the same way on all the cluster members.
- If you enable CoreXL, disable CoreXL, or change the number of CoreXL Firewall instances, you should treat this change as a version upgrade. Schedule a full maintenance window and follow the instructions in the *R81 Installation and Upgrade Guide* - Chapter *Upgrading ClusterXL Deployments*. Perform either a *Minimal Effort Upgrade* procedure (requires downtime), or a *Zero Downtime Upgrade* procedure (no downtime, but active connections are lost). Instead of the version upgrade, configure the CoreXL on each cluster member.

To change the CoreXL configuration

Step	Instructions
1	Connect to the command line on the Security Gateway.
2	Log in to Gaia Clish or Expert mode.
3	Run: `cpconfig`
4	Enter the number of the **Check Point CoreXL** option.

5	Enter the number of the applicable option:
	`(1) Change the number of firewall instances`
	`(2) Change the number of IPv6 firewall instances`
	`(3) Disable Check Point CoreXL`

6	Follow the instructions on the screen.
7	Exit from the `cpconfig` menu.
8	Reboot the Security Gateway.

Default Configuration of CoreXL

Important - This default configuration applies **only** to Security Gateways that do not support Dynamic Balancing of CoreXL Instances. See *"Dynamic Balancing of CoreXL Instances" on page 278*.

When you enable CoreXL, the default number of CoreXL Firewall instances is based on the total number of CPU cores.

The default affinity setting for all interfaces is automatic when SecureXL is enabled. See *"Allocation of Processing CPU Cores" on page 270*.

Traffic from all interfaces is directed to the CPU cores that run the CoreXL Secure Network Distributor (SND).

Default number of IPv4 CoreXL Firewall instances

Number of CPU cores	Default number of CoreXL IPv4 FW instances	Default number of Secure Network Distributors (SNDs)
1	1 (CoreXL is disabled)	1 (CoreXL is disabled)
2	2	2
4	3	1
6-20	Number of CPU cores, minus 2	2
More than 20	Number of CPU cores, minus 4. However, no more than 40. **Note** - This limit applies only to the Kernel Mode Firewall (KMFW).	4

The numbers of CoreXL Firewall instances start from zero.

The numbers of CPU cores start from the highest CPU ID allowed by the current Check Point license on your Security Gateway.

Refer to the **ID** and **CPU** columns in this example:

```
# fw ctl multik stat

ID | Active   | CPU    | Connections | Peak
-----------------------------------------------
 0 | Yes     | 7      |           5 |        21
 1 | Yes     | 6      |           3 |        23
 2 | Yes     | 5      |           5 |        25
 3 | Yes     | 4      |           4 |        21
 4 | Yes     | 3      |           5 |        21
 5 | Yes     | 2      |           5 |        20
```

```
# fw6 ctl multik stat

ID | Active   | CPU    | Connections | Peak
-----------------------------------------------
 0 | Yes     | 7      |           0 |         4
 1 | Yes     | 6      |           0 |         4
```

Maximal number of IPv4 CoreXL Firewall instances

Gaia kernel edition	Check Point Appliance	Open Server
64-bit	40	40

Notes:

- Starting in R80.20, the Gaia kernel edition is 64-bit only.
- The total number of IPv4 CoreXL Firewall instances and IPv6 CoreXL Firewall instances cannot exceed the numbers in the table above. This limit applies only to the Kernel Mode Firewall (KMFW).

Configuring IPv4 and IPv6 CoreXL Firewall instances

In This Section:

IPv4 and IPv6 CoreXL Firewall Instances ... 258

Configuring the Number of IPv4 CoreXL Firewall Instances ... 260

Configuring the Number of IPv6 CoreXL Firewall Instances ... 261

Example CoreXL Configuration ... 262

Important Notes for Cluster:

- You must configure the CoreXL in the same way on all the cluster members.
- If you enable CoreXL, disable CoreXL, or change the number of CoreXL Firewall instances, you should treat this change as a version upgrade. Schedule a full maintenance window and follow the instructions in the *R81 Installation and Upgrade Guide* - Chapter *Upgrading ClusterXL Deployments*. Perform either a *Minimal Effort Upgrade* procedure (requires downtime), or a *Zero Downtime Upgrade* procedure (no downtime, but active connections are lost). Instead of the version upgrade, configure the CoreXL on each cluster member.

IPv4 and IPv6 CoreXL Firewall Instances

After you enable Gaia IPv6 support on the Security Gateway (see *R81 Gaia Administration Guide*), configure the CPU cores to run different combinations of IPv4 and IPv6 CoreXL Firewall instances:

- The number of IPv4 CoreXL Firewall instances you can configure is from a minimum of two to a maximum equal to the total number of CPU cores on the Security Gateway:

```
2 <= (Number of IPv4 CoreXL Firewall instances) <= (Total
Number of CPU cores)
```

- By default, the number of IPv6 CoreXL Firewall instances is set to two.

  When the SMT (Hyper-Threading) is enabled, the default number of IPv6 CoreXL Firewall instances is four.

- The number of IPv6 CoreXL Firewall instances you can configure is from a minimum of two to a maximum equal to the total number of IPv4 CoreXL Firewall instances.

The number of IPv6 CoreXL Firewall instances cannot be greater than the number of IPv4 CoreXL Firewall instances:

```
2 <= (Number of IPv6 CoreXL Firewall instances) <= (Total
Number of IPv4 CoreXL Firewall instances)
```

- The total number of IPv4 *and* IPv6 CoreXL Firewall instances cannot be greater than forty:

Note - This limit applies only to the Kernel Mode Firewall (KMFW).

```
(Number of IPv4 CoreXL Firewall instances) + (Number of IPv6
CoreXL Firewall instances) <= 40
```

Configuring the Number of IPv4 CoreXL Firewall Instances

Step	Instructions
1	Connect to the command line on the Security Gateway.
2	Log in to Gaia Clish or Expert mode.
3	Run: `cpconfig`
4	Enter the number of the **Check Point CoreXL** option.
5	Enter **1** to select **Change the number of firewall instances**.
6	Enter the total number of IPv4 CoreXL Firewall instances you wish the Security Gateway to run. **Note** - You can only select a number from the range shown. Follow the instructions on the screen.
7	Exit from the `cpconfig` menu.
8	Reboot the Security Gateway.

Configuring the Number of IPv6 CoreXL Firewall Instances

Step	Instructions
1	Connect to the command line on the Security Gateway.
2	Log in to Gaia Clish or Expert mode.
3	Run: `cpconfig`
4	Enter the number of the **Check Point CoreXL** option.
5	Enter **2** to select **Change the number of IPv6 firewall instances**.
6	Enter the total number of IPv6 CoreXL Firewall instances you wish the Security Gateway to run. **Note** - You can only select a number from the range shown. Follow the instructions on the screen.
7	Exit from the `cpconfig` menu.
8	Reboot the Security Gateway.

Example CoreXL Configuration

Security Gateway has four CPU cores.

By default, there are three IPv4 CoreXL Firewall instances and two IPv6 CoreXL Firewall instances:

CPU Core	IPv4 CoreXL Firewall instances	IPv6 CoreXL Firewall instances
CPU 0	N / A	N / A
CPU 1	`fw4_2`	N / A
CPU 2	`fw4_1`	`fw6_1`
CPU 3	`fw4_0`	`fw6_0`

- IPv4 CoreXL Firewall instances: The minimum allowed number is two and the maximum is four
- IPv6 CoreXL Firewall instances: The minimum allowed number is two and the maximum is three

To increase the number of IPv6 CoreXL Firewall instances to four, first you must increase the number of IPv4 CoreXL Firewall instances to the maximum of four and reboot:

```
CoreXL is currently enabled with 3 IPv4 firewall instances and 2 IPv6 firewall instances.

(1) Change the number of firewall instances
(2) Change the number of IPv6 firewall instances
(3) Disable Check Point CoreXL

(4) Exit
Enter your choice (1-4) : 1

How many IPv4 firewall instances would you like to enable (2 to 4) [3] ? 4

CoreXL was enabled successfully with 4 firewall instances.
Important: This change will take effect after reboot.
```

After the reboot, the CoreXL configuration on the Security Gateway looks like this:

CPU Core	IPv4 CoreXL Firewall instances	IPv6 CoreXL Firewall instances
CPU 0	`fw4_3`	N / A
CPU 1	`fw4_2`	N / A
CPU 2	`fw4_1`	`fw6_1`
CPU 3	`fw4_0`	`fw6_0`

Increase the number of IPv6 CoreXL Firewall instances to four and reboot:

```
CoreXL is currently enabled with 4 IPv4 firewall instances and 2 IPv6 firewall instances.

(1) Change the number of firewall instances
(2) Change the number of IPv6 firewall instances
(3) Disable Check Point CoreXL

(4) Exit
Enter your choice (1-4) : 2

How many IPv6 firewall instances would you like to enable (2 to 4)[2] ? 4

CoreXL was enabled successfully with 3 IPv6 firewall instances.
Important: This change will take effect after reboot.
```

After the reboot, the CoreXL configuration on the Security Gateway looks like this:

CPU Core	IPv4 CoreXL Firewall instances	IPv6 CoreXL Firewall instances
CPU 0	fw4_3	fw6_3
CPU 1	fw4_2	fw6_2
CPU 2	fw4_1	fw6_1
CPU 3	fw4_0	fw6_0

CoreXL Limitations

- R81 CoreXL does not support:
  - Overlapping NAT
  - VPN Traditional Mode
- The global CoreXL Firewall instance #0 (`fw_worker_0`) always processes all the 6in4 traffic.

Configuring Affinity Settings

In This Section:

Introduction *265*

The $FWDIR/conf/fwaffinity.conf Configuration File *265*

The $FWDIR/scripts/fwaffinity_apply Script *268*

Introduction

The script `$FWDIR/scripts/fwaffinity_apply` on Security Gateway (Scalable Platform Security Group Members) executes automatically during boot and controls the affinity settings. When you make a change in the affinity settings, the changes do not take effect until you either reboot the Security Gateway (Scalable Platform Security Group), or manually execute the `$FWDIR/scripts/fwaffinity_apply` script.

The `$FWDIR/scripts/fwaffinity_apply` script configures the affinity of interfaces based on the settings in the `$FWDIR/conf/fwaffinity.conf` configuration file. To change these affinity settings, edit that configuration file.

The $FWDIR/conf/fwaffinity.conf Configuration File

The configuration file `$FWDIR/conf/fwaffinity.conf` controls CoreXL affinity settings.

Each line in this plain-text file uses the same format:

```
<Type> <ID> <CPU_ID>
```

Where:

Field	Allowed Value	Description
<Type>	**i**	Configures the affinity of an interface.
	n	Configures the affinity of a Check Point daemon.
	k	Configures the affinity of a CoreXL Firewall instance.
<ID>	Name of Interface	If **<type>** = **i**.
	Name of Daemon	If **<type>** = **n**.

Field	Allowed Value	Description
	ID of CoreXL Firewall instance	If **<type> = k**.
	default	Configures affinities for interfaces that are not specified other lines.
<CPU_ID>	Number (ID) of CPU core	Specifies the ID numbers of processing CPU cores, to which you affine an interface, a Check Point daemon, or a CoreXL Firewall instance.
	all	Specifies all processing CPU cores as available to configure the affinity of an interface, a Check Point daemon, or a CoreXL Firewall instance.
	auto	Configures Automatic mode. See *"Allocation of Processing CPU Cores" on page 270*.
	ignore	No specified affinity. This is useful to exclude an interface from the "**default**" configuration.

Notes:

- The default configuration in this file is:

  ```
  i default auto
  ```

- Possible combinations:

  - To configure the affinity for an interface:

    ```
    i <Name of Interface> {<CPU ID Number> | all | ignore |
    auto}
    i default {<CPU ID Number> | all | ignore | auto}
    ```

  - To configure the affinity of a Check Point daemon:

    ```
    n <Name of Daemon> {<CPU ID Number> | all | ignore |
    auto}
    ```

  - To configure the affinity of a CoreXL Firewall instance:

    ```
    k <ID of CoreXL Firewall instance> {<CPU ID Number> | all
    | ignore | auto}
    ```

- To view the IRQs of all interfaces, run:

  - On a Security Gateway (each Cluster Member), run in Gaia Clish or the Expert mode:

    ```
    fw ctl affinity -l -v -a
    ```

  - On a Scalable Platform Security Group, run in Gaia gClish:

    ```
    fw ctl affinity -l -v -a
    ```

  - On a Scalable Platform Security Group, run in the Expert mode:

    ```
    g_fw ctl affinity -l -v -a
    ```

  See *"fw ctl affinity" on page 317*.

- Interfaces that share an IRQ cannot have different CPU cores as their affinities.

  This also applies when one interface is included in the **default** affinity setting.

  You must either configure the same affinity of all interfaces, or use **ignore** for one of these interfaces.

- On a Scalable Platform Security Group, after you edit the `$FWDIR/conf/fwaffinity.conf` file, you must copy it to all Security Group Members:

  ```
  asg_cp2blades $FWDIR/conf/fwaffinity.conf
  ```

The $FWDIR/scripts/fwaffinity_apply Script

Syntax

- To execute this shell script on a Security Gateway (each Cluster Member), run in the Expert mode:

```
$FWDIR/scripts/fwaffinity_apply <Parameter>
```

- To execute this shell script on a Scalable Platform Security Group, run in the Expert mode:

```
g_all $FWDIR/scripts/fwaffinity_apply <Parameter>
```

Parameters

Parameter	Description
`-q`	Quiet mode - prints only error messages (standard output goes to `/dev/null`).

`-t i`
`-t n`
`-t k`
	Applies affinity only for the specified type:
	- `-t i` - For interfaces
	- `-t n` - For Check Point daemons
	- `-t k` - For CoreXL Firewall instances

Performance Tuning

This section describes how to fine tune the CoreXL performance.

Allocation of Processing CPU Cores

The CoreXL software architecture includes the Secure Network Distributor (SND).

The SND is responsible for these:

- Processing the incoming traffic from the network interfaces.
- Accelerating authorized packets (when SecureXL is enabled).
- Distributing non-accelerated packets between the CoreXL Firewall instances.

The association of a specific interface with a specific processing CPU core is called the interface's *affinity* with that CPU core. This affinity causes the interface's traffic to be directed to that CPU core and the CoreXL SND to run on that CPU core.

The association of a specific CoreXL Firewall instance with a specific CPU core is called the CoreXL Firewall instance's *affinity* with that CPU core.

The association of a specific user space process with a specific CPU core is called the process's *affinity* with that CPU core.

The default affinity setting for all interfaces is Automatic. Automatic affinity means that if SecureXL is enabled, the affinity for each interface is changed at specific intervals and balanced between the available CPU cores. If SecureXL is disabled, the default affinities of all interfaces are with one available CPU core. In both cases, all processing CPU cores that run a CoreXL Firewall instance, or defined as the affinity for a different user space process, is considered unavailable, and the affinity for interfaces is not set to those CPU cores.

In some cases, which we discuss in the sections below, it can be necessary to change the distribution of CoreXL Firewall instances, the CoreXL SND, and other user space processes, between the processing CPU cores. To do so, you change the affinities of different NICs (interfaces) or user space processes. To make sure CoreXL operates at an efficient level, traffic from all interfaces must be directed to CPU cores that do not run CoreXL Firewall instances. Therefore, if you change affinities of interfaces or other user space processes, you must configure the corresponding number of CoreXL Firewall instances. In addition, you must make sure that the CoreXL Firewall instances run on other processing CPU cores.

Usually, we do not recommend for a CoreXL SND and a CoreXL Firewall instance to use the same CPU core. It is necessary for the CoreXL SND and a CoreXL Firewall instance to use a CPU core when Security Gateway runs on a platform with only two CPU cores.

Adding Processing CPU Cores to the Hardware

If you increase the number of processing CPU cores on the computer, it does **not** automatically increase the number of CoreXL Firewall instances.

You must manually configure the applicable number of CoreXL Firewall instances in the `cpconfig` menu (see *"Configuring IPv4 and IPv6 CoreXL Firewall instances" on page 258*).

Important Notes for Cluster:

- You must configure the CoreXL in the same way on all the cluster members.
- If you enable CoreXL, disable CoreXL, or change the number of CoreXL Firewall instances, you should treat this change as a version upgrade.
  Schedule a full maintenance window and follow the instructions in the *R81 Installation and Upgrade Guide* - Chapter *Upgrading ClusterXL Deployments*.
  Perform either a *Minimal Effort Upgrade* procedure (requires downtime), or a *Zero Downtime Upgrade* procedure (no downtime, but active connections are lost). Instead of the version upgrade, configure the CoreXL on each cluster member.

Allocating Additional CPU Cores to the CoreXL SND

The default configuration of CoreXL Firewall instances and the CoreXL SND instances might not be optimal for your needs.

If the default number of CoreXL SND instances is not enough to process the incoming traffic, and your Security Gateway computer contains enough CPU cores, you can decrease the number of CoreXL Firewall instances. This automatically allocates additional CPU cores to run the CoreXL SND instances.

This scenario is likely to occur if much of the traffic is accelerated by SecureXL. In this case, the task load of the CoreXL SND instances may be disproportionate to that of the CoreXL Firewall instances.

To check if the SND is slowing down the traffic:

Step	Instructions
1	Identify the processing CPU core, to which the interfaces direct their traffic: `fw ctl affinity -l -r`
2	Under heavy traffic conditions, run the `top` command. Examine the values for the different CPU cores in the `idle` column.

Best Practice - We recommend to allocate an additional CPU core to the CoreXL SND only if *all* these conditions are met:

- There are at least 8 processing CPU cores.
- In the output of the `top` command, the `idle` values for the CPU cores that run the CoreXL SND instances are in the 0%-5% range.
- In the output of the `top` command, the sum of the `idle` values for the CPU cores that run the CoreXL Firewall instances is significantly higher than 100%.

If at least one of the above conditions is not met, the default CoreXL configuration is sufficient.

To allocate an additional processing CPU core to the CoreXL SND:

Item	Description
1	Decrease the number of CoreXL Firewall instances in the `cpconfig` menu. See *"Configuring IPv4 and IPv6 CoreXL Firewall instances" on page 258*.
2	Configure interface affinities to the remaining CPU cores. See *"Configuring Affinities for Interfaces" on page 275*.
3	Reboot to apply the new configuration.

Allocating a CPU Core for Heavy Logging

If the Security Gateway generates very large number of logs, it may be advisable to allocate a processing CPU core to the **fwd** daemon, which generates the logs.

Note - This change decreases the number of CPU cores available for CoreXL Firewall instances.

Important Notes for Cluster:

- You must configure the CoreXL in the same way on all the cluster members.
- If you enable CoreXL, disable CoreXL, or change the number of CoreXL Firewall instances, you should treat this change as a version upgrade. Schedule a full maintenance window and follow the instructions in the *R81 Installation and Upgrade Guide* - Chapter *Upgrading ClusterXL Deployments*. Perform either a *Minimal Effort Upgrade* procedure (requires downtime), or a *Zero Downtime Upgrade* procedure (no downtime, but active connections are lost). Instead of the version upgrade, configure the CoreXL on each cluster member.

To allocate a processing CPU core to the *fwd* daemon:

See *"Configuring Affinity Settings" on page 265*.

Step	Instructions
1	Connect to the command line on Security Gateway (each Cluster Member).
2	Log in to the Expert mode.
3	Run: `cpconfig`
4	Enter the number of the **Check Point CoreXL** option.
5	Decrease the number of CoreXL Firewall instances. See *"Configuring IPv4 and IPv6 CoreXL Firewall instances" on page 258*.
6	Exit from the `cpconfig` menu.
7	Examine which processing CPU cores run the CoreXL Firewall instances and which CPU cores handle the traffic from interfaces: `fw ctl affinity -l -r` See *"fw ctl affinity" on page 317*.
8	Back up the `$FWDIR/conf/fwaffinity.conf` file: `$FWDIR/conf/fwaffinity.conf{,_BKP}`

Step	Instructions
9	Edit the `$FWDIR/conf/fwaffinity.conf` file: `vi $FWDIR/conf/fwaffinity.conf`

10	Allocate one of the remaining CPU cores to the **fwd** daemon.
	To do so, configure the affinity of the **fwd** daemon to that CPU core.
	`n fwd <CPU ID>`
	For example, to affine the **fwd** daemon to CPU core **#2**, add this line:
	`n fwd 2`
	Note - It is important to avoid the CPU cores that run the CoreXL SND instances only if these CPU cores are explicitly defined for the affinities of interfaces. If affinity of interfaces is configured in the Automatic mode, the **fwd** daemon can use all CPU cores that do not run CoreXL Firewall instances. Traffic from interfaces is automatically diverted to other CPU cores.

11	Save the changes in the file and exit the editor.
12	Apply the new configuration: ■ To apply immediately, run: `$FWDIR/scripts/fwaffinity_apply` ■ To apply later, reboot the Security Gateway (each Cluster Member).

Configuring Affinities for Interfaces

The association of a specific interface with a specific processing CPU core is called the interface's *affinity* with that CPU core. This affinity causes the interface's traffic to be directed to that CPU core and the CoreXL SND to run on that CPU core.

Security Gateway loads (Scalable Platform Security Group Members load) affinities for interfaces during the boot from the CoreXL configuration file `$FWDIR/conf/fwaffinity.conf`. In this configuration file, lines that begin with the letter "`i`", define the affinities for interfaces.

Workflow:

Step	Instructions

1	Check which processing CPU cores run the CoreXL Firewall instances and which CPU cores handle the traffic from interfaces:
	■ On a Security Gateway (each Cluster Member), run in Gaia Clish or the Expert mode:
	`fw ctl affinity -l -r`
	■ On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl affinity -l -r`
	■ On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl affinity -l -r`
	See *"fw ctl affinity" on page 317*.

2	Allocate the remaining CPU cores to run the CoreXL SND instances. To do so, configure the affinity of interfaces to the applicable CPU cores. For more information, see *"Allocation of Processing CPU Cores" on page 270*. **Notes:** ■ To set the affinity of VLAN interfaces, use their physical interfaces. ■ If you allocate more than one processing CPU core to the CoreXL SND, it is necessary to configure affinities for interfaces explicitly to the remaining CPU cores. If you have multiple interfaces, decide which interfaces to affine to which CPU cores. Try to achieve a balance of expected traffic between the CPU cores. Examine the traffic balance with the `top` command.

Configuring affinities for interfaces explicitly:

Step	Instructions
1	Connect to the command line on the Security Gateway (each Cluster Member / Scalable Platform Security Group).
2	Log in to the Expert mode.

3	Configure the affinity of each interface in the `$FWDIR/conf/fwaffinity.conf` file.
	See *"Configuring Affinity Settings" on page 265*.
	For each interface, there must be a separate line that begins with the letter "`i`".
	Each of these lines must have this syntax:
	`i <Name of Interface> <CPU ID>`
	For example, if it is necessary that the traffic from `eth0` and `eth1` (`eth1-05` and `eth1-07`) goes to CPU core `#0`, and the traffic from `eth2` (`eth1-09`) goes to CPU core `#1`, add these lines:
	■ On the Security Gateway (each Cluster Member):
	`i eth0 0`
	`i eth1 0`
	`i eth2 1`
	■ On the Scalable Platform Security Group:
	`i eth1-05 0`
	`i eth1-07 0`
	`i eth1-09 1`

Step	Instructions

	Alternatively, you can choose to configure affinities for interface explicitly for only one processing CPU core, and define other CPU cores as the default affinity of the remaining interfaces.
	`i default <CPU ID>`
	For example, if it is necessary that the traffic from `eth2` (`eth1-05`) goes to CPU core `#1`, and the traffic from all other interfaces goes to CPU core `#0`, add these lines:
	- On the Security Gateway (each Cluster Member):
	`i eth2 1`
	`i default 0`
	- On the Scalable Platform Security Group:
	`i eth1-05 1`
	`i default 0`

4	Load the new configuration.
	- To load it immediately:
	• On the Security Gateway (each Cluster Member), run:
	`$FWDIR/scripts/fwaffinity_apply`
	• On the Scalable Platform Security Group, run:
	`g_all $FWDIR/scripts/fwaffinity_apply`
	- To load it later, reboot.
	• On the Security Gateway (each Cluster Member), run:
	`reboot`
	• On the Scalable Platform Security Group, run:
	`g_reboot -a`

Best Practice - If you allocate only one CPU core to the CoreXL SND, it is best to have that CPU core selected automatically. To do so, leave the default automatic interface affinity and do not configure explicit affinities for interfaces to CPU cores. Make sure the `$FWDIR/conf/fwaffinity.conf` file contains this line:

```
i default auto
```

Make sure that the `$FWDIR/conf/fwaffinity.conf` file does not contain other lines that begin with "`i`", so that there are no explicit affinities for interfaces configured. This makes sure that Security Gateway directs (Scalable Platform Security Group Members direct) all traffic to the remaining CPU cores.

Best Practice - In addition, see *"Multi-Queue" on page 351*.

Dynamic Balancing of CoreXL Instances

Introduction

On Check Point Appliances, R80.40 added the ability to change the number of CoreXL Firewall and SND instances without reboot (Dynamic Balancing).

Important:

- By default, this feature is *enabled*.
- We do **not** recommend manual configuration of CoreXL Firewall and SND instances, because such configuration *disables* the CoreXL Dynamic Balancing.
  To enable the CoreXL Dynamic Balancing again, you must disable it and enable it.
- For CoreXL Dynamic Balancing requirements, see sk164155.

When CoreXL Dynamic Balancing is enabled, Security Gateway monitors the average CPU utilization of CoreXL Firewall and SND instances and automatically increases or decreases the number of CoreXL Firewall instances.

The Dynamic Balancing Daemon (*dsd*) has three stages in each iteration:

1. Examine the current CPU utilization.

2. Decide if and what changes to make based on the current CPU utilization.

3. If needed, change the current CoreXL configuration in one of these ways:

   - Add a CoreXL Firewall instance.

     This change is possible only under these conditions:

     a. Average difference in CPU utilization between CoreXL Firewall and SND instances is greater than 10%.

     b. The current number of CoreXL Firewall instances is less than it was during the boot.

- Add a CoreXL SND instance.

  This change stops a CoreXL Firewall instance and moves it to another CPU core.

  This change is possible only under these conditions:

  a. Average difference in CPU utilization between CoreXL Firewall and SND instances is greater than 10%.

  b. CoreXL Firewall instances consume the CPU cores at less than 40%.

  c. There is an available CPU core.

Syntax

Important:

- There are commands for Gaia Clish and for the Expert mode.
- In a Cluster, you must configure all the Cluster Members in the same way.

Enabling the feature

- In Gaia Clish:

```
set dynamic-balancing state enable
reboot
```

- In the Expert mode:

```
dynamic_balancing -o enable
reboot
```

Important:

- After you enable this feature for the first time, the Security Gateway may require a reboot in these cases:
  - The current CoreXL configuration is not the default
  - More CoreXL SND instances are required for the current CPU load
- After the boot, you can stop, start, and restart this feature without a reboot.

Stopping the feature

This command lets you stop the CoreXL Dynamic Balancing ("freeze" it).

- In Gaia Clish:

```
set dynamic-balancing state stop
```

- In the Expert mode:

```
dynamic_balancing -o stop
```

Important:

- When you stop this feature, the Security Gateway uses the last CoreXL Balancing configuration.
- This change does **not** require a reboot.
- This change survives the reboot.
- The status of the CoreXL Dynamic Balancing shows as "off".

Starting the feature

This command lets you start the CoreXL Dynamic Balancing after it was stopped.

- In Gaia Clish:

```
set dynamic-balancing state start
```

- In the Expert mode:

```
dynamic_balancing -o start
```

Important:

- When you start this feature, the Security Gateway continues to change the CoreXL Balancing configuration automatically based on the CPU utilization.
- This change does **not** require a reboot.
- This change survives the reboot.

Resetting the feature

This command lets you reset the CoreXL configuration to the default and keep the CoreXL Dynamic Balancing enabled.

This command is equivalent to the "`disable`" command followed by the "`enable`" command.

- In Gaia Clish:

```
set dynamic-balancing state reset
```

- In the Expert mode:

```
dynamic_balancing -r
```

Important:

- After this feature restarts, the CoreXL configuration returns to the default (see *"Default Configuration of CoreXL" on page 256*).
- This change does **not** require a reboot.

Disabling the feature

- In Gaia Clish:

```
set dynamic-balancing state disable
```

- In the Expert mode:

```
dynamic_balancing -o disable
```

Important:

- When you disable this feature, the CoreXL configuration returns to the default (see *"Default Configuration of CoreXL" on page 256*).
- After you disable this feature, the Security Gateway requires a reboot.
  The command shows the applicable message.

Monitoring

- You can monitor the *status* of the CoreXL Dynamic Balancing with CLI commands:
  - In Gaia Clish:

    ```
    show dynamic-balancing state
    ```

  - In the Expert mode:

    ```
    dynamic_balancing -p
    ```

- You can monitor the *status* of the CoreXL Dynamic Balancing in the CPView tool:

  Procedure

  1. Connect to the command line on the Security Gateway.
  2. Run:

     ```
     cpview
     ```

  3. From the top, click:

     SysInfo

  4. Examine this field:

     DS Status

     - **On** - Means the CoreXL Dynamic Balancing is enabled
     - **Off** - Means the CoreXL Dynamic Balancing is disabled

- You can monitor the *performance* of the CoreXL Dynamic Balancing in the CPView tool:

  Procedure

  1. Connect to the command line on the Security Gateway.
  2. Run:

     ```
     cpview
     ```

  3. From the top, click:

     CPU > **Overview** > **Host**

4. Examine these sections:
    - **Overview** - Shows the current number of CoreXL instances and the average CPU utilization
    - **CPU** - Shows the CPU cores, the CoreXL instance types they run, and the CPU utilization in different categories

- You can monitor the CoreXL Firewall instances with this command:

```
fw ctl multik stat
```

- You can monitor the CoreXL Affinity with this command:

```
fw ctl affinity -l -r -a
```

- You can examine these log files:
    - When the CoreXL Dynamic Balancing changes the CoreXL configuration, it writes the applicable entries in the `$FWDIR/log/dsd.elg` file.
    - When the CoreXL Dynamic Balancing starts, it writes the applicable entries in the `$FWDIR/log/dynamic_balancing.log` file.

CoreXL Commands

This section describes different CLI commands CoreXL.

Syntax Legend

Whenever possible, this guide lists commands, parameters and options in the alphabetical order.

This guide uses this convention in the Command Line Interface (CLI) syntax:

<table>
<tr><th>Character</th><th>Description</th></tr>
<tr><td>TAB</td><td>Shows the available nested subcommands:
<pre>
main command
→ nested subcommand 1
→ → nested subsubcommand 1-1
→ → nested subsubcommand 1-2
→ nested subcommand 2
</pre>
Example:
<pre>
cpwd_admin
    config
        -a <options>
        -d <options>
        -p
        -r
    del <options>
</pre>
Meaning, you can run only <b>one</b> of these commands:
<ul>
<li>This command:<pre>cpwd_admin config -a <options></pre></li>
<li>Or this command:<pre>cpwd_admin config -d <options></pre></li>
<li>Or this command:<pre>cpwd_admin config -p</pre></li>
<li>Or this command:<pre>cpwd_admin config -r</pre></li>
<li>Or this command:<pre>cpwd_admin del <options></pre></li>
</ul>
</td></tr>
<tr><td>Curly brackets or braces
<b>{ }</b></td><td>Enclose a list of available commands or parameters, separated by the vertical bar <b>|</b>.
User can enter only one of the available commands or parameters.</td></tr>
</table>

Character	Description
Angle brackets **< >**	Enclose a variable. User must explicitly specify a supported value.
Square brackets or brackets **[ ]**	Enclose an optional command or parameter, which user can also enter.

cp_conf corexl

Description

Enables or disables CoreXL.

Important:

- This command is for Check Point use only.
  To configure CoreXL, use the **Check Point CoreXL** option in the *"cpconfig" on page 289* menu.
- After all changes in CoreXL configuration on the Security Gateway, you must reboot it.
- In Custer, you must configure all the Cluster Members in the same way.

Syntax

- To enable CoreXL with '`n`' IPv4 Firewall instances and optionally '`k`' IPv6 Firewall instances:

```
cp_conf corexl [-v] enable [n] [-6 k]
```

- To disable CoreXL:

```
cp_conf corexl [-v] disable
```

The related command is: *"fwboot corexl" on page 338*.

Parameters

Parameter	Description
`-v`	Leaves the high memory (`vmalloc`) unchanged.
`n`	Denotes the number of IPv4 CoreXL Firewall instances.
`k`	Denotes the number of IPv6 CoreXL Firewall instances.

Example

Currently, the Security Gateway runs two IP4v CoreXL Firewall instances (`KERN_INSTANCE_NUM = 2`).

We change the number of IP4v CoreXL Firewall instances to three.

```
[Expert@MyGW:0]# fw ctl multik stat
ID | Active  | CPU    | Connections | Peak
----------------------------------------------
 0 | Yes     | 2      |           7 |       28
 1 | Yes     | 1      |           0 |       11
[Expert@MyGW:0]#
[Expert@MyGW:0]# cat /etc/fw.boot/boot.conf
CTL_IPFORWARDING 1
DEFAULT_FILTER_PATH 0
KERN_INSTANCE_NUM 2
COREXL_INSTALLED 1
KERN6_INSTANCE_NUM 2
IPV6_INSTALLED 0
CORE_OVERRIDE 4
[Expert@MyGW:0]#
[Expert@MyGW:0]# cp_conf corexl -v enable 3
[Expert@MyGW:0]#
[Expert@MyGW:0]# cat /etc/fw.boot/boot.conf
CTL_IPFORWARDING 1
DEFAULT_FILTER_PATH 0
KERN_INSTANCE_NUM 3
COREXL_INSTALLED 1
KERN6_INSTANCE_NUM 2
IPV6_INSTALLED 0
CORE_OVERRIDE 4
[Expert@MyGW:0]#
[Expert@MyGW:0]# reboot
.. ... ...
[Expert@MyGW:0]# fw ctl multik stat
ID | Active  | CPU    | Connections | Peak
----------------------------------------------
 0 | Yes     | 3      |           7 |       28
 1 | Yes     | 2      |           0 |       11
 2 | Yes     | 1      |           4 |       10
[Expert@MyGW:0]#
```

cpconfig

Description

This command starts the Check Point Configuration Tool.

This tool configures specific settings for the installed Check Point products.

Important - In a Cluster, you must configure all the Cluster Members in the same way.

Syntax

```
cpconfig
```

Menu Options

Note - The options shown depend on the configuration and installed products.

Menu Option	Description
Licenses and contracts	Manages Check Point licenses and contracts on this Security Gateway or Cluster Member.
SNMP Extension	Obsolete. Do **not** use this option anymore. To configure SNMP, see the *R81 Gaia Administration Guide* - Chapter *System Management* - Section *SNMP*.
PKCS#11 Token	Register a cryptographic token, for use by Gaia Operating System. See details of the token, and test its functionality.
Random Pool	Configures the RSA keys, to be used by Gaia Operating System.
Secure Internal Communication	Manages SIC on the Security Gateway or Cluster Member. This change requires a restart of Check Point services on the Security Gateway or Cluster Member. For more information, see: ■ The *R81 Security Management Administration Guide*. ■ sk65764: How to reset SIC.

Menu Option	Description
Enable cluster membership for this gateway	Enables the cluster membership on the Security Gateway. This change requires a reboot of the Security Gateway. For more information, see the: ■ *R81 Installation and Upgrade Guide*. ■ *R81 ClusterXL Administration Guide*.
Disable cluster membership for this gateway	Disables the cluster membership on the Security Gateway. This change requires a reboot of the Security Gateway. For more information, see the: ■ *R81 Installation and Upgrade Guide*. ■ *R81 ClusterXL Administration Guide*.
Enable Check Point Per Virtual System State	Enables Virtual System Load Sharing on the VSX Cluster Member. For more information, see the *R81 VSX Administration Guide*.
Disable Check Point Per Virtual System State	Disables Virtual System Load Sharing on the VSX Cluster Member. For more information, see the *R81 VSX Administration Guide*.
Enable Check Point ClusterXL for Bridge Active/Standby	Enables Check Point ClusterXL for Bridge mode. This change requires a reboot of the Cluster Member. For more information, see the: ■ *R81 Installation and Upgrade Guide*. ■ *R81 ClusterXL Administration Guide*.
Disable Check Point ClusterXL for Bridge Active/Standby	Disables Check Point ClusterXL for Bridge mode. This change requires a reboot of the Cluster Member. For more information, see the: ■ *R81 Installation and Upgrade Guide*. ■ *R81 ClusterXL Administration Guide*.
Check Point CoreXL	Manages CoreXL on the Security Gateway or Cluster Member. After all changes in CoreXL configuration, you must reboot the Security Gateway or Cluster Member. For more information, see *"CoreXL" on page 254*.

Menu Option	Description
Automatic start of Check Point Products	Shows and controls which of the installed Check Point products start automatically during boot.
Exit	Exits from the Check Point Configuration Tool.

Example 1 - Menu on a single Security Gateway

```
[Expert@MySingleGW:0]# cpconfig
This program will let you re-configure
your Check Point products configuration.


Configuration Options:
----------------------
(1) Licenses and contracts
(2) SNMP Extension
(3) PKCS#11 Token
(4) Random Pool
(5) Secure Internal Communication
(6) Enable cluster membership for this gateway
(7) Check Point CoreXL
(8) Automatic start of Check Point Products

(9) Exit

Enter your choice (1-9) :
```

Example 2 - Menu on a Cluster Member

```
[Expert@MyClusterMember:0]# cpconfig
This program will let you re-configure
your Check Point products configuration.


Configuration Options:
----------------------
(1) Licenses and contracts
(2) SNMP Extension
(3) PKCS#11 Token
(4) Random Pool
(5) Secure Internal Communication
(6) Disable cluster membership for this gateway
(7) Enable Check Point Per Virtual System State
(8) Enable Check Point ClusterXL for Bridge Active/Standby
(9) Check Point CoreXL
(10) Automatic start of Check Point Products

(11) Exit

Enter your choice (1-11) :
```

cpview

Overview of CPView

Description

CPView is a text based built-in utility on a Check Point computer.

CPView Utility shows statistical data that contain both general system information (CPU, Memory, Disk space) and information for different Software Blades (only on Security Gateway).

The CPView continuously updates the data in easy to access views.

On Security Gateway, you can use this statistical data to monitor the performance.

For more information, see sk101878.

Syntax

```
cpview --help
```

CPView User Interface

The CPView user interface has three sections:

Section	Description
Header	This view shows the time the statistics in the third view are collected. It updates when you refresh the statistics.
Navigation	This menu bar is interactive. Move between menus with the arrow keys and mouse. A menu can have sub-menus and they show under the menu bar.
View	This view shows the statistics collected in that view. These statistics update at the refresh rate.

Using CPView

Use these keys to navigate the CPView:

Key	Description
Arrow keys	Moves between menus and views. Scrolls in a view.
Home	Returns to the **Overview** view.
Enter	Changes to the **View Mode**. On a menu with sub-menus, the **Enter** key moves you to the lowest level sub-menu.
Esc	Returns to the **Menu Mode**.
Q	Quits CPView.

Use these keys to change CPView interface options:

Key	Description
R	Opens a window where you can change the refresh rate. The default refresh rate is 2 seconds.
W	Changes between wide and normal display modes. In wide mode, CPView fits the screen horizontally.
S	Manually sets the number of rows or columns.
M	Switches on/off the mouse.
P	Pauses and resumes the collection of statistics.

Use these keys to save statistics, show help, and refresh statistics:

Key	Description
C	Saves the current page to a file. The file name format is: `cpview_<ID of the cpview process>.cap<Number of the capture>`
H	Shows a tooltip with CPView options.
Space bar	Immediately refreshes the statistics.

fw ctl multik

Description

The *fw ctl multik* and *fw6 ctl multik* commands control CoreXL for IPv4 and IPv6, respectively.

Syntax for IPv4

```
fw ctl multik
      add_bypass_port <options>
      del_bypass_port <options>
      dynamic_dispatching <options>
      gconn <options>
      get_instance <options>
      print_heavy_conn
      prioq <options>
      show_bypass_ports
      stat
      start
      stop
      utilize
```

Syntax for IPv6

```
fw6 ctl multik
      add_bypass_port <options>
      del_bypass_port <options>
      dynamic_dispatching <options>
      gconn <options>
      get_instance <options>
      print_heavy_conn
      prioq <options>
      show_bypass_ports
      stat
      start
      stop
      utilize
```

Parameters

Parameter	Description
`add_bypass_port <options>`	Adds the specified TCP and UDP ports to the CoreXL Dynamic Dispatcher bypass list. See *"fw ctl multik add_bypass_port" on page 296*.
`del_bypass_port <options>`	Removes the specified TCP and UDP ports from the CoreXL Dynamic Dispatcher bypass list. See *"fw ctl multik del_bypass_port" on page 298*.
`dynamic_dispatching <options>`	Shows and controls CoreXL Dynamic Dispatcher (see sk105261). See *"fw ctl multik dynamic_dispatching" on page 300*.
`gconn <options>`	Shows statistics about CoreXL Global Connections. See *"fw ctl multik gconn" on page 301*.
`get_instance <options>`	Shows CoreXL Firewall instance that processes the specified IPv4 connection. See *"fw ctl multik get_instance" on page 306*.
`print_heavy_conn`	Shows the table with Heavy Connections (that consume the most CPU resources) in the CoreXL Dynamic Dispatcher. See *"fw ctl multik print_heavy_conn" on page 308*.
`prioq <options>`	Configures the CoreXL Firewall Priority Queues (see sk105762). See *"fw ctl multik prioq" on page 310*.
`show_bypass_ports`	Shows the TCP and UDP ports configured in the bypass port list of the CoreXL Dynamic Dispatcher. See *"fw ctl multik show_bypass_ports" on page 311*.
`stat`	Shows the CoreXL status. See *"fw ctl multik stat" on page 312*.
`start`	Starts all CoreXL Firewall instances on-the-fly. See *"fw ctl multik start" on page 314*.
`stop`	Stops all CoreXL Firewall instances temporarily. See *"fw ctl multik stop" on page 315*.
`utilize`	Shows the CoreXL queue utilization for each CoreXL Firewall instance. See *"fw ctl multik utilize" on page 316*.

fw ctl multik add_bypass_port

Description

Adds the specified TCP and UDP ports to the bypass port list of the CoreXL Dynamic Dispatcher.

For more information about the CoreXL Dynamic Dispatcher, see sk105261.

Important - This command saves the configuration in the `$FWDIR/conf/dispatcher_bypass.conf` file. You must **not** edit this file manually.

Syntax

```
fw ctl multik add_bypass_port <Port Number 1>,<Port Number
2>,...,<Port Number N>
```

Parameters

Parameter	Description
<Port Number>	Specifies the numbers of TCP and UDP ports to add to the list. **Important** - You can add 10 ports maximum.

Example

```
[Expert@MyGW:0]# fw ctl multik show_bypass_ports
dynamic dispatcher bypass port list:
[Expert@MyGW:0]#
[Expert@MyGW:0]# cat $FWDIR/conf/dispatcher_bypass.conf
dynamic_dispatcher_bypass_ports_number = 0
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik add_bypass_port 8888
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik show_bypass_ports
dynamic dispatcher bypass port list:
(8888)
[Expert@MyGW:0]
[Expert@MyGW:0]# cat $FWDIR/conf/dispatcher_bypass.conf
dynamic_dispatcher_bypass_ports_number = 1
dynamic_dispatcher_bypass_port_table=8888
[Expert@MyGW:0]
[Expert@MyGW:0]# fw ctl multik add_bypass_port 9999
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik show_bypass_ports
dynamic dispatcher bypass port list:
(8888,9999)
[Expert@MyGW:0]
[Expert@MyGW:0]# cat $FWDIR/conf/dispatcher_bypass.conf
dynamic_dispatcher_bypass_ports_number = 2
dynamic_dispatcher_bypass_port_table=8888,9999
[Expert@MyGW:0]
```

fw ctl multik del_bypass_port

Description

Removes the specified TCP and UDP ports from the bypass port list of the CoreXL Dynamic Dispatcher.

For more information about the CoreXL Dynamic Dispatcher, see sk105261.

Important - This command saves the configuration in the `$FWDIR/conf/dispatcher_bypass.conf` file. You must **not** edit this file manually.

Syntax

```
fw ctl multik del_bypass_port <Port Number 1>,<Port Number 2>,...,<Port Number N>
```

Parameters

Parameter	Description
<Port Number>	Specifies the numbers of TCP and UDP ports to remove from the list.

Example

```
[Expert@MyGW:0]# fw ctl multik show_bypass_ports
dynamic dispatcher bypass port list:
[Expert@MyGW:0]#
[Expert@MyGW:0]# cat $FWDIR/conf/dispatcher_bypass.conf
dynamic_dispatcher_bypass_ports_number = 0
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik add_bypass_port 8888
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik show_bypass_ports
dynamic dispatcher bypass port list:
(8888)
[Expert@MyGW:0]
[Expert@MyGW:0]# cat $FWDIR/conf/dispatcher_bypass.conf
dynamic_dispatcher_bypass_ports_number = 1
dynamic_dispatcher_bypass_port_table=8888
[Expert@MyGW:0]
[Expert@MyGW:0]# fw ctl multik add_bypass_port 9999
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik show_bypass_ports
dynamic dispatcher bypass port list:
(8888,9999)
[Expert@MyGW:0]
[Expert@MyGW:0]# cat $FWDIR/conf/dispatcher_bypass.conf
dynamic_dispatcher_bypass_ports_number = 2
dynamic_dispatcher_bypass_port_table=8888,9999
[Expert@MyGW:0]
[Expert@MyGW:0]# fw ctl multik add_bypass_port 9999
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik show_bypass_ports
dynamic dispatcher bypass port list:
(8888)
[Expert@MyGW:0]
[Expert@MyGW:0]# cat $FWDIR/conf/dispatcher_bypass.conf
dynamic_dispatcher_bypass_ports_number = 1
dynamic_dispatcher_bypass_port_table=8888
[Expert@MyGW:0]
```

fw ctl multik dynamic_dispatching

Description

Shows and controls the CoreXL Dynamic Dispatcher that dynamically assigns new connections to a CoreXL Firewall instances based on the utilization of CPU cores.

For more information, see sk105261.

Syntax for IPv4

```
fw ctl multik dynamic_dispatching
      get_mode
      off
      on
```

Syntax for IPv6

```
fw6 ctl multik dynamic_dispatching
      get_mode
      off
      on
```

Parameters

Parameter	Description
get_mode	Shows the current state of the CoreXL Dynamic Dispatcher.
off	Disables the CoreXL Dynamic Dispatcher.
on	Enables the CoreXL Dynamic Dispatcher.

Example

```
[Expert@MyGW:0]# fw ctl multik dynamic_dispatching get_mode
Current mode is Off
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik dynamic_dispatching on
New mode is: On
Please reboot the system
[Expert@MyGW:0]#
```

fw ctl multik gconn

Description

Shows statistics about CoreXL Global Connections that Security Gateway stores in the kernel table `fw_multik_ld_gconn_table`.

The CoreXL Global Connections table contains information about which CoreXL Firewall instance owns which connections.

Notes:

- This command does not support VSX.
- This command does not support IPv6.

Syntax

```
fw [-d] ctl multik gconn
      -h
      -p
      -sec
      -seg <Number>
```

Parameters

Parameter	Description
`-d`	Runs the command in debug mode. Use only if you troubleshoot the command itself. **Best Practice** - If you use this parameter, then redirect the output to a file, or use the `script` command to save the entire CLI session.
none	Shows the interactive menu for the CoreXL Firewall Priority Queues.
`-h`	Shows the built-in help.

Parameter	Description
`-p`	Shows the additional information about each CoreXL Firewall instance, including the information about Firewall Priority Queues: ■ `I/O` (In or Out) ■ `Inst. ID` (CoreXL Firewall instance ID) ■ `Flags` ■ `Seq` (Sequence) ■ `Hold_ref` (Hold reference) ■ `Prio` (Firewall Priority Queues mode) ■ `last_enq_jiff` (Jiffies since last enqueue) ■ `queue_indx` (Queue index number) ■ `conn_tokens` (Connection Tokens)
`-s`	Shows the total number of global connections.
`-sec`	Shows the additional information about each CoreXL Firewall instance: ■ `I/O` (In or Out) ■ `Inst. ID` (CoreXL Firewall instance ID) ■ `Flags` ■ `Seq` (Sequence) ■ `Hold_ref` (Hold reference)
`-seg` *`<Number>`*	Shows the default information about the specified Global Connections Segment.

Example 1 - Default information

```
[Expert@MyGW:0]# fw ctl multik gconn
Default:

===============================================================================================
===========================
| Segm | Src IP | S.port | Dst IP | D.port | Proto | Flags | PP |Ref Cnt(I/O)|Inst|PPAK ID|clstr
mem ID|Rec. ref|Rec. Type|

===============================================================================================
===========================
|  0  | 192.168.3.52    | 18192 | 192.168.3.240   | 46082 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |
|  0  | 192.168.3.52    | 54216 | 192.168.3.240   | 257   | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |
|  0  | 192.168.3.240   | 53925 | 192.168.3.53    | 18192 | 6  |FP .. ..| No | 0/0 |  0  | 32 |
 1    |   0   | UNDEF |
|  0  | 192.168.3.240   | 257   | 192.168.3.52    | 54216 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |
|  0  | 192.168.3.53    | 18192 | 192.168.3.240   | 64216 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 15   |   0   | UNDEF |
|  0  | 0.0.0.0         | 8116  | 192.168.3.53    | 8116  | 17 |FP .. ..| No | 0/0 |  1  | 32 |
 1    |   0   | UNDEF |
|  0  | 0.0.0.0         | 8116  | 192.168.3.52    | 8116  | 17 |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |
|  0  | 192.168.3.240   | 64216 | 192.168.3.53    | 18192 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 15   |   0   | UNDEF |
|  0  | 192.168.3.52    | 8116  | 0.0.0.0         | 8116  | 17 |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |
|  0  | 172.20.168.16   | 63800 | 192.168.3.53    | 22    | 6  |FP .. ..| No | 0/0 |  0  | 32 |
 1    |   0   | UNDEF |
|  0  | 192.168.3.240   | 46082 | 192.168.3.52    | 18192 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |
|  0  | 192.168.3.53    | 8116  | 0.0.0.0         | 8116  | 17 |FP .. ..| No | 0/0 |  1  | 32 |
 1    |   0   | UNDEF |
|  0  | 192.168.3.53    | 22    | 172.20.168.16   | 63800 | 6  |FP .. ..| No | 0/0 |  0  | 32 |
 1    |   0   | UNDEF |
|  0  | 192.168.3.53    | 18192 | 192.168.3.240   | 53925 | 6  |FP .. ..| No | 0/0 |  0  | 32 |
 1    |   0   | UNDEF |

===============================================================================================
===========================
FP - from pool.    T - temporary connection.    PP - pending pernament.
[Expert@MyGW:0]#
```

Example 2 - Summary information only

```
[Expert@MyGW:0]# fw ctl multik gconn -s
Summary:
        Total number of global connections: 12
[Expert@MyGW:0]#
```

Example 3 - Additional information about each CoreXL Firewall instance, including the information about Firewall Priority Queues

```
[Expert@MyGW:0]# fw ctl multik gconn -p
Instance section prio info:

===============================================================================================
===============================================================================================
=======
| Segm | Src IP | S.port | Dst IP | D.port | Proto | Flags | PP |Ref Cnt(I/O)|Inst|PPAK ID|clstr
mem ID|Rec. ref|Rec. Type|Inst. Section: I/O|Inst. ID|Flags| Seq | Hold_ref |Prio:|last_enq_
jiff|queue_indx|conn_tokens

===============================================================================================
===============================================================================================
=======
|  0  | 192.168.3.52    | 18192 | 192.168.3.240   | 46082 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |Inst. Section: Out |  1  | Perm |  0    |  0  |Prio:|  0  |  -1  |  0  |
|  0  | 192.168.3.240   | 53925 | 192.168.3.53    | 18192 | 6  |FP .. ..| No | 0/0 |  0  | 32 |
 1    |   0   | UNDEF |Inst. Section: In  |  0  | Perm |  0    |  0  |Prio:|  0  |  -1  |  0  |
|  0  | 192.168.3.240   | 257   | 192.168.3.52    | 35883 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |Inst. Section: In  |  1  | Perm |  0    |  0  |Prio:|  0  |  -1  |  0  |
|  0  | 192.168.3.53    | 18192 | 192.168.3.240   | 64216 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 15   |   0   | UNDEF |Inst. Section: Out |  1  | Perm |  0    |  0  |Prio:|  0  |  -1  |  0  |
|  0  | 0.0.0.0         | 8116  | 192.168.3.53    | 8116  | 17 |FP .. ..| No | 0/0 |  1  | 32 |
 1    |   0   | UNDEF |Inst. Section: In  |  1  | Perm |  0    |  0  |Prio:|  0  |  -1  |  0  |
|  0  | 0.0.0.0         | 8116  | 192.168.3.52    | 8116  | 17 |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |Inst. Section: In  |  1  | Perm |  0    |  0  |Prio:|  0  |  -1  |  0  |
|  0  | 192.168.3.240   | 64216 | 192.168.3.53    | 18192 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 15   |   0   | UNDEF |Inst. Section: In  |  1  | Perm |  0    |  0  |Prio:|  0  |  -1  |  0  |
|  0  | 192.168.3.52    | 8116  | 0.0.0.0         | 8116  | 17 |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |Inst. Section: Out |  1  | Perm |  0    |  0  |Prio:|  0  |  -1  |  0  |
|  0  | 172.20.168.16   | 63800 | 192.168.3.53    | 22    | 6  |FP .. ..| No | 0/0 |  0  | 32 |
 1    |   0   | UNDEF |Inst. Section: In  |  0  | Perm |  494  |  0  |Prio:|  0  |  -1  |  0  |
|  0  | 192.168.3.240   | 46082 | 192.168.3.52    | 18192 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |Inst. Section: In  |  1  | Perm |  0    |  0  |Prio:|  0  |  -1  |  0  |
|  0  | 192.168.3.52    | 35883 | 192.168.3.240   | 257   | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |Inst. Section: Out |  1  | Perm |  0    |  0  |Prio:|  0  |  -1  |  0  |
|  0  | 192.168.3.53    | 8116  | 0.0.0.0         | 8116  | 17 |FP .. ..| No | 0/0 |  1  | 32 |
 1    |   0   | UNDEF |Inst. Section: Out |  1  | Perm |  0    |  0  |Prio:|  0  |  -1  |  0  |
|  0  | 192.168.3.53    | 22    | 172.20.168.16   | 63800 | 6  |FP .. ..| No | 0/0 |  0  | 32 |
 1    |   0   | UNDEF |Inst. Section: Out |  0  | Perm |  280  |  0  |Prio:|  0  |  -1  |  0  |
|  0  | 192.168.3.53    | 18192 | 192.168.3.240   | 53925 | 6  |FP .. ..| No | 0/0 |  0  | 32 |
 1    |   0   | UNDEF |Inst. Section: Out |  0  | Perm |  219  |  0  |Prio:|  0  |  -1  |  0  |

===============================================================================================
===============================================================================================
=======
FP - from pool.    T - temporary connection.    PP - pending pernament.    In - inbound.    Out
- outbound.
[Expert@MyGW:0]#
```

Example 4 - Additional information about each CoreXL Firewall instance

```
[Expert@MyGW:0]# fw ctl multik gconn -sec
Instance section:

===============================================================================================
=======================================================================
| Segm | Src IP | S.port | Dst IP | D.port | Proto | Flags | PP |Ref Cnt(I/O)|Inst|PPAK ID|clstr
mem ID|Rec. ref|Rec. Type|Inst. Section: I/O|Inst. ID|Flags| Seq | Hold_ref |

===============================================================================================
=======================================================================
|  0  | 192.168.3.52    | 18192 | 192.168.3.240   | 46082 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |Inst. Section: Out |  1  | Perm |  0    |  0  |
|  0  | 192.168.3.52    | 52864 | 192.168.3.240   | 257   | 6  |FP .. ..| No | 0/0 |  2  | 32 |
 0    |   0   | UNDEF |Inst. Section: Out |  2  | Perm |  0    |  0  |
|  0  | 192.168.3.240   | 53925 | 192.168.3.53    | 18192 | 6  |FP .. ..| No | 0/0 |  0  | 32 |
 1    |   0   | UNDEF |Inst. Section: In  |  0  | Perm |  0    |  0  |
|  0  | 192.168.3.53    | 18192 | 192.168.3.240   | 64216 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 15   |   0   | UNDEF |Inst. Section: Out |  1  | Perm |  0    |  0  |
|  0  | 192.168.3.53    | 60186 | 192.168.3.240   | 257   | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 1    |   0   | UNDEF |Inst. Section: Out |  1  | Perm |  76   |  0  |
|  0  | 0.0.0.0         | 8116  | 192.168.3.53    | 8116  | 17 |FP .. ..| No | 0/0 |  1  | 32 |
 1    |   0   | UNDEF |Inst. Section: In  |  1  | Perm |  0    |  0  |
|  0  | 0.0.0.0         | 8116  | 192.168.3.52    | 8116  | 17 |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |Inst. Section: In  |  1  | Perm |  0    |  0  |
|  0  | 192.168.3.240   | 64216 | 192.168.3.53    | 18192 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 15   |   0   | UNDEF |Inst. Section: In  |  1  | Perm |  0    |  0  |
|  0  | 192.168.3.52    | 8116  | 0.0.0.0         | 8116  | 17 |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |Inst. Section: Out |  1  | Perm |  0    |  0  |
|  0  | 172.20.168.16   | 63800 | 192.168.3.53    | 22    | 6  |FP .. ..| No | 0/0 |  0  | 32 |
 1    |   0   | UNDEF |Inst. Section: In  |  0  | Perm |  479  |  0  |
|  0  | 192.168.3.240   | 46082 | 192.168.3.52    | 18192 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 0    |   0   | UNDEF |Inst. Section: In  |  1  | Perm |  0    |  0  |
|  0  | 192.168.3.53    | 8116  | 0.0.0.0         | 8116  | 17 |FP .. ..| No | 0/0 |  1  | 32 |
 1    |   0   | UNDEF |Inst. Section: Out |  1  | Perm |  0    |  0  |
|  0  | 192.168.3.240   | 257   | 192.168.3.52    | 52864 | 6  |FP .. ..| No | 0/0 |  2  | 32 |
 0    |   0   | UNDEF |Inst. Section: In  |  2  | Perm |  0    |  0  |
|  0  | 192.168.3.53    | 22    | 172.20.168.16   | 63800 | 6  |FP .. ..| No | 0/0 |  0  | 32 |
 1    |   0   | UNDEF |Inst. Section: Out |  0  | Perm |  257  |  0  |
|  0  | 192.168.3.53    | 18192 | 192.168.3.240   | 53925 | 6  |FP .. ..| No | 0/0 |  0  | 32 |
 1    |   0   | UNDEF |Inst. Section: Out |  0  | Perm |  219  |  0  |
|  0  | 192.168.3.240   | 257   | 192.168.3.53    | 60186 | 6  |FP .. ..| No | 0/0 |  1  | 32 |
 1    |   0   | UNDEF |Inst. Section: In  |  1  | Perm |  0    |  0  |

===============================================================================================
=======================================================================
FP - from pool.    T - temporary connection.    PP - pending pernament.    In - inbound.    Out
- outbound.
[Expert@MyGW:0]#
```

fw ctl multik get_instance

Description

Shows CoreXL Firewall instance that processes the specified IPv4 connection.

ℹ **Important** - This command works only if the CoreXL Dynamic Dispatcher is disabled (see sk105261).

Syntax

- **To show the CoreXL Firewall instance that processes the specified IPv4 connection:**

```
fw ctl multik get_instance sip=<Source IPv4 Address>
dip=<Destination IPv4 Address> proto=<Protocol Number>
```

- **To show the CoreXL Firewall instance that processes the specified range of IPv4 connections:**

```
fw ctl multik get_instance sip=<Source IPv4 Address Start> -
<Source IPv4 Address End> dip=<Destination IPv4 Address Start>
- <Destination IPv4 Address End> proto=<Protocol Number>
```

Parameters

Parameter	Description
<Source IPv4 Address>	Source IPv4 address of the specified connection
<Source IPv4 Address Start>	First source IPv4 address of the specified range of IPv4 addresses
<Source IPv4 Address End>	Last source IPv4 address of the specified range of IPv4 addresses
<Destination IPv4 Address>	Destination IPv4 address of the specified connection
<Destination IPv4 Address Start>	First destination IPv4 address of the specified range of IPv4 addresses
<Destination IPv4 Address End>	Last destination IPv4 address of the specified range of IPv4 addresses
<Protocol Number>	See *IANA Protocol Numbers*. For example: ■ 1 = ICMP ■ 6 = TCP ■ 17 = UDP

Example for a specified IPv4 connection

```
[Expert@MyGW:0]# fw ctl multik get_instance sip=192.168.2.3 dip=172.30.241.66 proto=6
protocol: 6
192.168.2.3 -> 172.30.241.66 => 3
[Expert@MyGW:0]#
```

Example for a specified range of IPv4 connections

```
[Expert@MyGW:0]# fw ctl multik get_instance sip=192.168.2.3-192.168.2.8 dip=172.30.241.66
proto=6
protocol: 6
192.168.2.3 -> 172.30.241.66 => 3
192.168.2.4 -> 172.30.241.66 => 0
192.168.2.5 -> 172.30.241.66 => 3
192.168.2.6 -> 172.30.241.66 => 5
192.168.2.7 -> 172.30.241.66 => 4
192.168.2.8 -> 172.30.241.66 => 5
[Expert@MyGW:0]#
```

fw ctl multik print_heavy_conn

Description

Shows the table with Heavy Connections (that consume the most CPU resources) in the CoreXL Dynamic Dispatcher.

For more information about the CoreXL Dynamic Dispatcher, see sk105261.

CoreXL suspects that a connection is "heavy" if it meets these conditions:

- Security Gateway detected the suspected connection during the last 24 hours
- The suspected connection lasts more than 10 seconds
- CoreXL Firewall instance that processes this connection causes a CPU load of over 60%
- The suspected connection utilizes more than 50% of the total work the applicable CoreXL Firewall instance does

The output table shows this information about the Heavy Connections:

- Source IP address
- Source Port
- Destination IP address
- Destination Port
- Protocol Number
- CoreXL Firewall instance ID that processes this connection
- CoreXL Firewall instance load on the CPU
- Connection's relative load on the CoreXL Firewall instance

Notes:

- This command shows the suspected heavy connections even if they are already closed.
- In the *"CPView" on page 370* utility, go to **CPU > Top-Connections > InstancesX-Y > InstanceZ**. Refer to the **Top Connections** section.

Syntax

```
fw [-d] ctl multik print_heavy_conn
```

Parameters

Parameter	Description
`-d`	Runs the command in debug mode. Use only if you troubleshoot the command itself. **Best Practice** - If you use this parameter, then redirect the output to a file, or use the `script` command to save the entire CLI session.

Example

```
[Expert@MyGW:0]# fw ctl multik print_heavy_conn
Source: 192.168.20.31; SPort: 51006; Dest: 172.30.40.55; DPort: 80; IPP: 6; Instance 1; Instance
Load 61%; Connection instance load 100%
Source: 192.168.20.31; SPort: 50994; Dest: 172.30.40.55; DPort: 80; IPP: 6; Instance 1; Instance
Load 61%; Connection instance load 100%
Source: 192.168.20.31; SPort: 50992; Dest: 172.30.40.55; DPort: 80; IPP: 6; Instance 1; Instance
Load 61%; Connection instance load 100%
[Expert@MyGW:0]#
```

fw ctl multik prioq

Description

Configures the CoreXL Firewall Priority Queues. For more information, see sk105762.

Important - This command saves the configuration in the `$FWDIR/conf/prioq_mode.conf` file. You must **not** edit this file manually.

Syntax for IPv4

```
fw ctl multik prioq [{0 | 1 | 2}]
```

Syntax for IPv6

```
fw6 ctl multik prioq [{0 | 1 | 2}]
```

Parameters

Parameter	Description
No Parameters	Shows the interactive menu for configuration of the CoreXL Firewall Priority Queues.
`0`	Disables the CoreXL Firewall Priority Queues.
`1`	Enables the CoreXL Firewall Priority Queues.
`2`	Enables the CoreXL Firewall Priority Queues in the Evaluator-only mode.

Example

```
[Expert@MyGW:0]# fw ctl multik prioq
Current mode is Off

Available modes:
0.      Off
1.      Evaluator-only
2.      On

Choose the desired mode number: (or 3 to Quit)
[Expert@MyGW:0]#
```

fw ctl multik show_bypass_ports

Description

Shows the TCP and UDP ports configured in the bypass port list of the CoreXL Dynamic Dispatcher with the *"fw ctl multik add_bypass_port" on page 296* command.

For more information about the CoreXL Dynamic Dispatcher, see sk105261.

Important - This command reads the configuration from the `$FWDIR/conf/dispatcher_bypass.conf` file. You must **not** edit this file manually.

Syntax

```
fw ctl multik show_bypass_ports
```

Example

```
[Expert@MyGW:0]# fw ctl multik show_bypass_ports
dynamic dispatcher bypass port list:
(9999,8888)
[Expert@MyGW:0]#
```

fw ctl multik stat

Description

Shows information for each CoreXL Firewall instance.

Syntax for IPv4

```
fw [-d] ctl multik stat
```

Syntax for IPv6

```
fw6 [-d] ctl multik stat
```

Information in the output

- The ID number of each CoreXL Firewall instance (numbers starts from zero).
- The state of each CoreXL Firewall instance.
- The ID number of CPU core, on which the CoreXL Firewall instance runs (numbers starts from the highest available CPU ID).
- The number of concurrent connections the CoreXL Firewall instance currently handles.
- The peak number of concurrent connections the CoreXL Firewall instance handled from the time it started.

Parameters

Parameter	Description
`-d`	Runs the command in debug mode. Use only if you troubleshoot the command itself. **Best Practice** - If you use this parameter, then redirect the output to a file, or use the `script` command to save the entire CLI session.

Example

```
[Expert@MyGW:0]# fw ctl multik stat
ID | Active  | CPU    | Connections | Peak
----------------------------------------------
 0 | Yes     | 7      |           5 |        21
 1 | Yes     | 6      |           3 |        23
 2 | Yes     | 5      |           5 |        25
 3 | Yes     | 4      |           4 |        21
 4 | Yes     | 3      |           5 |        21
 5 | Yes     | 2      |           5 |        20
[Expert@MyGW:0]#

[Expert@MyGW:0]# fw6 ctl multik stat
ID | Active  | CPU    | Connections | Peak
----------------------------------------------
 0 | Yes     | 7      |           0 |         4
 1 | Yes     | 6      |           0 |         4
[Expert@MyGW:0]#
```

fw ctl multik start

Description

Starts all CoreXL Firewall instances on-the-fly, if they were stopped with the *"fw ctl multik stop" on page 315* command.

Syntax for IPv4

```
fw ctl multik start
```

Syntax for IPv6

```
fw6 ctl multik start
```

Example

```
[Expert@MyGW:0]# fw ctl multik stat
ID | Active  | CPU    | Connections | Peak
----------------------------------------------
 0 | No      | -      |           6 |       13
 1 | No      | -      |           3 |       11
 2 | No      | -      |           4 |       13
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik start
Instance 1 started (2 of 3 are active)
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik start
Instance 2 started (3 of 3 are active)
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik stat
ID | Active  | CPU    | Connections | Peak
----------------------------------------------
 0 | Yes     | 3      |           5 |       13
 1 | Yes     | 2      |           4 |       11
 2 | Yes     | 1      |           4 |       13
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik start
All instances are already active
[Expert@MyGW:0]#
```

fw ctl multik stop

Description

Stops all CoreXL Firewall instances on-the-fly.

Important - To start all CoreXL Firewall instances on-the-fly, run the *"fw ctl multik start" on page 314* command.

Syntax for IPv4

```
fw ctl multik stop
```

Syntax for IPv6

```
fw6 ctl multik stop
```

Example

```
[Expert@MyGW:0]# fw ctl multik stat
ID | Active   | CPU    | Connections | Peak
----------------------------------------------
 0 | Yes     | 3      |           5 |       13
 1 | Yes     | 2      |           4 |       11
 2 | Yes     | 1      |           4 |       13
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik stop
Instance 2 stopped (2 of 3 are active)
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik stop
Instance 1 stopped (1 of 3 are active)
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik stat
ID | Active   | CPU    | Connections | Peak
----------------------------------------------
 0 | Yes     | 3      |           4 |       13
 1 | No      | -      |           3 |       11
 2 | No      | -      |           7 |       13
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik stop
All instances are already inactive
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik stat
ID | Active   | CPU    | Connections | Peak
----------------------------------------------
 0 | No      | -      |           6 |       13
 1 | No      | -      |           3 |       11
 2 | No      | -      |           4 |       13
[Expert@MyGW:0]#
```

fw ctl multik utilize

Description

Shows the CoreXL queue utilization for each CoreXL Firewall instance.

Note - This command does not support VSX.

Syntax for IPv4

```
fw ctl multik utilize
```

Syntax for IPv6

```
fw6 ctl multik utilize
```

Example

```
[Expert@MyGW:0]# fw ctl multik utilize
ID | Utilize(%)   | Queue Elements
----------------------------------
 0 |            1 |             30
 1 |            0 |             10
 2 |            0 |             17
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw6 ctl multik utilize
ID | Utilize(%)   | Queue Elements
----------------------------------
 0 |            0 |              0
 1 |            0 |              0
[Expert@MyGW:0]#
```

fw ctl affinity

The `fw ctl affinity` command shows and configures the CoreXL affinity settings for:

- Interfaces
- User-space processes
- CoreXL Firewall instances

Running the 'fw ctl affinity -l' command in Gateway Mode

Description

The `fw ctl affinity -l` command shows the current CoreXL affinity settings on a Security Gateway for:

- Interfaces
- User-space processes
- CoreXL Firewall instances

Syntax

- **To see the built-in help:**

```
fw ctl affinity
```

- **To show all the existing affinities:**

```
fw ctl affinity -l [-a] [-v] [-r] [-q]
```

- **To show the affinity for a specified interface:**

```
fw ctl affinity -l -i <Interface Name>
```

- **To show the affinity for a specified CoreXL Firewall instance:**

```
fw ctl affinity -l -k <CoreXL Firewall instance ID>
```

- **To show the affinity for a specified user-space process by its PID:**

```
fw ctl affinity -l -p <Process ID>
```

- **To show the affinity for a specified user-space process by its name:**

```
fw ctl affinity -l -n <Process Name>
```

- **To show the number of system CPU cores allowed by the installed CoreXL license:**

```
fw -d ctl affinity -corelicnum
```

Parameters

Parameter	Description
`-i <Interface Name>`	Shows the affinity for the specified interface.
`-k <CoreXL Firewall instance ID>`	Shows the affinity for the specified CoreXL Firewall instance.
`-p <Process ID>`	Shows the affinity for the Check Point user-space process (for example: *fwd*, *vpnd)* specified by its PID.
`-n <Process Name>`	Shows the affinity for the Check Point user-space process (for example: *fwd*, *vpnd)* specified by its name.
`all`	Shows the affinity for all CPU cores (numbers start from zero).
`<CPU ID0> ... <CPU IDn>`	Shows the affinity for the specified CPU cores (numbers start from zero).
`-a`	Shows all current CoreXL affinities.
`-v`	Shows verbose output with IRQ numbers of interfaces.
`-r`	Shows the CoreXL affinities in reverse order.
`-q`	Suppresses the errors in the output.

Example 1

```
[Expert@MyGW:0]# fw ctl affinity -l
eth0: CPU 0
eth1: CPU 0
eth2: CPU 0
eth3: CPU 0
fw_0: CPU 7
fw_1: CPU 6
fw_2: CPU 5
fw_3: CPU 4
fw_4: CPU 3
fw_5: CPU 2
fwd: CPU 2 3 4 5 6 7
fgd50: CPU 2 3 4 5 6 7
status_proxy: CPU 2 3 4 5 6 7
rad: CPU 2 3 4 5 6 7
cpstat_monitor: CPU 2 3 4 5 6 7
mpdaemon: CPU 2 3 4 5 6 7
cpsead: CPU 2 3 4 5 6 7
cserver: CPU 2 3 4 5 6 7
rtmd: CPU 2 3 4 5 6 7
fwm: CPU 2 3 4 5 6 7
cpsemd: CPU 2 3 4 5 6 7
cpca: CPU 2 3 4 5 6 7
cprid: CPU 2 3 4 5 6 7
cpd: CPU 2 3 4 5 6 7
[Expert@MyGW:0]#
```

Example 2

```
[Expert@MyGW:0]# fw ctl affinity -l -a -v
Interface eth0 (irq 67): CPU 0
Interface eth1 (irq 75): CPU 0
Interface eth2 (irq 83): CPU 0
Interface eth3 (irq 59): CPU 0
fw_0: CPU 7
fw_1: CPU 6
fw_2: CPU 5
fw_3: CPU 4
fw_4: CPU 3
fw_5: CPU 2
fwd: CPU 2 3 4 5 6 7
fgd50: CPU 2 3 4 5 6 7
status_proxy: CPU 2 3 4 5 6 7
rad: CPU 2 3 4 5 6 7
cpstat_monitor: CPU 2 3 4 5 6 7
mpdaemon: CPU 2 3 4 5 6 7
cpsead: CPU 2 3 4 5 6 7
cserver: CPU 2 3 4 5 6 7
rtmd: CPU 2 3 4 5 6 7
fwm: CPU 2 3 4 5 6 7
cpsemd: CPU 2 3 4 5 6 7
cpca: CPU 2 3 4 5 6 7
cprid: CPU 2 3 4 5 6 7
cpd: CPU 2 3 4 5 6 7
[Expert@MyGW:0]#
```

Example 3

```
[Expert@MyGW:0]# fw ctl affinity -l -a -v -r
CPU 0:  eth0 (irq 67) eth1 (irq 75) eth2 (irq 83) eth3 (irq 59)
CPU 1:
CPU 2:  fw_5
        fwd fgd50 status_proxy rad cpstat_monitor mpdaemon cpsead cserver rtmd fwm cpsemd cpca
cprid cpd
CPU 3:  fw_4
        fwd fgd50 status_proxy rad cpstat_monitor mpdaemon cpsead cserver rtmd fwm cpsemd cpca
cprid cpd
CPU 4:  fw_3
        fwd fgd50 status_proxy rad cpstat_monitor mpdaemon cpsead cserver rtmd fwm cpsemd cpca
cprid cpd
CPU 5:  fw_2
        fwd fgd50 status_proxy rad cpstat_monitor mpdaemon cpsead cserver rtmd fwm cpsemd cpca
cprid cpd
CPU 6:  fw_1
        fwd fgd50 status_proxy rad cpstat_monitor mpdaemon cpsead cserver rtmd fwm cpsemd cpca
cprid cpd
CPU 7:  fw_0
        fwd fgd50 status_proxy rad cpstat_monitor mpdaemon cpsead cserver rtmd fwm cpsemd cpca
cprid cpd
All:
[Expert@MyGW:0]#
```

Example 4

```
[Expert@MyGW:0]# fw ctl affinity -l -i eth0
eth0: CPU 0
[Expert@MyGW:0]#
```

Example 5

```
[Expert@MyGW:0]# ps -ef | grep -v grep | egrep "PID|fwd"
UID        PID  PPID  C STIME TTY          TIME CMD
admin    26641 26452  0 Mar27 ?        00:06:56 fwd
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl affinity -l -p 26641
Process 26641: CPU 2 3 4 5 6 7
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl affinity -l -n fwd
fwd: CPU 2 3 4 5 6 7
[Expert@MyGW:0]#
```

Example 6

```
[Expert@MyGW:0]# fw ctl affinity -l -k 1
fw_1: CPU 6
[Expert@MyGW:0]#
```

Example 7

```
[Expert@MyGW:0]# fw -d ctl affinity -corelicnum
[5363 4134733584]@MyGW[4 Apr 18:11:03] Number of system CPUs 8
[5363 4134733584]@MyGW[4 Apr 18:11:03] cplic_get_navailable_cpus: fw_get_allowed_cpus_num
returned invalid value (100000) - all cpus considered as allowed!!!
4
[5363 4134733584]@MyGW[4 Apr 18:11:03] cpKeyTaskManager::~cpKeyTaskManager: called.
[Expert@MyGW:0]#
```

Running the 'fw ctl affinity -l' command in VSX Mode

Description

The `fw ctl affinity -l` command shows the CoreXL affinity settings on a VSX Gateway for:

- Interfaces
- User-space processes
- CoreXL Firewall instances

Note - Before running the `fw ctl affinity -l -x` commands, you must go to the context of the applicable Virtual System or Virtual Router with the Gaia Clish command `set virtual-system <VSID>`.

Syntax

- **To show the affinities in VSX mode (you can combine the optional parameters):**

```
fw ctl affinity -l -x
      [-vsid <VSID ranges>]
      [-cpu <CPU ID ranges>]
      [-flags {e | k | t | n | h | o}]
```

- **To show the number of system CPU cores allowed by the installed CoreXL license:**

```
fw -d ctl affinity -corelicnum
```

Parameters

Parameter	Description
`-vsid <VSID ranges>`	Shows the affinity for: - The specified single Virtual System (for example, `-vsid 7`) - The specified several Virtual Systems (for example, `-vsid 0-2 4`) **Important** - If you omit the `-vsid` parameter, the command runs in the current virtual context.
`<CPU ID ranges>`	Shows the affinity for: - The specified single CPU (for example, `-cpu 7`) - The specified several CPU cores (for example, `-cpu 0-2 4`)
`-flags {e | k | t | n | h | o}`	The `-flags` parameter requires at least one of these arguments: - `e` - Do not print the exception processes - `k` - Do not print the kernel threads - `t` - Print all process threads - `n` - Print the process name instead of the `/proc/`*<PID>* `/cmdline` - `h` - Print the CPU mask in Hex format - `o` - Print the output into the file called `/tmp/affinity_list_output` **Important** - You must specify multiple arguments together. For example: `-flags tn`

Example 1

```
[Expert@VSX_GW:0]# fw ctl affinity -l -x -cpu 0
---------------------------------------------------------------------
|PID     |VSID |               CPU                    |SRC|V|KT |EXC| NAME
---------------------------------------------------------------------
|      2 |   0 |                                    0 |   | | K |   |
|      3 |   0 |                                    0 |   | | K |   |
|      4 |   0 |                                    0 |   | | K |   |
|     14 |   0 |                                    0 |   | | K |   |
|     99 |   0 |                                    0 |   | | K |   |
|    278 |   0 |                                    0 |   | | K |   |
|    382 |   0 |                                    0 |   | | K |   |
|    674 |   0 |                                    0 |   | | K |   |
|   2195 |   0 |                                    0 |   | | K |   |
|   6348 |   0 |                                    0 |   | | K |   |
|   6378 |   0 |                                    0 |   | | K |   |
---------------------------------------------------------------------
PID   - represents the pid of the process
VSID  - represents the virtual device id
CPU   - represents the CPUs assigned to the specific process
SRC   - represents the source configuration file of the process - (V)SID / (I)nstance /
(P)rocess
V     - represents validity,star means that the actual affinity is different than the configured
affinity
KT    - represents whether the process is a kernel thread
EXC   - represents whether the process belongs to the process exception list (vsaffinity_
exception.conf)
[Expert@VSX_GW:0]#
```

Example 2

```
[Expert@VSX_GW:0]# fw ctl affinity -l -x -vsid 1
---------------------------------------------------------------------
|PID     |VSID |               CPU                    |SRC|V|KT |EXC| NAME
---------------------------------------------------------------------
|   3593 |   1 |                                1 2 3 |   | |   |   | httpd
|  10997 |   1 |                                1 2 3 |   | |   |   | cvpn_rotatelogs
|  11005 |   1 |                                1 2 3 |   | |   |   | httpd
|  22294 |   1 |                                1 2 3 |   | |   |   | routed
|  22328 |   1 |                                1 2 3 |   | |   |   | fwk_wd
|  22333 |   1 |                                1 2 3 | P | |   |   | fwk
|  22488 |   1 |                                1 2 3 |   | |   |   | cpd
|  22492 |   1 |                                1 2 3 |   | |   |   | fwd
|  22504 |   1 |                                1 2 3 |   | |   |   | cpviewd
|  22525 |   1 |                                1 2 3 |   | |   |   | mpdaemon
|  22527 |   1 |                                1 2 3 |   | |   |   | ci_http_server
|  30629 |   1 |                                1 2 3 |   | |   |   | vpnd
|  30631 |   1 |                                1 2 3 |   | |   |   | pdpd
|  30632 |   1 |                                1 2 3 |   | |   |   | pepd
|  30635 |   1 |                                1 2 3 |   | |   |   | fwpushd
|  30743 |   1 |                                1 2 3 |   | |   |   | dbwriter
|  30748 |   1 |                                1 2 3 |   | |   |   | cvpnproc
|  30752 |   1 |                                1 2 3 |   | |   |   | MoveFileServer
|  30756 |   1 |                                1 2 3 |   | |   |   | CvpnUMD
|  30760 |   1 |                                1 2 3 |   | |   |   | Pinger
|  30764 |   1 |                                1 2 3 |   | |   |   | IdlePinger
|  30770 |   1 |                                1 2 3 |   | |   |   | cvpnd
---------------------------------------------------------------------
[Expert@VSX_GW:0]#
```

Running the 'fw ctl affinity -s' command in Gateway Mode

Description

The `fw ctl affinity -s` command configures the CoreXL affinity settings on a Security Gateway for:

- Interfaces
- User-space processes
- CoreXL Firewall instances

Note - The Security Gateway saves these changes in the *$FWDIR/conf/fwaffinity.conf* configuration file.

Syntax

- **To see the built-in help:**

```
fw ctl affinity
```

- **To configure the affinity for a specified interface by its name:**

```
fw ctl affinity -s -i <Interface Name>
      all
      <CPU ID0> [ <CPU ID1> ... <CPU IDn> ]
```

- **To configure the affinity for a specified CoreXL Firewall instance:**

```
fw ctl affinity -s -k <CoreXL Firewall instance ID>
      all
      <CPU ID0> [ <CPU ID1> ... <CPU IDn> ]
```

- **To configure the affinity for a specified user-space process by its PID:**

```
fw ctl affinity -s -p <Process ID>
      all
      <CPU ID0> [ <CPU ID1> ... <CPU IDn> ]
```

- **To configure the affinity for a specified user-space process by its name:**

```
fw ctl affinity -s -n <Process Name>
      all
      <CPU ID0> [ <CPU ID1> ... <CPU IDn> ]
```

Parameters

Parameter	Description
-i *<Interface Name>*	Configures the affinity for the specified interface.
-k *<CoreXL Firewall instance ID>*	Configures the affinity for the specified CoreXL Firewall instance.
-p *<Process ID>*	Configures the affinity for the Check Point user-space process (for example: *fwd*, *vpnd)* specified by its PID.
-n *<Process Name>*	Configures the affinity for the Check Point user-space process (for example: *fwd*, *vpnd)* specified by its name. **Important** - The process name is case-sensitive.
all	Configures the affinity for all CPU cores (numbers start from zero).
<CPU ID0> ... *<CPU IDn>*	Configures the affinity for the specified CPU cores (numbers start from zero).

Example 1 - Affine the interface eth1 to the CPU core #1

```
[Expert@MyGW:0]# fw ctl affinity -s -i eth1 1
eth1: CPU 1 - set successfully
Multi-queue affinity was not changed.  For More info, see sk113834.
[Expert@MyGW:0]#
```

Example 2 - Affine the CoreXL Firewall instance #1 to the CPU core #2

```
[Expert@MyGW:0]# fw ctl affinity -s -k 1 2
fw_1: CPU 2 - set successfully
Multi-queue affinity was not changed.  For More info, see sk113834.
[Expert@MyGW:0]#
```

Example 3 - Affine the process CPD by its PID to the CPU core #2

```
[Expert@MyGW:0]# cpwd_admin list | egrep "PID|cpd"
APP         PID    STAT  #START  START_TIME             MON  COMMAND
CPD         6080   E     1       [13:46:27] 17/9/2018   Y    cpd
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl affinity -s -p 6080 2
Process 6080: CPU 2 - set successfully
Multi-queue affinity was not changed.  For More info, see sk113834.
[Expert@MyGW:0]#
```

Example 4 - Affine the process CPD by its name to the CPU core #2

```
[Expert@MyGW:0]# fw ctl affinity -s -n cpd 2
cpd: CPU 2 - set successfully
Multi-queue affinity was not changed.  For More info, see sk113834.
[Expert@MyGW:0]#
```

Running the 'fw ctl affinity -s' command in VSX Mode

Description

The `fw ctl affinity -s` command configures the CoreXL affinity settings on a VSX Gateway for:

- Interfaces
- User-space processes
- CoreXL Firewall instances

Syntax

- **To see the built-in help:**

```
fw ctl affinity
```

- **To configure the affinities of Virtual Systems:**

```
fw ctl affinity -s -d [-vsid <VSID ranges> ] -cpu <CPU ID
ranges>
```

- **To configure the affinities of a specified user-space process:**

```
fw ctl affinity -s -d -pname <Process Name> [-vsid <VSID
ranges>]
      -cpu all
      -cpu <CPU ID ranges>
```

- **To configure the affinities of specified FWK daemon instances (user-space Firewall):**

```
fw ctl affinity -s -d -inst <Instances Ranges> -cpu <CPU ID
ranges>
```

- **To configure the affinities of all FWK instances (user-space Firewalls):**

```
fw ctl affinity -s -d -fwkall <Number of CPUs>
```

- **To reset the affinities to defaults:**

```
fw ctl affinity
      -vsx_factory_defaults
      -vsx_factory_defaults_no_prompt
```

Important

- The VSX Gateway saves these changes in the *$FWDIR/conf/fwaffinity.conf* configuration file.
- When you configure affinity of an interface, it automatically configures the affinities of all other interfaces that share the same IRQ to the same CPU core.

Parameters

Parameter	Description
`-vsid <VSID ranges>`	Configures the affinity for: - One specified Virtual System. For example: `-vsid 7` - Several specified Virtual Systems. For example: `-vsid 0-2 4` **Note** - If you omit the `-vsid` parameter, the command uses the current virtual context.
`<CPU ID ranges>`	Configures the affinity to: - One specified CPU core. For example: `-cpu 7` - Several specified CPU cores. For example: `-cpu 0-2 4` **Important** - Numbers of CPU cores start from zero.
`-pname <Process Name>`	Configures the affinity for the Check Point daemon specified by its name (for example: *fwd*, *vpnd)*. **Important** - The process name is case-sensitive.
`-inst <Instances Ranges>`	Configures the affinity for: - One specified FWK daemon instance. For example: `-inst 7` - Several specified FWK daemon instances. For example: `-inst 0 2 4`

Parameter	Description
`-fwkall <Number of CPUs>`	Configures the affinity for all running FWK daemon instances to the specified number of CPU cores. If it is necessary to affine all running FWK daemon instances to all CPU cores, enter the number of all available CPU cores.
`-vsx_factory_defaults`	Deletes all existing affinity settings and creates the default affinity settings during the next reboot. **Important** - Before this operation, the command prompts the user whether to proceed. You must reboot to complete the operation.
`-vsx_factory_defaults_no_prompt`	Deletes all current affinity settings and creates the default affinity settings during the next reboot. **Important** - Before this operation, the command does **not** prompt the user whether to proceed. You must reboot to complete the operation.

Example 1 - Affine the Virtual Devices #0,1,2,4,7,8 to the CPU cores #0,1,2,4

```
[Expert@MyGW:0]# fw ctl affinity -s -d -vsid 0-2 4 6-8 -cpu 0-2 4
VDevice 0-2 4 6-8 : CPU 0 1 2 4 - set successfully
Multi-queue affinity was not changed.  For More info, see sk113834.
[Expert@MyGW:0]#
```

Example 2 - Affine the process CPD by its name for Virtual Devices #0-12 to the CPU core #7

```
[Expert@MyGW:0]# fw ctl affinity -s -d -pname cpd -vsid 0-12 -cpu 7
VDevice 0-12 : CPU 7 - set successfully
Multi-queue affinity was not changed.  For More info, see sk113834.
Warning: some of the VSIDs did not exist
[Expert@MyGW:0]#
```

Example 3 - Affine the FWK daemon instances #0,2,4 to the CPU core #5

```
[Expert@MyGW:0]# fw ctl affinity -s -d -inst 0 2 4 -cpu 5
VDevice 0 2 4: CPU 5 - set successfully
Multi-queue affinity was not changed.  For More info, see sk113834.
[Expert@MyGW:0]#
```

Example 4 - Affine all FWK daemon instances to the last two CPU cores

```
[Expert@MyGW:0]# fw ctl affinity -s -d -fwkall 2
VDevice 0-2 : CPU 2 3 - set successfully
Multi-queue affinity was not changed.  For More info, see sk113834.
[Expert@MyGW:0]#
```

Example 5 - Affine all FWK daemon instances to all CPU cores

```
[Expert@MyGW:0]# fw ctl affinity -s -d -fwkall 4
There are configured processes/FWK instances
(y) will override all currently configured affinity and erase the configuration files
(n) will set affinity only for unconfigured processes/threads
Do you want to override existing configurations (y/n) ? y
VDevice 0-2 : CPU all - set successfully
Multi-queue affinity was not changed.  For More info, see sk113834.
[Expert@MyGW:0]#
```

fw -i

Description

By default, the "`fw`" commands apply to the entire Security Gateway.

The `fw` commands show aggregated information for all CoreXL Firewall instances.

The `fw -i` commands apply to the specified CoreXL Firewall instance.

Syntax

```
fw -i <ID of CoreXL Firewall instance> <Command>
```

Parameters

Parameter	Description
`<ID of CoreXL Firewall instance>`	Specifies the ID of the CoreXL Firewall instance. To see the available IDs, run the "`fw ctl multik stat`" *"fw ctl multik stat" on page 312* command.
`<Command>`	Only these commands support the `fw -i` syntax: ■ `fw -i <ID> conntab ...` ■ `fw -i <ID> ctl get ...` ■ `fw -i <ID> ctl leak ...` ■ `fw -i <ID> ctl pstat ...` ■ `fw -i <ID> ctl set ...` ■ `fw -i <ID> monitor ...` ■ `fw -i <ID> tab ...` For details and additional parameters for any of these commands, refer to the corresponding entry for each command.

Example 1 - Show the Connections table for CoreXL Firewall instance #1

```
fw -i 1 tab -t connections
```

Example 2 - Show various internal statistics for CoreXL Firewall instance #1

```
fw -i 1 ctl pstat
```

fwboot bootconf

Description

Configures boot security options.

Notes:

- You must run this command from the Expert mode.
- The settings are saved in the `$FWDIR/boot/boot.conf` file.

  Warning - To avoid issues, do not edit the `$FWDIR/boot/boot.conf` file manually. Edit the file only with this command.
- Refer to these related commands:
  - *"fwboot corexl" on page 338*

Syntax to show the current boot security options

```
[Expert@HostName:0]# $FWDIR/boot/fwboot bootconf
      get_corexl
      get_core_override
      get_def
      get_ipf
      get_ipv6
      get_kernnum
      get_kern6num
```

Syntax to configure the boot security options

```
[Expert@HostName:0]# $FWDIR/boot/fwboot bootconf
      set_corexl {0 | 1}
      set_core_override <number>
      set_def [</path/filename>]
      set_ipf {0 | 1}
      set_ipv6 {0 | 1}
      set_kernnum <number>
      set_kern6num <number>
```

Parameters

Parameter	Description
No Parameters	Shows the built-in help with available parameters.

Parameter	Description
`get_corexl`	Shows if the CoreXL is enabled or disabled: ■ 0 - disabled ■ 1 - enabled **Note** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `COREXL_INSTALLED`.
`get_core_override`	Shows the number of overriding CPU cores. The SMT (HyperThreading) feature (sk93000) uses this configuration to set the number of CPU cores after reboot. **Note** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `CORE_OVERRIDE`.
`get_def`	Shows the configured path and the name of the Default Filter policy file (default is `$FWDIR/boot/default.bin`). **Note** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `DEFAULT_FILTER_PATH`.
`get_ipf`	Shows if the IP Forwarding during boot is enabled or disabled: ■ 0 - disabled (Security Gateway does not forward traffic between its interfaces during boot) ■ 1 - enabled **Note** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `CTL_IPFORWARDING`.
`get_ipv6`	Shows if the IPv6 support is enabled or disabled: ■ 0 - disabled ■ 1 - enabled **Note** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `IPV6_INSTALLED`.
`get_kernnum`	Shows the configured number of IPv4 CoreXL Firewall instances. **Note** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `KERN_INSTANCE_NUM`.
`get_kern6num`	Shows the configured number of IPv6 CoreXL Firewall instances. **Note** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `KERN6_INSTANCE_NUM`.

Parameter	Description
`set_corexl {0 | 1}`	Enables or disables CoreXL: - 0 - disables - 1 - enables **Notes:** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `COREXL_INSTALLED`. - To configure CoreXL, use the *"cpconfig" on page 289* menu.
`set_core_override <number>`	Configures the number of overriding CPU cores. The SMT (HyperThreading) feature (sk93000) uses this configuration to set the number of CPU cores after reboot. **Note** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `CORE_OVERRIDE`.
`set_def [</path/filename>]`	Configures the path and the name of the Default Filter policy file (default is `$FWDIR/boot/default.bin`). **Notes:** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `DEFAULT_FILTER_PATH`. - If you do not specify the path and the name explicitly, then the value of the `DEFAULT_FILTER_PATH` is set to 0. As a result, Security Gateway does not load a Default Filter during boot. **Best Practice** - The best location for this file is the `$FWDIR/boot/` directory.
`set_ipf {0 | 1}`	Configures the IP forwarding during boot: - 0 - disables (forbids the Security Gateway to forward traffic between its interfaces during boot) - 1 - enables **Note** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `CTL_IPFORWARDING`.

Parameter	Description
`set_ipv6 {0 | 1}`	Enables or disables the IPv6 Support: - 0 - disables - 1 - enables **Notes:** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `IPV6_INSTALLED`. - Configure the IPv6 Support in Gaia Portal, or Gaia Clish. See the *R81 Gaia Administration Guide*.
`set_kernnum <number>`	Configures the number of IPv4 CoreXL Firewall instances. **Notes:** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `KERN_INSTANCE_NUM`. - To configure CoreXL, use the *"cpconfig" on page 289* menu.
`set_kern6num <number>`	Configures the number of IPv6 CoreXL Firewall instances. **Notes:** - In the `$FWDIR/boot/boot.conf` file, refer to the value of the `KERN6_INSTANCE_NUM`. - To configure CoreXL, use the *"cpconfig" on page 289* menu.

fwboot corexl

Description

Configures and monitors the CoreXL.

Note - The settings are saved in the `$FWDIR/boot/boot.conf` file.

Warning - To avoid issues, do not edit the `$FWDIR/boot/boot.conf` file manually. Edit the file only with this command.

Syntax to show CoreXL configuration

```
[Expert@HostName:0]# $FWDIR/boot/fwboot corexl
      core_count
      curr_instance4_count
      curr_instance6_count
      def_instance4_count
      def_instance6_count
      eligible
      installed
      max_instance4_count
      max_instances4_32bit
      max_instances4_64bit
      max_instance6_count
      max_instances_count
      max_instances_32bit
      max_instances_64bit
      min_instance_count
      unsupported_features
```

Syntax to configure CoreXL

Important:

- The configuration commands are for Check Point use only. To configure CoreXL, use the **Check Point CoreXL** option in the *"cpconfig" on page 289* menu.
- After all changes in CoreXL configuration on the Security Gateway, you must reboot it.
- In a Cluster, you must configure all the Cluster Members in the same way.

```
[Expert@HostName:0]# $FWDIR/boot/fwboot corexl
      def_by_allowed [n]
      default
      [-v] disable
      [-v] enable [n] [-6 k]
      vmalloc_recalculate
```

Parameters

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td>No Parameters</td><td>Shows the built-in help with available parameters.</td></tr>
<tr><td><code>core_count</code></td><td>Returns the number of CPU cores on this computer.
<b>Example</b>
<pre>
[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl core_
count
[Expert@MyGW:0]# echo $?
4
[Expert@MyGW:0]#
[Expert@MyGW:0]# cat /proc/cpuinfo | grep processor
processor : 0
processor : 1
processor : 2
processor : 3
[Expert@MyGW:0]#
</pre></td></tr>
</table>

<table>
<thead>
<tr><th>Parameter</th><th>Description</th></tr>
</thead>
<tbody>
<tr><td>

`curr_instance4_count`

</td><td>

Returns the current configured number of IPv4 CoreXL Firewall instances.

Example

```
[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl curr_
instance4_count
[Expert@MyGW:0]# echo $?
3
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl multik stat
ID | Active  | CPU    | Connections | Peak
----------------------------------------------
 0 | Yes     | 3      |          11 |       18
 1 | Yes     | 2      |          12 |       18
 2 | Yes     | 1      |          13 |       18
[Expert@MyGW:0]#
```

</td></tr>
<tr><td>

`curr_instance6_count`

</td><td>

Returns the current configured number of IPv6 CoreXL Firewall instances.

Example

```
[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl curr_
instance6_count
[Expert@MyGW:0]# echo $?
2
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw6 ctl multik stat
ID | Active  | CPU    | Connections | Peak
----------------------------------------------
 0 | Yes     | 3      |          11 |       18
 1 | Yes     | 2      |          12 |       18
[Expert@MyGW:0]#
```

</td></tr>
<tr><td>

`def_by_allowed [n]`

</td><td>

Sets the default configuration for CoreXL according to the specified allowed number of CPU cores.

</td></tr>
<tr><td>

`default`

</td><td>

Sets the default configuration for CoreXL.

</td></tr>
</tbody>
</table>

Parameter	Description

`def_instance4_count`	Returns the default number of IPv4 CoreXL Firewall instances for this Security Gateway.
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl def_instance4_count`
	`[Expert@MyGW:0]# echo $?`
	`3`
	`[Expert@MyGW:0]#`

`def_instance6_count`	Returns the default number of IPv4 CoreXL Firewall instances for this Security Gateway.
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl def_instance6_count`
	`[Expert@MyGW:0]# echo $?`
	`2`
	`[Expert@MyGW:0]#`

`[-v] disable`	Disables CoreXL. - `-v` - Leaves the high memory (`vmalloc`) unchanged. See the *"cp_conf corexl" on page 287* command.

`eligible`	Returns whether CoreXL can be enabled on this Security Gateway.
	- 0 - CoreXL cannot be enabled
	- 1 - CoreXL can be enabled
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl eligible`
	`[Expert@MyGW:0]# echo $?`
	`1`
	`[Expert@MyGW:0]#`

Parameter	Description
`[-v] enable [n] [-6 k]`	Enables CoreXL with '`n`' IPv4 Firewall instances and optionally '`k`' IPv6 Firewall instances. ■ `-v` - Leaves the high memory (`vmalloc`) unchanged. ■ `n` - Denotes the number of IPv4 CoreXL Firewall instances. ■ `k` - Denotes the number of IPv6 CoreXL Firewall instances. See the *"cp_conf corexl" on page 287* command.

`installed`	Returns whether CoreXL is installed (enabled) on this Security Gateway.
	■ 0 - CoreXL is not enabled
	■ 1 - CoreXL is enabled
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl installed`
	`[Expert@MyGW:0]# echo $?`
	`1`
	`[Expert@MyGW:0]#`

`max_instance4_count`	Returns the maximal allowed number of IPv4 CoreXL Firewall instances for this Security Gateway.
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl max_instance4_count`
	`[Expert@MyGW:0]# echo $?`
	`4`
	`[Expert@MyGW:0]#`

`max_instances4_32bit`	Returns the maximal allowed number of IPv4 CoreXL Firewall instances for a Security Gateway that runs Gaia with 32-bit kernel.
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl max_instances4_32bit`
	`[Expert@MyGW:0]# echo $?`
	`14`
	`[Expert@MyGW:0]#`

Parameter	Description

`max_instances4_64bit`	Returns the maximal allowed number of IPv4 CoreXL Firewall instances for a Security Gateway that runs Gaia with 64-bit kernel.
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl max_instances4_64bit`
	`[Expert@MyGW:0]# echo $?`
	`38`
	`[Expert@MyGW:0]#`

`max_instance6_count`	Returns the maximal allowed number of IPv6 CoreXL Firewall instances for this Security Gateway.
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl max_instance6_count`
	`[Expert@MyGW:0]# echo $?`
	`3`
	`[Expert@MyGW:0]#`

`max_instances_count`	Returns the total maximal allowed number of CoreXL Firewall instances (IPv4 and IPv6) for this Security Gateway.
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl max_instances_count`
	`[Expert@MyGW:0]# echo $?`
	`40`
	`[Expert@MyGW:0]#`

`max_instances_32bit`	Returns the total maximal allowed number of CoreXL Firewall instances for a Security Gateway that runs Gaia with 32-bit kernel.
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl max_instances_32bit`
	`[Expert@MyGW:0]# echo $?`
	`16`
	`[Expert@MyGW:0]#`

Parameter	Description

`max_instances_64bit`	Returns the total maximal allowed number of CoreXL Firewall instances for a Security Gateway that runs Gaia with 64-bit kernel.
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl max_instances_64bit`
	`[Expert@MyGW:0]# echo $?`
	`40`
	`[Expert@MyGW:0]#`

`min_instance_count`	Returns the minimal allowed number of IPv4 CoreXL Firewall instances.
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl min_instance_count`
	`[Expert@MyGW:0]# echo $?`
	`2`
	`[Expert@MyGW:0]#`

`vmalloc_recalculate`	Updates the value of the `vmalloc` parameter in the `/boot/grub/grub.conf` file.

`unsupported_features`	Returns 1 if at least one feature is configured, which CoreXL does not support.
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot corexl unsupported_features`
	`corexl unsupported feature: QoS is configured.`
	`[Expert@MyGW:0]# echo $?`
	`1`
	`[Expert@MyGW:0]#`

fwboot cpuid

Description

Shows the number of available CPUs and CPU cores on this Security Gateway.

Syntax

```
[Expert@HostName:0]# $FWDIR/boot/fwboot cpuid
      {-h | -help | --help}
      -c
      --full
      ht_aware
      -n
      --possible
```

Parameters

Parameter	Description

No Parameters	Shows the IDs of the available CPU cores on this Security Gateway.
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot cpuid`
	`3 2 1 0`
	`[Expert@MyGW:0]#`

`-c`	Counts the number of available CPU cores on this Security Gateway.
	The command stores the returned number as its exit code.
	Example
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot cpuid -c`
	`[Expert@MyGW:0]# echo $?`
	`4`
	`[Expert@MyGW:0]#`

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td><code>--full</code></td><td>

Shows a full map of the available CPUs and CPU cores on this Security Gateway.

Example

```
[Expert@MyGW:0]# $FWDIR/boot/fwboot cpuid --full
cpuid phys_id core_id thread_id
 0  0  0  0
 1  2  0  0
 2  4  0  0
 3  6  0  0
[Expert@MyGW:0]#
```

</td></tr>
<tr><td><code>ht_aware</code></td><td>

Shows the CPU cores in the order of their awareness of Hyper-Threading.

Example

```
[Expert@MyGW:0]# $FWDIR/boot/fwboot cpuid ht_aware
3  2  1  0
[Expert@MyGW:0]#
```

</td></tr>
<tr><td><code>-n</code></td><td>

Counts the number of available CPUs on this Security Gateway.
The command stores the returned number as its exit code.

Example

```
[Expert@MyGW:0]# $FWDIR/boot/fwboot cpuid -n
[Expert@MyGW:0]# echo $?
4
[Expert@MyGW:0]#
```

</td></tr>
<tr><td><code>--possible</code></td><td>

Counts the number of possible CPU cores.
The command stores the returned number as its exit code.

Example

```
[Expert@MyGW:0]# $FWDIR/boot/fwboot cpuid --possible
[Expert@MyGW:0]# echo $?
4
[Expert@MyGW:0]#
```

</td></tr>
</table>

fwboot ht

Important - This command is obsolete and is not supported. To configure SMT (HyperThreading) feature, follow sk93000.

fwboot multik_reg

Description

Shows the internal memory address of the registration function for the specified CoreXL Firewall instance.

Important - This command is for Check Point use only.

Note - You must run this command from the Expert mode.

Syntax

```
[Expert@HostName:0]# $FWDIR/boot/fwboot multik_reg <Number of
CoreXL Firewall instance> {ipv4 | ipv6} [-d]
```

Parameters

Parameter	Description
No Parameters	Shows the built-in help with available parameters.
`<Number of CoreXL Firewall instance>`	Specifies the ID number of the CoreXL Firewall instance.
`ipv4`	Specifies to work with IPv4 CoreXL Firewall instances.
`ipv6`	Specifies to work with IPv6 CoreXL Firewall instances.
`-d`	Shows the decimal 64-bit address of the hook function.

Example

```
[Expert@MyGW:0]# fw ctl multik stat
ID | Active  | CPU    | Connections | Peak
----------------------------------------------
 0 | Yes     | 3      |          11 |       18
 1 | Yes     | 2      |          12 |       18
 2 | Yes     | 1      |          13 |       18
[Expert@MyGW:0]#


[Expert@MyGW:0]# $FWDIR/boot/fwboot multik_reg 0 ipv4
0
[Expert@MyGW:0]#

[Expert@MyGW:0]# $FWDIR/boot/fwboot multik_reg 1 ipv4
0xffffffff8a2a5690
[Expert@MyGW:0]#

[Expert@MyGW:0]# $FWDIR/boot/fwboot multik_reg 2 ipv4
0xffffffff8a2a5690
[Expert@MyGW:0]#
```

fwboot post_drv

Description

Loads the Firewall driver for CoreXL during boot.

Important:

- This command is for Check Point use only.
- If you run this command, Security Gateway can block all traffic. In such case, you must connect to the Security Gateway over a console and restart Check Point services with the "cpstop" and "cpstart" commands. Alternatively, you can reboot the Security Gateway.

Note - You must run this command from the Expert mode.

Syntax

```
[Expert@HostName:0]# $FWDIR/boot/fwboot post_drv {ipv4 | ipv6}
```

Parameters

Parameter	Description
No Parameters	Shows the built-in help with available parameters.
ipv4	Loads the IPv4 Firewall driver for CoreXL.
ipv6	Loads the IPv6 Firewall driver for CoreXL.

Multi-Queue

By default, each network interface has one traffic queue handled by one CPU.

You cannot use more CPU cores for acceleration than the number of interfaces handling traffic.

Multi-Queue configures more than one traffic queue for each network interface.

For each interface, more than one CPU core is used for acceleration.

Note - Multi-Queue is applicable only if SecureXL is enabled (this is the default).

Overview:

- Multi-Queue is enabled by default on all interfaces that use the supported drivers.
- The number of traffic queues on each supported interface is determined automatically, based on:
  - The number of available CPU cores that run CoreXL SND Instances.
  - The limitations of the interfaces and its driver.
- Traffic queues are automatically affined to the CPU cores that runs CoreXL SND Instances.
- Changes in Multi-Queue configuration do **not** require a reboot.
- You configure Multi-Queue on the command line - either in Gaia Clish, or in the Expert mode.

Multi-Queue Requirements and Limitations

- Multi-Queue only supports Security Gateways with two or more CPU cores.
- Multi-Queue only supports interfaces that use these drivers:

Driver	Max Speed	Description
igb	1 Gbps	Intel® Network Adapter Driver for PCIe 1 Gigabit Ethernet Network
ixgbe	10 Gbps	Intel® Network Adapter Driver for PCIe 10 Gigabit Ethernet Network
i40e	40 Gbps	Intel® Network Adapter Driver for PCIe 40 Gigabit Ethernet Network
i40evf	40 Gbps	Intel® i40e driver for Virtual Function Network Devices
mlx5_core	40 Gbps	Mellanox® ConnectX® mlx5 core driver
ena	20 Gbps	Elastic Network Adapter in Amazon® EC2
virtio_net	10 Gbps	VirtIO paravirtualized device driver from KVM®
vmxnet3	10 Gbps	VMXNET Generation 3 driver from VMware®

- Multi-Queue does not use network interfaces that are currently in the down state.
- The number of traffic queues is limited by the number of CPU cores and the type of interface driver:

Interface Driver	Maximal Number of RX Queues
igb	2-16 (depends on the interface)
ixgbe	16
i40e	64
i40evf	4
mlx5_core	60
ena	Configured automatically

Interface Driver	Maximal Number of RX Queues
virtio_net	Configured automatically
vmxnet3	Configured automatically

- In a Cluster, you must configure all the Cluster Members in the same way.

Deciding Whether to Enable the Multi-Queue

This section helps you decide if you can benefit from the Multi-Queue.

Best Practice - We recommend that you perform the steps below *before* you
> configure the Multi-Queue.

1. **Make sure that network interfaces support the Multi-Queue**

   Only network cards that use these drivers can support the Multi-Queue.

   See *"Multi-Queue Requirements and Limitations" on page 352*.

   Important - Before you upgrade these drivers, make sure that the latest version supports the Multi-Queue.

   Notes:

   - To view, which driver an interface uses, run this command in the Expert mode:

     ```
     ethtool -i <Name of Interface>
     ```

   - When you install a new interface, you must run these two commands in the Expert mode:

     ```
     mq_mng --reconf
     reboot
     ```

2. **Make sure that SecureXL is enabled**

Step	Instructions
1	Connect to the command line on the Security Gateway.
2	Log in to the Gaia Clish, or the Expert mode.

3	Get the SecureXL state (see *"fwaccel stat" on page 92*):
	`fwaccel stat -t`

<table>
<tr><th>Step</th><th>Instructions</th></tr>
<tr><td>4</td><td>Examine the <b>Status</b> column.
<b>Example from a non-VSX Gateway</b>
<pre>
[Expert@MyGW:0]# fwaccel stat -t
+---
-----+
|Id|Name |Status |Interfaces |Features
    |
+---
-----+
|0 |SND |enabled |eth0,eth1,eth2,eth3,eth4,|
    |
| | | |eth5,eth6,eth7 |Acceleration,Cryptography
    |
+---
-----+
[Expert@MyGW:0]#
</pre></td></tr>
<tr><td>5</td><td>If the SecureXL is disabled, enable it (see <i>"fwaccel on" on page 81</i>):
<pre>
fwaccel on
</pre></td></tr>
</table>

3. **Examine the CPU roles allocation**

<table>
<tr><th>Step</th><th>Instructions</th></tr>
<tr><td>1</td><td>Connect to the command line on the Security Gateway.</td></tr>
<tr><td>2</td><td>Log in to the Gaia Clish, or the Expert mode.</td></tr>
<tr><td>3</td><td>Get the list of CPU roles (see <i>"fw ctl affinity" on page 317</i>):
<pre>
fw ctl affinity -l [-a] [-v] [-r]
</pre>
<b>Example</b>
CPU0 and CPU1 run the CoreXL SND instances:
<pre>
[Expert@GW:0]# fw ctl affinity -l
Mgmt: CPU 0
eth1-04: CPU 1
eth1-05: CPU 0
eth1-06: CPU 1
eth1-07: CPU 0
fw_0: CPU 5
fw_1: CPU 4
fw_2: CPU 3
fw_3: CPU 2
[Expert@GW:0]#
</pre></td></tr>
</table>

4. **Examine the CPU cores utilization**

<table>
<tr><th>Step</th><th>Instructions</th></tr>
<tr><td>1</td><td>Connect to the command line on the Security Gateway.</td></tr>
<tr><td>2</td><td>Log in to the Gaia Clish, or the Expert mode.</td></tr>
<tr><td>3</td><td>

Get the CPU cores utilization:

```
top
```

</td></tr>
<tr><td>4</td><td>

Press 1 to show all the CPU cores.

Example

- CPU cores that run CoreXL SND instances (CPU0 and CPU1) are approximately 30% idle.
- CPU cores that run CoreXL Firewall instances are approximately 70% idle.

```
 top - 18:02:33 up 8 days, 1:18, 1 user, load average: 1.22, 1.38, 1.48
Tasks:  137 total,  3 running,  134 sleeping,  0 stopped,  0 zombie

Cpu0 : 2.0%us, 0.0%sy, 0.0%ni, 28.7%id, 5.9%wa, 0.0%hi, 63.4%si, 0.0%st
Cpu1 : 0.0%us, 1.0%sy, 0.0%ni, 27.6%id, 0.0%wa, 0.0%hi, 71.4%si, 0.0%st
Cpu2 : 2.0%us, 2.0%sy, 0.0%ni, 66.5%id, 0.0%wa, 4.0%hi, 25.5%si, 0.0%st
Cpu3 : 1.0%us, 2.0%sy, 0.0%ni, 71.3%id, 0.0%wa, 0.0%hi, 25.7%si, 0.0%st
Cpu4 : 5.0%us, 1.0%sy, 0.0%ni, 69.0%id, 0.0%wa, 0.0%hi, 25.0%si, 0.0%st

Mem:  12224020k total, 70005820k used,   5218200k free, 273536k buffers
Swap: 14707496k total,        0k used,  14707496k free, 484340k cached

 PID USER   PR  NI  VIRT  RES  SHR  S  %CPU  %MEM   TIME+  COMMAND
3301 root   15   0     0    O    0  R    31   0.0  747:04  [fw_worker_3]
3326 root   15   0     0    O    0  R    29   0.0  593:35  [fw_worker_0]
... ... ...
```

</td></tr>
</table>

5. **Decide if you can allocate more CPU cores to run the CoreXL SND instances**

To decide if you can allocate more CPU cores to run the CoreXL SND instances

If you have more active network interfaces than the CPU cores that run CoreXL SND instances, you can allocate more CPU cores to run more CoreXL SND instances.

We recommend to configure the Multi-Queue when:

a. CoreXL SND instances cause high CPU load (idle is less than 20%).

b. CoreXL Firewall instances cause low CPU load (idle is greater than 50%).

Note - You cannot assign more CPU cores to run CoreXL SND instances if you change interface IRQ affinity.

Multi-Queue Basic Configuration

In This Section:

Multi-Queue Configuration in the Expert mode *357*

Multi-Queue Configuration in Gaia Clish *362*

You configure Multi-Queue on the command line in one of these shells:

- In the Expert mode
- In Gaia Clish

Multi-Queue Configuration in the Expert mode

Description

The `mq_mng` utility shows and configures the Multi-Queue on supported interfaces.

Syntax

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- You must run these commands in the Expert mode.
- Change in the Multi-Queue mode can cause short packet loss.

- **To see the built-in help**

```
mq_mng {-h | --help}
```

- **To show the existing Multi-Queue configuration:**

```
mq_mng {-o | --show} [{-v | -vv}] [-a]
```

- **To configure the Multi-Queue for the specified driver:**

```
mq_mng {-s | --set-mode}
      auto
      manual
            {-i | --interface} <Names of Interfaces>
            {-c | --core} <IDs of CPU Cores>
      off
            [{-i | --interface} <Names of Interfaces>]
```

- **To apply the existing Multi-Queue policy:**

```
mq_mng {-r | --reconf}
```

Parameters

Parameter	Description
`-h | --help`	Shows built-in help.
`-o | --show`	Shows the existing Multi-Queue configuration.
`-v | -vv`	Verbose output.
`-a`	Shows all interfaces in the output.

Parameter	Description

`-s | --set-mode`	Configures the Multi-Queue mode:
	■ `auto` - Automatic mode (this is the default). Multi-Queue automatically configures the affinity of all supported interfaces to CPU cores that run CoreXL SND Instances.
	■ `manual` - Manual mode. Administrator configures the affinity of interfaces to CPU cores that run CoreXL SND Instances. In this mode, you can specify interfaces, CPU cores, or both.
	■ `off` - Disables the Multi-Queue on all or specified supported interfaces.
	Important - Change in the Multi-Queue mode can cause short packet loss.
	Notes:
	■ To specify interfaces:
	• Use this syntax:
	`{-i | --interface} <Names of Interfaces>`
	• If you do not specify interfaces, then the configuration applies to all supported interfaces.
	• To specify a specific interface, enter its name (for example: `-i eth2`).
	• To specify several interfaces, enter their names separates with spaces (for example: `-i eth2 eth4`).
	■ To specify CPU cores:
	• Use this syntax:
	`{-c | --core} <IDs of CPU Cores that run CoreXL SND Instances>`
	• To specify a specific CPU core, enter its ID number (for example: `-c 1`).
	• To specify several nonconsecutive CPU cores, enter their ID numbers separated with spaces (for example: `-c 1 3`) or commas (for example: `-c 1,3`).
	• To specify several consecutive CPU cores, enter their first and last ID numbers separated with a hyphen (for example: `-c 3-6`).
	■ To see the current CoreXL affinity configuration, run the *"fw ctl affinity" on page 317* command (with applicable parameters).
	■ To see the CoreXL Firewall Instances and which CPU cores they use, run the *"fw ctl multik stat" on page 312* command.
	■ To see all available CPU cores, run:
	`cat /proc/cpuinfo | grep processor`

Parameter	Description
`-r | --reconf`	Applies the existing Multi-Queue policy.

Examples

Show the current Multi-Queue configuration on all interfaces

```
[Expert@MyGW:0]# mq_mng --show

Total 8 cores. Multiqueue 2
cores i/f       type            state           config          cores
--------------------------------------------------------------------------
eth1            igb             Up              Auto            0,4
eth2            igb             Up              Auto            0,4
eth2-01         igb             Up              Auto            0,4
[Expert@MyGW:0]#
```

Show the current Multi-Queue verbose configuration on all interfaces

```
[Expert@MyGW:0]# mq_mng --show -v

Total 8 cores. Multiqueue 2 cores: 0,4
i/f             type            state           config          cores
--------------------------------------------------------------------------
eth1            igb             Up              Auto            0(58),4(78)
eth2            igb             Up              Auto            4(62),0(79)
eth2-01         igb             Up              Auto            0(42),4(86)

core            interfaces      queue                irq        rx packets      tx packets
-------------------------------------------------------------------------------------------
0               eth1            eth1-TxRx-0          58         2350            3012
                eth2            eth2-TxRx-1          79         0               0
                eth2-01         eth2-01-TxRx-0       42         0               45
4               eth1            eth1-TxRx-1          78         652             764
                eth2            eth2-TxRx-0          62         0               0
                eth2-01         eth2-01-TxRx-1       86         0               12
[Expert@MyGW:0]#
```

Show the current Multi-Queue verbose configuration on the interface eth2

```
[Expert@MyGW:0]# mq_mng --show -v -i eth2

Total 8 cores. Multiqueue 2 cores: 0,4
i/f             type            state           config          cores
--------------------------------------------------------------------------------------
eth2            igb             Up              Auto            4(62),0(79)
--------------------------------------------------------------------------------------
eth2 <igb> max 8 cur 2
06:00.2 Ethernet controller: Intel Corporation 82580 Gigabit Network Connection (rev 01)
core            interfaces      queue                irq        rx packets      tx packets
-------------------------------------------------------------------------------------------
0               eth2            eth2-TxRx-1          79         4212            3965
4               eth2            eth2-TxRx-0          62         0               0
[Expert@MyGW:0]#
```

Set automatic Multi-Queue mode on all interfaces

```
mq_mng --set-mode auto
```

Set manual Multi-Queue mode on the interfaces eth1 and eth2 to CPU cores 0, 1, 2, 4, 5, and 6

```
mq_mng -s manual -i eth1 eth2 -c 0-2 4-6
```

Multi-Queue Configuration in Gaia Clish

Syntax

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- You must run these commands in Gaia Clish.
- Change in the Multi-Queue mode can cause short packet loss.

- **To show the existing Multi-Queue configuration for the specified interface:**

```
show interface <Name of Interface> multi-queue [verbose]
```

- **To configure the Multi-Queue for the specified interface:**

```
set interface <Name of Interface> multi-queue
      auto
      manual core <IDs of CPU Cores that run CoreXL SND
Instances>
      off
```

Parameters

Parameter	Description
`<Name of Interface>`	Specifies the interface.
`verbose`	Verbose output that also includes: ■ IRQ numbers for traffic queues ■ Total number of RX and TX packets in traffic queues
`auto`	Configures the automatic Multi-Queue mode (this is the default). Multi-Queue automatically configures the affinity of the specified interface to CPU cores that run CoreXL SND Instances.

Parameter	Description
`manual core <IDs of CPU Cores>`	Configures the manual Multi-Queue mode. Administrator configures the affinity of the specified interface to CPU cores that run CoreXL SND Instances. **Notes:** ■ To specify a specific CPU core, enter its ID number (for example: `manual core 1`). ■ To specify several nonconsecutive CPU cores, enter their ID numbers separated with commas and without spaces (for example: `manual core 1,3`). ■ To specify several consecutive CPU cores, enter their first and last ID numbers separated with a hyphen (for example: `manual core 3-6`). ■ To see the current CoreXL affinity configuration, run the *"fw ctl affinity" on page 317* command (with applicable parameters). ■ To see the CoreXL Firewall Instances and which CPU cores they use, run the *"fw ctl multik stat" on page 312* command. ■ To see all available CPU cores, run: `cat /proc/cpuinfo | grep processor`
`off`	Disables the Multi-Queue on the specified interface.

Examples

Show Multi-Queue configuration on the interface eth2

```
MyGW> show interface eth2 multi-queue

Total 8 cores. Multiqueue 2 cores
i/f             type            state           config          cores
--------------------------------------------------------------------------
eth2            igb             Up              Auto            4,0

Note: The output does not include network interfaces that are currently in the down state.
MyGW>
```

Show Multi-Queue verbose configuration on the interface eth2

```
MyGW> show interface eth2 multi-queue verbose

Total 8 cores. Multiqueue 2 cores: 0,4
i/f             type            state           config          cores
--------------------------------------------------------------------------
eth2            igb             Up              Auto            4(62),0(79)

core            interfaces      queue                irq        rx packets      tx packets
-------------------------------------------------------------------------------------------
0               eth2            eth2-TxRx-1          79         212             80
4               eth2            eth2-TxRx-0          62         16232           18901
MyGW>
```

Set automatic Multi-Queue mode on the interface eth2

```
set interface eth2 multi-queue auto
```

Set manual Multi-Queue mode on the interface eth2 to CPU cores 0, 1, 2, 4, 5, and 6

```
set interface eth2 multi-queue manual core 0-2,4-6
```

Multi-Queue Special Scenarios and Configurations

This section provides instructions for configuring the Multi-Queue in special scenarios.

Default Number of Active RX Queues

Gateway Mode

Changing the number of CoreXL Firewall instances when the Multi-Queue is enabled on some, or all interfaces

For best performance, the Multi-Queue calculates the default number of active RX queues based on this formula:

```
Number of active RX queues = (Number of CPU cores) - (Number of
CoreXL Firewall instances)
```

This configuration is set automatically when you configure the Multi-Queue.

When you change the number of CoreXL Firewall instances, the number of active RX queues changes automatically, if it is not set manually.

VSX Mode

Changing the number of CPU cores, to which the FWK processes are assigned

For best performance, the Multi-Queue calculates the default number of active RX queues based on this formula:

```
Number of active RX queues = The lowest CPU ID, to which an FWK
process is assigned
```

Example

- The number of active RX queues is set to 2.
- This configuration is set automatically when you configure the Multi-Queue.
- It does not automatically update when you change the affinity of Virtual Systems.

```
[Expert@GW:0]# fw ctl affinity -l
Mgmt: CPU 0
eth1-05: CPU 0
eth1-06: CPU 1
VS_0 fwk: CPU 2 3 4 5
VS_1 fwk: CPU 2 3 4 5
[Expert@GW:0]#
```

Adding a Network Interface

When you add a network interface card to a Security Gateway, the Multi-Queue configuration can change due to the way the operating system indexes the interfaces.

If you added a network interface card to a Security Gateway, make sure to either configure the Multi-Queue again, or apply the existing Multi-Queue configuration:

```
mq_mng --reconf
```

Changing the Affinity of CoreXL Firewall instances

Best Practice - For best performance, we recommend that you do **not** assign both CoreXL SND instance and a CoreXL Firewall instance to the same CPU core.

Processing Packets that Arrive in the Wrong Order on an Interface that Works in Monitor Mode

Best Practice - If you enable Multi-Queue on an interface that works in Monitor Mode, then enable the Symmetric Hash for Receive-Side Scaling (RSS). This lets the corresponding interface drivers handle better packets that arrive in the wrong order (for example, TCP [SYN-ACK] that arrives before the TCP [SYN]). As a result, the same CPU core handles the applicable Client-to-Server and Server-to-Client packets.

Follow the instructions in sk101670 to download and run the special shell script `asym2sym.sh` on the Security Gateway or Cluster Members.

Multi-Queue Troubleshooting

Scenario	Explanation and next steps

After reboot, the wrong interfaces are configured for Multi-Queue.	This can happen after changing the physical interfaces on the Security Gateway.
	Follow one of these steps:
	▪ Run:
	`mq_mng --reconf`
	`reboot`
	▪ Configure the Multi-Queue again

After you configure the Multi-Queue and reboot the Security Gateway, some of the configured interfaces show as `Down`. These interfaces were up before the Security Gateway reboot. The "`mq_mng --show`" command shows the interface status as "`Pending on`".	This can happen when not enough IRQs are available on the Security Gateway. Follow one of these steps: ▪ Remove unused expansion cards, if possible ▪ Disable some of the interfaces configured for Multi-Queue
When you change the status of interfaces, all the interface IRQs are assigned to CPU 0, or to all of the CPU cores.	This can happen when an interface status is changed to UP after the automatic affinity procedure runs (during each boot). Run: `mq_mng --reconf`

In VSX mode, an **fwk** process runs on the same CPU core as some of the interface queues.	This can happen when the affinity of the Virtual System was manually changed but Multi-Queue was not reconfigured accordingly.
	Follow one of these steps:
	▪ Run:
	`mq_mng --reconf`
	`reboot`
	▪ Configure the number of active RX queues manually

Scenario	Explanation and next steps
In Gateway mode, after you change the number of CoreXL Firewall instances, the Multi-Queue is disabled on all interfaces.	When you change the number of CoreXL Firewall instances, the number of active RX queues automatically changes based on this formula: `Active RX queues = (Number of CPU cores) - (Number of CoreXL Firewall instances)` If the difference between the number of CPU cores and the number of CoreXL Firewall instances is 1, Multi-Queue is disabled.

CPView

Overview of CPView

Description

CPView is a text based built-in utility on a Check Point computer.

CPView Utility shows statistical data that contain both general system information (CPU, Memory, Disk space) and information for different Software Blades (only on Security Gateway).

The CPView continuously updates the data in easy to access views.

On Security Gateway, you can use this statistical data to monitor the performance.

For more information, see sk101878.

Syntax

```
cpview --help
```

CPView User Interface

The CPView user interface has three sections:

Section	Description
Header	This view shows the time the statistics in the third view are collected. It updates when you refresh the statistics.
Navigation	This menu bar is interactive. Move between menus with the arrow keys and mouse. A menu can have sub-menus and they show under the menu bar.
View	This view shows the statistics collected in that view. These statistics update at the refresh rate.

Using CPView

Use these keys to navigate the CPView:

Key	Description
Arrow keys	Moves between menus and views. Scrolls in a view.
Home	Returns to the **Overview** view.
Enter	Changes to the **View Mode**. On a menu with sub-menus, the **Enter** key moves you to the lowest level sub-menu.
Esc	Returns to the **Menu Mode**.
Q	Quits CPView.

Use these keys to change CPView interface options:

Key	Description
R	Opens a window where you can change the refresh rate. The default refresh rate is 2 seconds.
W	Changes between wide and normal display modes. In wide mode, CPView fits the screen horizontally.
S	Manually sets the number of rows or columns.
M	Switches on/off the mouse.
P	Pauses and resumes the collection of statistics.

Use these keys to save statistics, show help, and refresh statistics:

Key	Description
C	Saves the current page to a file. The file name format is: `cpview_`*`<ID of the cpview process>`*`.cap`*`<Number of the capture>`*
H	Shows a tooltip with CPView options.
Space bar	Immediately refreshes the statistics.

CPU Spike Detective

The CPU Spike Detective is a tool that monitors the CPU utilization and saves information about the CPU utilization spikes it detects.

This tool does **not** impact the performance.

Use these commands in Gaia Clish:

```
show spike-detective[ESC][ESC]
```

```
set spike-detective[ESC][ESC]
```

```
delete spike-detective[ESC][ESC]
```

For more information, see sk166454.

Command Line Reference

See the *R81 CLI Reference Guide*.

Working with Kernel Parameters on Security Gateway

This section describes what are kernel parameters, and how to view and configure their values.

Introduction to Kernel Parameters

Kernel parameters let you change the advanced behavior of your Security Gateway.

These are the supported types of kernel parameters:

Type	Description
Integer	Accepts only one integer value.
String	Accepts only a plain-text string.

Important:

- In Cluster, you must see and configure the same value for the same kernel parameter on *each* Cluster Member.
- In VSX Gateway, the configured values of kernel parameters apply to all existing Virtual Systems and Virtual Routers.

Security Gateway gets the names and the default values of the kernel parameters from these kernel module files:

- `$FWDIR/boot/modules/fw_kern_64.o`
- `$FWDIR/boot/modules/fw_kern_64_v6.o`
- `$PPKDIR/boot/modules/sim_kern_64.o`
- `$PPKDIR/boot/modules/sim_kern_64_v6.o`

Firewall Kernel Parameters

To change the internal default behavior of Firewall or to configure special advanced settings for Firewall, you can use Firewall kernel parameters.

The names of applicable Firewall kernel parameters and their values appear in various SK articles in *Check Point Support Center*, and provided by *Check Point Support*.

ⓘ **Important:**

- The names of Firewall kernel parameters are case-sensitive.
- You can configure most of the Firewall kernel parameters on-the-fly with the "`fw ctl set`" command.
  This change does **not** survive a reboot.
  You can use the "`fw ctl set -f`" command to make this change permanent as well.
- You can configure some of the Firewall kernel parameters only permanently in the special configuration file `$FWDIR/boot/modules/fwkern.conf` command.
  This requires a maintenance window, because the new values of the kernel parameters take effect only after a reboot.
- You can configure some of the Firewall kernel parameters only permanently in the special configuration files - `$FWDIR/boot/modules/fwkern.conf` or `$FWDIR/boot/modules/vpnkern.conf`.
  You must manually edit these files.
  This requires a maintenance window, because the new values of the kernel parameters take effect only after a reboot.
- In a Cluster, you must configure all the Cluster Members in the same way.
- On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.

Examples of Firewall kernel parameters

Type	Name

Integer	`fw_allow_simultaneous_ping`
	`fw_kdprintf_limit`
	`fw_log_bufsize`
	`send_buf_limit`

String	`simple_debug_filter_addr_1`
	`simple_debug_filter_daddr_1`
	`simple_debug_filter_vpn_1`
	`ws_debug_ip_str`
	`fw_lsp_pair1`

Working with Integer Kernel Parameters

Viewing the list of the available Firewall *integer* kernel parameters and their values

<table>
<tr><th>Step</th><th>Instructions</th></tr>
<tr><td>1</td><td>Connect to the command line on your Security Gateway / each Cluster Member.
<b>Note</b> - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.</td></tr>
<tr><td>2</td><td>Log in to the Expert mode.</td></tr>
<tr><td>3</td><td>Make sure you can get the list of the available integer kernel parameters and their values without errors:
<b>Note</b> - The configuration of your Security Gateway might not support all kernel parameters. As a result, the Security Gateway might fail to get the value of some kernel parameters.<pre>modinfo -p $FWDIR/boot/modules/fw_kern*.o | sort -u |
grep ':int param' | awk 'BEGIN {FS=":"} ; {print $1}' |
xargs -n 1 fw ctl get int</pre></td></tr>
<tr><td>4</td><td>If in the previous step there were <b>no</b> errors, get the list of the available integer kernel parameters and their values, and save the list to a file:<pre>modinfo -p $FWDIR/boot/modules/fw_kern*.o | sort -u |
grep ':int param' | awk 'BEGIN {FS=":"} ; {print $1}' |
xargs -n 1 fw ctl get int 1>> /var/log/fw_integer_
kernel_parameters.txt 2>> /var/log/fw_integer_kernel_
parameters.txt</pre></td></tr>
<tr><td>5</td><td>Analyze the output file:<pre>/var/log/fw_integer_kernel_parameters.txt</pre></td></tr>
</table>

Viewing the current value of a Firewall *integer* kernel parameter

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

3	Get the current value of an integer kernel parameter:
	■ On the Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get int <Name of Integer Kernel Parameter> [-a]`
	■ On the Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get int <Name of Integer Kernel Parameter> [-a]`
	■ On the Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get int <Name of Integer Kernel Parameter> [-a]`
	Example:
	`[Expert@MyGW:0]# fw ctl get int send_buf_limit`
	`send_buf_limit = 80`
	`[Expert@MyGW:0]#`

Configuring a value for a Firewall *integer* kernel parameter *temporarily*

Important - This change does **not** survive reboot.

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

3	Configure the new value for an integer kernel parameter:
	■ On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl set int <Name of Integer Kernel Parameter> <Integer Value>`
	■ On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl set int <Name of Integer Kernel Parameter> <Integer Value>`
	■ On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl set int <Name of Integer Kernel Parameter> <Integer Value>`
	Example:
	`[Expert@MyGW:0]# fw ctl set int send_buf_limit 100`
	`Set operation succeeded`
	`[Expert@MyGW:0]#`

Step	Instructions

4	Make sure the new value is configured.
	■ On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get int <Name of Integer Kernel Parameter>`
	■ On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get int <Name of Integer Kernel Parameter>`
	■ On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get int <Name of Integer Kernel Parameter>`
	Example:
	`[Expert@MyGW:0]# fw ctl get int send_buf_limit`
	`send_buf_limit = 100`
	`[Expert@MyGW:0]#`

Configuring a value for a Firewall *integer* kernel parameter *permanently*

To make a kernel parameter configuration permanent (to survive reboot), you must edit one of the applicable configuration files:

- For Firewall kernel parameters:

  ```
  $FWDIR/boot/modules/fwkern.conf
  ```

- For VPN kernel parameters:

  ```
  $FWDIR/boot/modules/vpnkern.conf
  ```

The exact parameters appear in various SK articles in *Check Point Support Center*, and provided by *Check Point Support*.

Short procedure for the "fwkern.conf" file

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to the Expert mode.

3	Back up the current configuration file, if it exists:
	■ On the Security Gateway (each Cluster Member), run:
	`cp -v $FWDIR/boot/modules/fwkern.conf{,_BKP}`
	■ On the Scalable Platform Security Group, run:
	`g_cp -v $FWDIR/boot/modules/fwkern.conf{,_BKP}`

Step	Instructions

4	Configure the required Firewall kernel parameter with the assigned value in the exact format specified below.
	■ On the Security Gateway (each Cluster Member), run:
	`fw ctl set -f int <Name_of_Integer_Kernel_Parameter> <Integer_Value>`
	■ On the Scalable Platform Security Group, run **one** of these commands:
	`g_fw ctl set -f int <Name_of_Integer_Kernel_Parameter> <Integer_Value>`
	`g_update_conf_file fwkern.conf <Name_of_Integer_Kernel_Parameter>=<Integer_Value>`
	Example:
	`[Expert@MyGW:0]# fw ctl set -f int send_buf_limit 100`
	`"fwkern.conf" was updated successfully`
	`[Expert@MyGW:0]#`
	`[Expert@MyGW:0]# g_update_conf_file fwkern.conf send_buf_limit=100`
	`"fwkern.conf" was updated successfully`
	`[Expert@MyGW:0]#`

5	Examine the configuration file.
	■ On the Security Gateway (each Cluster Member), run:
	`cat $FWDIR/boot/modules/fwkern.conf`
	■ On the Scalable Platform Security Group, run:
	`g_cat $FWDIR/boot/modules/fwkern.conf`

6	Reboot.
	■ On the Security Gateway / Cluster Member, run:
	`reboot`
	Important - In cluster, this can cause a failover.
	■ On the Scalable Platform Security Group, run:
	`g_reboot -a`

Step	Instructions
7	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
8	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

9	Make sure the new value of the kernel parameter is configured.
	■ On a Security Gateway / each Cluster Member, run:
	`fw ctl get int <Name of Integer Kernel Parameter> [-a]`
	■ On a Scalable Platform Security Group, run:
	`g_fw ctl get int <Name of Integer Kernel Parameter> [-a]`

Long procedure for the "fwkern.conf" and "vpnkern.conf" files

For more information, see sk26202: Changing the kernel global parameters for Check Point Security Gateway.

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to the Expert mode.

Step	Instructions

3	See if the configuration file already exists.
	■ On a Security Gateway / each Cluster Member:
	• For Firewall kernel parameters, run:
	`ls -l $FWDIR/boot/modules/fwkern.conf`
	• For VPN kernel parameters, run:
	`ls -l $FWDIR/boot/modules/vpnkern.conf`
	■ On a Scalable Platform Security Group:
	• For Firewall kernel parameters, run:
	`g_ls -l $FWDIR/boot/modules/fwkern.conf`
	• For VPN kernel parameters, run:
	`g_ls -l $FWDIR/boot/modules/vpnkern.conf`

4	If this file already exists, skip to **Step 5**.
	If this file does not exist, then create it manually and then skip to **Step 6**.
	■ On a Security Gateway / each Cluster Member:
	• For Firewall kernel parameters, run:
	`touch $FWDIR/boot/modules/fwkern.conf`
	• For VPN kernel parameters, run:
	`touch $FWDIR/boot/modules/fwkern.conf`
	■ On a Scalable Platform Security Group:
	• For Firewall kernel parameters, run:
	`g_all touch $FWDIR/boot/modules/fwkern.conf`
	• For VPN kernel parameters, run:
	`g_all touch $FWDIR/boot/modules/vpnkern.conf`

Step	Instructions

5	Back up the current configuration file.
	■ On a Security Gateway / each Cluster Member:
	• For Firewall kernel parameters, run:
	`cp -v $FWDIR/boot/modules/fwkern.conf{,_BKP}`
	• For VPN kernel parameters, run:
	`cp -v $FWDIR/boot/modules/vpnkern.conf{,_BKP}`
	■ On a Scalable Platform Security Group:
	• For Firewall kernel parameters, run:
	`g_cp -v $FWDIR/boot/modules/fwkern.conf{,_BKP}`
	• For VPN kernel parameters, run:
	`g_cp -v $FWDIR/boot/modules/vpnkern.conf{,_BKP}`

6	Edit the current configuration file.
	The same syntax applies to the Security Gateway / each Cluster Member and the Scalable Platform Security Group:
	■ For Firewall kernel parameters, run:
	`vi $FWDIR/boot/modules/fwkern.conf`
	■ For VPN kernel parameters, run:
	`vi $FWDIR/boot/modules/vpnkern.conf`

7	Add the required Firewall kernel parameter with the assigned value in the exact format specified below. **Important** - These configuration files do not support space characters, tabulation characters, and comments (lines that contain the # character). `<Name_of_Integer_Kernel_Parameter>=<Integer_Value>`
8	Save the changes in the file and exit the editor.

Step	Instructions

9	On the Scalable Platform Security Group, copy the updated configuration file to all other Security Group Members:
	- For Firewall kernel parameters, run:
	`asg_cp2blades $FWDIR/boot/modules/fwkern.conf`
	- For VPN kernel parameters, run:
	`asg_cp2blades $FWDIR/boot/modules/vpnkern.conf`

10	Reboot.
	- On the Security Gateway / Cluster Member, run:
	`reboot`
	Important - In cluster, this can cause a failover.
	- On the Scalable Platform Security Group, run:
	`g_reboot -a`

11	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
12	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

13	Make sure the new value of the kernel parameter is configured.
	- On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get int <Name of Integer Kernel Parameter> [-a]`
	- On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get int <Name of Integer Kernel Parameter> [-a]`
	- On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get int <Name of Integer Kernel Parameter> [-a]`

Working with String Kernel Parameters

Viewing the list of the available Firewall *string* kernel parameters and their values

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to the Expert mode.
3	Make sure you can get the list of the available integer kernel parameters and their values without errors: **Note** - The configuration of your Security Gateway might not support all kernel parameters. As a result, the Security Gateway might fail to get the value of some kernel parameters. `modinfo -p $FWDIR/boot/modules/fw_kern*.o | sort -u | grep ':string param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get str`
4	If in the previous step there were **no** errors, get the list of the available string kernel parameters and their values, and save the list to a file: `modinfo -p $FWDIR/boot/modules/fw_kern*.o | sort -u | grep ':string param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get str 1>> /var/log/fw_string_kernel_parameters.txt 2>> /var/log/fw_string_kernel_parameters.txt`
5	Analyze the output file: `/var/log/fw_string_kernel_parameters.txt`

Viewing the current value of a Firewall *string* kernel parameter

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

3	Get the current value of a string kernel parameter:
	■ On the Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get str <Name of String Kernel Parameter> [-a]`
	■ On the Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get str <Name of String Kernel Parameter> [-a]`
	■ On the Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get str <Name of String Kernel Parameter> [-a]`
	Example:
	`[Expert@MyGW:0]# fw ctl get str fileapp_default_encoding_charset`
	`fileapp_default_encoding_charset = 'UTF-8'`
	`[Expert@MyGW:0]#`

Configuring a value for a Firewall *string* kernel parameter *temporarily*

Important - This change does **not** survive reboot.

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

Step	Instructions

3	Configure the new value for a string kernel parameter.
	Note - You must write the value in single quotes, or double quotes. Use **one** of these syntax options.
	■ On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl set str <Name of String Kernel Parameter> '<String Text>'`
	or
	`fw ctl set str <Name of String Kernel Parameter> "<String Text>"`
	■ On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl set str <Name of String Kernel Parameter> '<String Text>'`
	or
	`fw ctl set str <Name of String Kernel Parameter> "<String Text>"`
	■ On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl set str <Name of String Kernel Parameter> '<String Text>'`
	or
	`g_fw ctl set str <Name of String Kernel Parameter> "<String Text>"`
	Example:
	`[Expert@MyGW:0]# fw ctl set str debug_filter_saddr_ip '1.1.1.1'`
	`Set operation succeeded`
	`[Expert@MyGW:0]#`

Step	Instructions

4	Make sure the new value is configured.
	■ On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get str <Name of String Kernel Parameter>`
	■ On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get str <Name of String Kernel Parameter>`
	■ On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get str <Name of String Kernel Parameter>`
	Example:
	`[Expert@MyGW:0]# fw ctl get str debug_filter_saddr_ip`
	`debug_filter_saddr_ip = '1.1.1.1'`
	`[Expert@MyGW:0]#`

Configuring a value for a Firewall *string* kernel parameter *permanently*

To make a kernel parameter configuration permanent (to survive reboot), you must edit one of the applicable configuration files:

- For Firewall kernel parameters:

  `$FWDIR/boot/modules/fwkern.conf`

- For VPN kernel parameters:

  `$FWDIR/boot/modules/vpnkern.conf`

The exact parameters appear in various SK articles in *Check Point Support Center*, and provided by *Check Point Support*.

Short procedure for the "fwkern.conf" file

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to the Expert mode.

3	Back up the current configuration file, if it exists:
	■ On the Security Gateway (each Cluster Member), run:
	`cp -v $FWDIR/boot/modules/fwkern.conf{,_BKP}`
	■ On the Scalable Platform Security Group, run:
	`g_cp -v $FWDIR/boot/modules/fwkern.conf{,_BKP}`

Step	Instructions

4	Configure the required Firewall kernel parameter with the assigned value in the exact format specified below.
	Note - You must write the value in single quotes, or double quotes. Use **one** of these syntax options.
	■ On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl set -f str <Name_of_String_Kernel_Parameter> '<String_Text>'`
	or
	`fw ctl set -f str <Name_of_String_Kernel_Parameter> "<String_Text>"`
	■ On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl set -f str <Name_of_String_Kernel_Parameter> '<String_Text>'`
	or
	`g_fw ctl set -f str <Name_of_String_Kernel_Parameter> "<String_Text>"`
	Example:
	`[Expert@MyGW:0]# fw ctl set -f str ws_debug_ip_str '1.1.1.1'`
	`"fwkern.conf" was updated successfully`
	`[Expert@MyGW:0]#`

5	Examine the configuration file.
	■ On the Security Gateway / each Cluster Member, run:
	`cat $FWDIR/boot/modules/fwkern.conf`
	■ On the Scalable Platform Security Group, run:
	`g_cat $FWDIR/boot/modules/fwkern.conf`

Step	Instructions

6	Reboot.
	■ On the Security Gateway / Cluster Member, run:
	`reboot`
	Important - In cluster, this can cause a failover.
	■ On the Scalable Platform Security Group, run:
	`g_reboot -a`

7	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
8	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

9	Make sure the new value of the kernel parameter is configured.
	■ On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get str <Name of String Kernel Parameter> [-a]`
	■ On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get str <Name of String Kernel Parameter> [-a]`
	■ On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get str <Name of String Kernel Parameter> [-a]`

Long procedure for the "fwkern.conf" and "vpnkern.conf" files

For more information, see sk26202: Changing the kernel global parameters for Check Point Security Gateway.

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to the Expert mode.

3	■ On a Security Gateway / each Cluster Member:
	• For Firewall kernel parameters, run:
	`ls -l $FWDIR/boot/modules/fwkern.conf`
	• For VPN kernel parameters, run:
	`ls -l $FWDIR/boot/modules/vpnkern.conf`
	■ On a Scalable Platform Security Group:
	• For Firewall kernel parameters, run:
	`g_ls -l $FWDIR/boot/modules/fwkern.conf`
	• For VPN kernel parameters, run:
	`g_ls -l $FWDIR/boot/modules/vpnkern.conf`

4	If this file already exists, skip to **Step 5**.
	If this file does not exist, then create it manually and then skip to **Step 6**.
	■ On a Security Gateway / each Cluster Member:
	• For Firewall kernel parameters, run:
	`touch $FWDIR/boot/modules/fwkern.conf`
	• For VPN kernel parameters, run:
	`touch $FWDIR/boot/modules/fwkern.conf`
	■ On a Scalable Platform Security Group:
	• For Firewall kernel parameters, run:
	`g_all touch $FWDIR/boot/modules/fwkern.conf`
	• For VPN kernel parameters, run:
	`g_all touch $FWDIR/boot/modules/vpnkern.conf`

Step	Instructions

5	Back up the current configuration file.
	■ On a Security Gateway / each Cluster Member:
	• For Firewall kernel parameters, run:
	`cp -v $FWDIR/boot/modules/fwkern.conf{,_BKP}`
	• For VPN kernel parameters, run:
	`cp -v $FWDIR/boot/modules/vpnkern.conf{,_BKP}`
	■ On a Scalable Platform Security Group:
	• For Firewall kernel parameters, run:
	`g_cp -v $FWDIR/boot/modules/fwkern.conf{,_BKP}`
	• For VPN kernel parameters, run:
	`g_cp -v $FWDIR/boot/modules/vpnkern.conf{,_BKP}`

6	Edit the current configuration file.
	The same syntax applies to the Security Gateway / each Cluster Member and the Scalable Platform Security Group:
	■ For Firewall kernel parameters, run:
	`vi $FWDIR/boot/modules/fwkern.conf`
	■ For VPN kernel parameters, run:
	`vi $FWDIR/boot/modules/vpnkern.conf`

7	Add the required kernel parameter with the assigned value in the exact format specified below.
	Important - These configuration files do not support space characters, tabulation characters, and comments (lines that contain the # character).
	Note - You must write the value in single quotes, or double quotes. Use **one** of these syntax options.
	`<Name_of_String_Kernel_Parameter>='<String_Text>'`
	or
	`<Name_of_String_Kernel_Parameter>="<String_Text>"`

8	Save the changes in the file and exit the editor.

Step	Instructions

9	On the Scalable Platform Security Group, copy the updated configuration file to all other Security Group Members:
	- For Firewall kernel parameters, run:
	`asg_cp2blades $FWDIR/boot/modules/fwkern.conf`
	- For VPN kernel parameters, run:
	`asg_cp2blades $FWDIR/boot/modules/vpnkern.conf`

10	Reboot.
	- On the Security Gateway / Cluster Member, run:
	`reboot`
	Important - In cluster, this can cause a failover.
	- On the Scalable Platform Security Group, run:
	`g_reboot -a`

11	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
12	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

13	Make sure the new value of the kernel parameter is configured.
	- On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get str <Name of String Kernel Parameter> [-a]`
	- On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get str <Name of String Kernel Parameter> [-a]`
	- On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get str <Name of String Kernel Parameter> [-a]`

Removing the current value from a Firewall *string* kernel parameter *temporarily*

Important - This change does **not** survive reboot.

Step	Instructions
1	Connect to the command line on your Security Gateway or Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

3	Clear the current value from a string kernel parameter:
	Note - You must set an empty value in single quotes, or double quotes. Use **one** of these syntax options.
	■ On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl set str '<Name of String Kernel Parameter>'`
	or
	`fw ctl set str "<Name of String Kernel Parameter>"`
	■ On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl set str '<Name of String Kernel Parameter>'`
	or
	`fw ctl set str "<Name of String Kernel Parameter>"`
	■ On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl set str '<Name of String Kernel Parameter>'`
	or
	`g_fw ctl set str "<Name of String Kernel Parameter>"`
	Example:
	`[Expert@MyGW:0]# fw ctl set str debug_filter_saddr_ip ''`
	`Set operation succeeded`
	`[Expert@MyGW:0]#`

Step	Instructions

4	Make sure the value is cleared (the new value is empty):
	- On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get str <Name of String Kernel Parameter>`
	- On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get str <Name of String Kernel Parameter>`
	- On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get str <Name of String Kernel Parameter>`
	Example:
	`[Expert@MyGW:0]# fw ctl get str debug_filter_saddr_ip`
	`debug_filter_saddr_ip = ''`
	`[Expert@MyGW:0]#`

SecureXL Kernel Parameters

To change the internal default behavior of SecureXL or to configure special advanced settings for SecureXL, you can use SecureXL kernel parameters.

The names of applicable SecureXL kernel parameters and their values appear in various SK articles in *Check Point Support Center*, and provided by *Check Point Support*.

Important:

- The names of SecureXL kernel parameters are case-sensitive.
- You can configure SecureXL kernel parameters in the current session with the "`fw ctl set`" command.
  This change does **not** survive reboot.
- To configure SecureXL kernel parameters permanently, you must configure them in the special configuration file - `$PPKDIR/conf/simkern.conf`
  Schedule a maintenance window, because this procedure requires a reboot.
- For some SecureXL kernel parameters, you **cannot** get their current value on-the-fly with the "`fw ctl get`" command (see sk43387).
- In a Cluster, you must configure all the Cluster Members in the same way.
- On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.

Examples of SecureXL kernel parameters

Type	Name

Integer	`num_of_sxl_devices`
	`sim_ipsec_dont_fragment`
	`tcp_always_keepalive`
	`sim_log_all_frags`
	`simple_debug_filter_dport_1`
	`simple_debug_filter_proto_1`

String	`simple_debug_filter_addr_1`
	`simple_debug_filter_daddr_2`
	`simlinux_excluded_ifs_list`

Working with Integer Kernel Parameters

Viewing the list of the available SecureXL *integer* kernel parameters and their values

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to the Expert mode.
3	Make sure you can get the list of the available integer kernel parameters and their values without errors: **Note** - The configuration of your Security Gateway might not support all kernel parameters. As a result, the Security Gateway might fail to get the value of some kernel parameters. `modinfo -p $PPKDIR/boot/modules/sim_kern*.o | sort -u | grep ':int param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get int`
4	If in the previous step there were **no** errors, get the list of the available integer kernel parameters and their values, and save the list to a file: `modinfo -p $PPKDIR/boot/modules/sim_kern*.o | sort -u | grep ':int param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get int 1>> /var/log/sxl_integer_kernel_parameters.txt 2>> /var/log/sxl_integer_kernel_parameters.txt`
5	Analyze the output file: `/var/log/sxl_integer_kernel_parameters.txt`

Viewing the current value of a SecureXL *integer* kernel parameter

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

3	Get the current value of an integer kernel parameter:
	■ On the Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get int <Name of Integer Kernel Parameter> [-a]`
	■ On the Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get int <Name of Integer Kernel Parameter> [-a]`
	■ On the Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get int <Name of Integer Kernel Parameter> [-a]`
	Example:
	`[Expert@MyGW:0]# fw ctl get int sim_ipsec_dont_fragment`
	`sim_ipsec_dont_fragment = 1`
	`[Expert@MyGW:0]#`

Configuring a value for a SecureXL *integer* kernel parameter *temporarily*

Important - This change does **not** survive reboot.

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

3	Configure the new value for an integer kernel parameter:
	- On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl set int <Name of Integer Kernel Parameter> <Integer Value>`
	- On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl set int <Name of Integer Kernel Parameter> <Integer Value>`
	- On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl set int <Name of Integer Kernel Parameter> <Integer Value>`
	Example:
	`[Expert@MyGW:0]# fw ctl set int sim_ipsec_dont_fragment 0`
	`Set operation succeeded`
	`[Expert@MyGW:0]#`

Step	Instructions

4	Make sure the new value is configured.
	■ On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get int <Name of Integer Kernel Parameter>`
	■ On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get int <Name of Integer Kernel Parameter>`
	■ On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get int <Name of Integer Kernel Parameter>`
	Example:
	`[Expert@MyGW:0]# fw ctl get int sim_ipsec_dont_fragment`
	`sim_ipsec_dont_fragment = 0`
	`[Expert@MyGW:0]#`

Configuring a value for a SecureXL *integer* kernel parameter *permanently*

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to the Expert mode.

3	See if the configuration file already exists.
	■ On a Security Gateway / each Cluster Member, run:
	`ls -l $PPKDIR/conf/simkern.conf`
	■ On a Scalable Platform Security Group, run:
	`g_ls -l $PPKDIR/conf/simkern.conf`

4	If this file already exists, skip to **Step 5**.
	If this file does not exist, then create it manually and then skip to **Step 6**:
	■ On a Security Gateway / each Cluster Member, run:
	`touch $PPKDIR/conf/simkern.conf`
	■ On a Scalable Platform Security Group, run:
	`g_all touch $PPKDIR/conf/simkern.conf`

5	Back up the current configuration file.
	■ On a Security Gateway / each Cluster Member, run:
	`cp -v $PPKDIR/conf/simkern.conf{,_BKP}`
	■ On a Scalable Platform Security Group, run:
	`g_cp -v $PPKDIR/conf/simkern.conf{,_BKP}`

6	Edit the current configuration file. The same syntax applies to the Security Gateway / each Cluster Member and the Scalable Platform Security Group: `vi $PPKDIR/conf/simkern.conf`

Step	Instructions
7	Add the required SecureXL kernel parameter with the assigned value in the exact format specified below. **Important** - This configuration file does not support space characters, tabulation characters, and comments (lines that contain the # character). `<Name_of_SecureXL_Integer_Kernel_Parameter>=<Integer_Value>`
8	Save the changes in the file and exit the editor.

9	Reboot.
	■ On the Security Gateway / Cluster Member, run:
	`reboot`
	Important - In cluster, this can cause a failover.
	■ On the Scalable Platform Security Group, run:
	`g_reboot -a`

10	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
11	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

12	Make sure the new value of the kernel parameter is configured.
	■ On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get int <Name of Integer Kernel Parameter> [-a]`
	■ On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get int <Name of Integer Kernel Parameter> [-a]`
	■ On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get int <Name of Integer Kernel Parameter> [-a]`

Working with String Kernel Parameters

Viewing the list of the available SecureXL *string* kernel parameters and their values

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to the Expert mode.
3	Make sure you can get the list of the available integer kernel parameters and their values without errors: **Note** - The configuration of your Security Gateway might not support all kernel parameters. As a result, the Security Gateway might fail to get the value of some kernel parameters. `modinfo -p $PPKDIR/boot/modules/sim_kern*.o | sort -u | grep ':string param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get str`
4	If in the previous step there were **no** errors, get the list of the available string kernel parameters and their values, and save the list to a file: `modinfo -p $PPKDIR/boot/modules/sim_kern*.o | sort -u | grep ':string param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get str 1>> /var/log/sxl_string_kernel_parameters.txt 2>> /var/log/sxl_string_kernel_parameters.txt`
5	Analyze the output file: `/var/log/sxl_string_kernel_parameters.txt`

Viewing the current value of a SecureXL *string* kernel parameter

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

3	Get the current value of an integer kernel parameter:
	- On the Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get str <Name of Integer Kernel Parameter> [-a]`
	- On the Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get str <Name of Integer Kernel Parameter> [-a]`
	- On the Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get str <Name of Integer Kernel Parameter> [-a]`
	Example:
	`[Expert@MyGW:0]# fw ctl get str fwkdebug_print_connkey_on_str`
	`fwkdebug_print_connkey_on_str = ''`
	`[Expert@MyGW:0]#`

Configuring a value for a SecureXL *string* kernel parameter *temporarily*

Important - This change does **not** survive reboot.

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

3	Configure the new value for a string kernel parameter.
	Note - You must write the value in single quotes, or double quotes. Use **one** of these syntax options.
	■ On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl set str <Name of String Kernel Parameter> '<String Text>'`
	or
	`fw ctl set str <Name of String Kernel Parameter> "<String Text>"`
	■ On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl set str <Name of String Kernel Parameter> '<String Text>'`
	or
	`fw ctl set str <Name of String Kernel Parameter> "<String Text>"`
	■ On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl set str <Name of String Kernel Parameter> '<String Text>'`
	or
	`g_fw ctl set str <Name of String Kernel Parameter> "<String Text>"`
	Example:
	`[Expert@MyGW:0]# fw ctl set str fwkdebug_print_connkey_on_str 'Packet accepted'`
	`Set operation succeeded`
	`[Expert@MyGW:0]#`

Step	Instructions

4	Make sure the new value is configured.
	■ On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get str <Name of String Kernel Parameter>`
	■ On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get str <Name of String Kernel Parameter>`
	■ On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get str <Name of String Kernel Parameter>`
	Example:
	`[Expert@MyGW:0]# fw ctl get str fwkdebug_print_connkey_on_str`
	`fwkdebug_print_connkey_on_str = 'Packet accepted'`
	`[Expert@MyGW:0]#`

Configuring a value for a SecureXL *string* kernel parameter *permanently*

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to the Expert mode.

3	See if the configuration file already exists.
	■ On a Security Gateway / each Cluster Member, run:
	`ls -l $PPKDIR/conf/simkern.conf`
	■ On a Scalable Platform Security Group, run:
	`g_ls -l $PPKDIR/conf/simkern.conf`

4	If this file already exists, skip to **Step 5**.
	If this file does not exist, then create it manually and then skip to **Step 6**:
	■ On a Security Gateway / each Cluster Member, run:
	`touch $PPKDIR/conf/simkern.conf`
	■ On a Scalable Platform Security Group, run:
	`g_all touch $PPKDIR/conf/simkern.conf`

5	Back up the current configuration file.
	■ On a Security Gateway / each Cluster Member, run:
	`cp -v $PPKDIR/conf/simkern.conf{,_BKP}`
	■ On a Scalable Platform Security Group, run:
	`g_cp -v $PPKDIR/conf/simkern.conf{,_BKP}`

6	Edit the current configuration file. The same syntax applies to the Security Gateway / each Cluster Member and the Scalable Platform Security Group: `vi $PPKDIR/conf/simkern.conf`

Step	Instructions

7	Add the required SecureXL kernel parameter with the assigned value in the exact format specified below.
	Important - This configuration file does not support space characters, tabulation characters, and comments (lines that contain the # character).
	Note - You must write the value in single quotes, or double quotes. Use **one** of these syntax options.
	`<Name_of_SecureXL_String_Kernel_Parameter>='<String_Text>'`
	or
	`<Name_of_SecureXL_String_Kernel_Parameter>="<String_Text>"`

8	Save the changes in the file and exit the editor.

9	Reboot.
	■ On the Security Gateway / Cluster Member, run:
	`reboot`
	Important - In cluster, this can cause a failover.
	■ On the Scalable Platform Security Group, run:
	`g_reboot -a`

10	Connect to the command line on your Security Gateway / each Cluster Member. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
11	Log in to Gaia Clish or the Expert mode. **Note** - On Scalable Platforms (Maestro and Chassis), you must use Gaia gClish or the Expert mode.

Step	Instructions

12	Make sure the new value of the kernel parameter is configured.
	- On a Security Gateway / each Cluster Member, run in Gaia Clish or the Expert mode:
	`fw ctl get str <Name of String Kernel Parameter> [-a]`
	- On a Scalable Platform Security Group, run in Gaia gClish:
	`fw ctl get str <Name of String Kernel Parameter> [-a]`
	- On a Scalable Platform Security Group, run in the Expert mode:
	`g_fw ctl get str <Name of String Kernel Parameter> [-a]`

Kernel Debug on Security Gateway

This section describes how to collect a kernel debug on Security Gateway.

Kernel Debug Syntax

Description:

During a kernel debug session, Security Gateway prints special debug messages that help Check Point Support and R&D understand how the Security Gateway processes the applicable connections.

Important - In Cluster, you must configure and perform the kernel debug procedure on all cluster members in the same way.

Action plan to collect a kernel debug:

Note - See the *"Kernel Debug Procedure" on page 432*, or the *"Kernel Debug Procedure with Connection Life Cycle" on page 436*.

Step	Action	Description
1	Configure the applicable debug settings: a. Restore the default settings. b. Allocate the debug buffer.	In this step, you prepare the kernel debug options: a. Restore the default debug settings, so that any other debug settings do not interfere with the kernel debug. b. Allocate the kernel debug buffer, in which Security Gateway holds the applicable debug messages.
2	Configure the applicable kernel debug modules and their debug flags.	In this step, you prepare the applicable kernel debug modules and their debug flags, so that Security Gateway collects only applicable debug messages.
3	Start the collection of the kernel debug into an output file.	In this step, you configure Security Gateway to write the debug messages from the kernel debug buffer into an output file.
4	Stop the kernel debug.	In this step, you configure Security Gateway to stop writing the debug messages into an output file.
5	Restore the default kernel debug settings.	In this step, you restore the default kernel debug options.

To see the built-in help for the kernel debug

```
fw ctl debug -h
```

To restore the default kernel debug settings

- To reset all debug flags and enable only the default debug flags in all kernel modules:

  ```
  fw ctl debug 0
  ```

- To disable all debug flags including the default flags in all kernel modules:

  Best Practice - Do **not** run this command, because it disables even the basic default debug messages.

  ```
  fw ctl debug -x
  ```

To allocate the kernel debug buffer

```
fw ctl debug -buf 8200 [-v {"<List of VSIDs>" | all}] [-k]
```

Notes:

- Security Gateway allocates the kernel debug buffer with the specified size for *every* CoreXL Firewall instance.
- The maximal supported buffer size is 8192 kilobytes..

To configure the debug modules and debug flags

- General syntax:

  ```
  fw ctl debug [-d <Strings to Search>] [-v {"<List of VSIDs>"
  | all}] -m <Name of Debug Module> {all | + <List of Debug
  Flags> | - <List of Debug Flags>}
  ```

  ```
  fw ctl debug [-s "<String to Stop Debug>"] [-v {"<List of
  VSIDs>" | all}] -m <Name of Debug Module> {all | + <List of
  Debug Flags> | - <List of Debug Flags>}
  ```

- To see a list of all debug modules and their flags:

  Note - The list of kernel modules depends on the Software Blades you enabled on the Security Gateway.

  ```
  fw ctl debug -m
  ```

- To see a list of debug flags that are already enabled:

  ```
  fw ctl debug
  ```

- To enable all debug flags in the specified kernel module:

```
fw ctl debug -m <Name of Debug Module> all
```

- To enable the specified debug flags in the specified kernel module:

```
fw ctl debug -m <Name of Debug Module> + <List of Debug
Flags>
```

- To disable the specified debug flags in the specified kernel module:

```
fw ctl debug -m <Name of Debug Module> - <List of Debug
Flags>
```

To collect the kernel debug output

- General syntax (only supported parameters are listed):

```
fw ctl kdebug [-p <List of Fields>] [-T] -f > /<Path>/<Name
of Output File>
```

```
fw ctl kdebug [-p <List of Fields>] [-T] -f -o /<Path>/<Name
of Output File> -m <Number of Cyclic Files> [-s <Size of
Each Cyclic File in KB>]
```

- To start the collection of the kernel debug into an output file:

```
fw ctl kdebug -T -f > /<Path>/<Name of Output File>
```

- To start collecting the kernel debug into cyclic output files:

```
fw ctl kdebug -T -f -o /<Path>/<Name of Output File> -m
<Number of Cyclic Files> [-s <Size of Each Cyclic File in
KB>]
```

Parameters

Note - Only supported parameters are listed.

Table: Parameters of the 'fw ctl debug' command

Parameter	Description
`0 | -x`	Controls how to disable the debug flags: - `0` Resets all debug flags and enables only the default debug flags in all kernel modules. - `-x` Disables all debug flags, including the default flags in all kernel modules. ★ **Best Practice** - Do **not** use the "`-x`" parameter, because it disables even the basic default debug messages.
`-d <Strings to Search>`	When you specify this parameter, the Security Gateway: 1. Examines the applicable debug messages based on the enabled kernel debug modules and their debug flags. 2. Collects only debug messages that contain at least one of the specified strings into the kernel debug buffer. 3. Writes the entire kernel debug buffer into the output file. ⓘ **Notes:** - These strings can be any plain text (not a regular expression) that you see in the debug messages. - Separate the applicable strings by commas without spaces: `-d String1,String2,...,StringN` - You can specify up to 10 strings, up to 250 characters in total.

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description
`-s "<String to Stop Debug>"`	When you specify this parameter, the Security Gateway: 1. Collects the applicable debug messages into the kernel debug buffer based on the enabled kernel debug modules and their debug flags. 2. Does not write any of these debug messages from the kernel debug buffer into the output file. 3. Stops collecting all debug messages when it detects the first debug message that contains the specified string in the kernel debug buffer. 4. Writes the entire kernel debug buffer into the output file. **Notes:** - This one string can be any plain text (not a regular expression) that you see in the debug messages. - String length is up to 50 characters.
`-m <Name of Debug Module>`	Specifies the name of the kernel debug module, for which you print or configure the debug flags.

`{all | + <List of Debug Flags> | - <List of Debug Flags>}`	Specifies which debug flags to enable or disable in the specified kernel debug module:
	- `all`
	Enables all debug flags in the specified kernel debug module.
	- `+ <List of Debug Flags>`
	Enables the specified debug flags in the specified kernel debug module.
	You must press the space bar key after the plus (+) character:
	`+ <Flag1> [<Flag2> ... <FlagN>]`
	Example: `+ drop conn`
	- `- <List of Debug Flags>`
	Disables the specified debug flags in the specified kernel debug module.
	You must press the space bar key after the minus (-) character:
	`- <Flag1> [<Flag2> ... <FlagN>]`
	Example: `- conn`

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description
`-v {"<List of VSIDs>" | all}`	Specifies the list of Virtual Systems. A VSX Gateway automatically filters the collected kernel debug information for debug messages only for these Virtual Systems. ■ `-v "<List of VSIDs>"` Monitors the messages only from the specified Virtual Systems. To specify the Virtual Systems, enter their VSID number separated with commas and without spaces: `"VSID1[,VSID2,VSID3,...,VSIDn]"` Example: `-v "1,3,7"` ■ `-v all` Monitors the messages from all configured Virtual Systems. **Notes:** ■ This parameter is supported only in VSX mode. ■ This parameter and the `-k` parameter are mutually exclusive.

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description

`-e <Expression>`
`-i <Name of Filter File>`
`-i -`
`-u`
	Specifies the INSPECT filter for the debug:
	■ `-e <Expression>` Specifies the INSPECT filter. See *"fw monitor" on page 150*.
	■ `-i <Name of Filter File>` Specifies the file that contains the INSPECT filter.
	■ `-i -` Specifies that the INSPECT filter arrives from the standard input. The Security Gateway prompts to enter the INSPECT filter on the screen.
	■ `-u` - Removes the INSPECT debug filter.
	Notes:
	■ These are *legacy* parameters ("`-e`" and "`-i`").
	■ When you use these parameters ("`-e`" and "`-i`"), the Security Gateway cannot apply the specified INSPECT filter to the accelerated traffic.
	■ For new debug filters, see *"Kernel Debug Filters" on page 427*.

`-z`	The Security Gateway processes some connections in both SecureXL code and in the Host appliance code (for example, Passive Streaming Library (PSL) - an IPS infrastructure, which transparently listens to TCP traffic as network packets, and rebuilds the TCP stream out of these packets.). The Security Gateway processes some connections in only in the Host appliance code. When you use this parameter, kernel debug output contains the debug messages only from the Host appliance code.

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description
`-k`	The Security Gateway processes some connections in both kernel space code and in the user space code (for example, Web Intelligence). The Security Gateway processes some connections only in the kernel space code. When you use this parameter, kernel debug output contains the debug messages only from the kernel space. **Notes:** ■ This parameter is not supported in the VSX mode, in which the Firewall works in the user space. ■ This parameter and the `-v` parameter are mutually exclusive.
`-p` *`<List of Fields>`*	By default, when the Security Gateway prints the debug messages, the messages start with the applicable CPU ID and CoreXL Firewall instance ID. You can print additional fields in the beginning of each debug message. **Notes:** ■ These fields are available: `all`, `proc`, `pid`, `date`, `mid`, `type`, `freq`, `topic`, `time`, `ticks`, `tid`, `text`, `errno`, `host`, `vsid`, `cpu`. ■ When you specify the applicable fields, separate them with commas and without spaces: `Field1,Field2,...,FieldN` ■ The more fields you specify, the higher the load on the CPU and on the hard disk.
`-T`	Prints the time stamp in microseconds in front of each debug message. **Best Practice** - Always use this parameter to make the debug analysis easier.
`-f`	Collects the debug data until you stop the kernel debug in one of these ways: ■ When you press the **CTRL+C** keys. ■ When you run the "`fw ctl debug 0`" command. ■ When you run the "`fw ctl debug -x`" command. ■ When you kill the "`fw ctl kdebug`" process.

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description
`/<Path>/<Name of Output File>`	Specifies the path and the name of the debug output file. ★ **Best Practice** - Always use the largest partition on the disk - `/var/log/`. Security Gateway can generate many debug messages within short time. As a result, the debug output file can grow to large size very fast.
`-o /<Path>/<Name of Output File> -m <Number of Cyclic Files> [-s <Size of Each Cyclic File in KB>]`	Saves the collected debug data into cyclic debug output files. When the size of the current `<Name of Output File>` reaches the specified `<Size of Each Cyclic File in KB>` (more or less), the Security Gateway renames the current `<Name of Output File>` to `<Name of Output File>.0` and creates a new `<Name of Output File>`. If the `<Name of Output File>.0` already exists, the Security Gateway renames the `<Name of Output File>.0` to `<Name of Output File>.1`, and so on - until the specified limit `<Number of Cyclic Files>`. When the Security Gateway reaches the `<Number of Cyclic Files>`, it deletes the oldest files. The valid values are: ▪ `<Number of Cyclic Files>` - from 1 to 999 ▪ `<Size of Each Cyclic File in KB>` - from 1 to 2097150

Kernel Debug Filters

By default, kernel debug output contains information about all processed connections.

You can configure filters for kernel debug to collect debug messages only for the applicable connections.

There are three types of debug filters:

- By connection tuple parameters
- By an IP address parameter
- By a VPN peer parameter

To configure these kernel debug filters, assign the applicable values to the applicable kernel parameters **before** you start the kernel debug.

You assign the values to the applicable kernel parameters temporarily with the "`fw ctl set`" command.

Notes:

- A Security Gateway supports:
  - up to **five** Connection Tuple filters in total (from all types)
  - up to **three** Host IP Address filters
  - up to **two** VPN Peer filters
- A Security Gateway applies these debug filters to both the non-accelerated and accelerated traffic.
- A Security Gateway applies these debug filters to *"Kernel Debug Procedure with Connection Life Cycle" on page 436*.

Best Practice - It is usually simpler to set the Connection Tuple and Host IP Address filters from within the "`fw ctl debug`" command (see the *R81 CLI Reference Guide*). To filter the kernel debug by a VPN Peer, use the procedure below.

To configure debug filter of the type "By connection tuple parameters":

A Security Gateway processes connections based on the 5-tuple:

- Source IP address
- Source Port (see *IANA Service Name and Port Number Registry*)
- Destination IP address
- Destination Port (see *IANA Service Name and Port Number Registry*)
- Protocol Number (see *IANA Protocol Numbers*)

With this debug filter you can filter by these tuple parameters:

Tuple Parameter	Syntax for Kernel Parameters
Source IP address	`fw ctl set str simple_debug_filter_saddr_<N> "<IPv4 or IPv6 Address>"`
Source Ports	`fw ctl set int simple_debug_filter_sport_<N> <1-65535>`
Destination IP address	`fw ctl set str simple_debug_filter_daddr_<N> "<IPv4 or IPv6 Address>"`
Destination Ports	`fw ctl set int simple_debug_filter_dport_<N> <1-65535>`
Protocol Number	`fw ctl set int simple_debug_filter_proto_<N> <0-254>`

Notes:

1. *<N>* is an integer between 1 and 5. This number is an index for the configured kernel parameters of this type.
2. When you specify IP addresses, you must enclose them in double quotes.
3. When you configure kernel parameters with the *same* index <N>, the debug filter is a logical "**AND**" of these kernel parameters.
   In this case, the final filter matches only *one* direction of the processed connection.
   - Example 1 - packets from the source IP address X to the destination IP address Y:

     ```
     simple_debug_filter_saddr_1 <Value X>
     AND
     simple_debug_filter_daddr_1 <Value Y>
     ```

   - Example 2 - packets from the source IP address X to the destination port Y:

     ```
     simple_debug_filter_saddr_1 <Value X>
     AND
     simple_debug_filter_dport_1 <Value Y>
     ```

4. When you configure kernel parameters with the *different* indices <N>, the debug filter is a logical "OR" of these kernel parameters.
   This means that if it is necessary the final filter matches both directions of the connection, then it is necessary to configure the applicable debug filters for both directions.
   - Example 1 - packets either from the source IP address X, or to the destination IP address Y:

     ```
     simple_debug_filter_saddr_1 <Value X>
     OR
     simple_debug_filter_daddr_2 <Value Y>
     ```

   - Example 2 - packets either from the source IP address X, or to the destination port Y:

     ```
     simple_debug_filter_saddr_1 <Value X>
     OR
     simple_debug_filter_dport_2 <Value Y>
     ```

5. For information about the Port Numbers, see *IANA Service Name and Port Number Registry*.
6. For information about the Protocol Numbers, see *IANA Protocol Numbers*.

To configure debug filter of the type "By an IP address parameter":

With this debug filter you can filter by one IP address, which is either the source or the destination IP address of the packet.

Syntax for Kernel Parameters:

```
fw ctl set str simple_debug_filter_addr_<N> "<IPv4 or IPv6
Address>"
```

Notes:

1. *<N>* is an integer between 1 and 3.
   This number is an index for the configured kernel parameters of this type.
2. You can configure one, two, or three of these kernel parameters at the same time.
   - Example 1:
     Configure one IP address (`simple_debug_filter_addr_1`).
   - Example 2:
     Configure two IP addresses (`simple_debug_filter_addr_1` and `simple_debug_filter_addr_2`).
     This would match packets, where any of these IP addresses appears, either as a source or a destination.
3. You must enclose the IP addresses in double quotes.

To configure debug filter of the type "By a VPN peer parameter":

With this debug filter you can filter by one IP address.

Syntax for Kernel Parameters:

```
fw ctl set str simple_debug_filter_vpn_<N> "<IPv4 or IPv6
Address>"
```

Notes:

1. *<N>* is an integer - 1 or 2.
   This number is an index for the configured kernel parameters of this type.
2. You can configure one or two of these kernel parameters at the same time.
   - Example 1:
     Configure one VPN peer (`simple_debug_filter_vpn_1`).
   - Example 2:
     Configure two VPN peers (`simple_debug_filter_vpn_1` and `simple_debug_filter_vpn_2`).
3. You must enclose the IP addresses in double quotes.

To disable all debug filters:

You can disable all the configured debug filters of all types.

Syntax for Kernel Parameter:

```
fw ctl set int simple_debug_filter_off 1
```

Usage Example

It is necessary to show in the kernel debug the information about the connection from Source IP address 192.168.20.30 from any Source Port to Destination IP address 172.16.40.50 to Destination Port 80 (192.168.20.30:<Any> --> 172.16.40.50:80).

Run these commands **before** you start the kernel debug:

```
fw ctl set int simple_debug_filter_off 1
fw ctl set str simple_debug_filter_saddr_1 "192.168.20.30"
fw ctl set str simple_debug_filter_daddr_1 "172.16.40.50"
fw ctl set str simple_debug_filter_saddr_2 "172.16.40.50"
fw ctl set str simple_debug_filter_daddr_2 "192.168.20.30"
fw ctl set int simple_debug_filter_dport_1 80
fw ctl set int simple_debug_filter_sport_2 80
```

Important - In the above example, two Connection Tuple filters are used ("`..._1`" and "`..._2`") - one for each direction, because we want the debug filter to match both directions of this connection.

Kernel Debug Procedure

Alternatively, use the *"Kernel Debug Procedure with Connection Life Cycle" on page 436*.

> **Important:**
>
> - Debug increases the load on the CPU on the Security Gateway / Cluster Members / Security Group Members. Schedule a maintenance window.
> - We strongly recommend to connect over serial console to your Security Gateway / each Cluster Member / Scalable Platform Security Group Members.
>   This is to prevent a possible issue when you cannot work with the CLI because of a high load on the CPU.
> - In Cluster, you must perform these steps on all the Cluster Members in the same way.
> - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.

Step	Instructions
1	Connect to the command line on the Security Gateway / each Cluster Member over SSH, or console. **Note** - On Scalable Platforms (Maestro and Chassis), you must connect to the applicable Security Group.
2	Log in to the Expert mode.

3	Reset the kernel debug options.
	■ On the Security Gateway / each Cluster Member, run:
	`fw ctl debug 0`
	■ On the Scalable Platform Security Group, run:
	`g_fw ctl debug 0`

4	Reset the kernel debug filters.
	■ On the Security Gateway / each Cluster Member, run:
	`fw ctl set int simple_debug_filter_off 1`
	■ On the Scalable Platform Security Group, run:
	`g_fw ctl set int simple_debug_filter_off 1`

Step	Instructions
5	Configure the applicable kernel debug filters. See *"Kernel Debug Filters" on page 427*.

6	Allocate the kernel debug buffer for each CoreXL Firewall instance.
	■ On the Security Gateway / each Cluster Member, run:
	`fw ctl debug -buf 8200`
	■ On the Scalable Platform Security Group, run:
	`g_fw ctl debug -buf 8200`

7	Make sure the kernel debug buffer was allocated.
	■ On the Security Gateway / each Cluster Member, run:
	`fw ctl debug | grep buffer`
	■ On the Scalable Platform Security Group, run:
	`g_fw ctl debug | grep buffer`

8	Enable the applicable debug flags in the applicable kernel modules.
	■ On the Security Gateway / each Cluster Member, run:
	`fw ctl debug -m <module> {all | + <flags>}`
	■ On the Scalable Platform Security Group, run:
	`g_fw ctl debug -m <module> {all | + <flags>}`
	See *"Kernel Debug Modules and Debug Flags" on page 443*.
	Important - The CPU load increases at this point because the Firewall kernel starts to write **some** debug messages to the `/var/log/messages` file and the `dmesg` buffer.

9	Examine the list of the debug flags that are enabled in the specified kernel modules.
	■ On the Security Gateway / each Cluster Member, run:
	`fw ctl debug -m <module>`
	■ On the Scalable Platform Security Group, run:
	`g_fw ctl debug -m <module>`

Step	Instructions

10	Save the kernel debug output to a file.
	■ On the Security Gateway / each Cluster Member, run:
	`fw ctl kdebug -T -f > /var/log/kernel_debug.txt`
	■ On the Scalable Platform Security Group, run:
	`g_fw ctl kdebug -T -f > /var/log/kernel_debug.txt`
	Important - The CPU load increases even more at this point because the Firewall starts to write **all** debug messages to the output file.

11	Replicate the issue, or wait for the issue to occur.
12	Stop the kernel debug output: Press the **CTRL+C** keys. **Important** - This does not stop all CPU load yet because the Firewall kernel continues to write **some** debug messages to the `/var/log/messages` file and the `dmesg` buffer.

13	Reset the kernel debug options.
	■ On the Security Gateway / each Cluster Member, run:
	`fw ctl debug 0`
	■ On the Scalable Platform Security Group, run:
	`g_fw ctl debug 0`
	Important - This stops all CPU load from the kernel debug.

14	Reset the kernel debug filters.
	■ On the Security Gateway / each Cluster Member, run:
	`fw ctl set int simple_debug_filter_off 1`
	■ On the Scalable Platform Security Group, run:
	`g_fw ctl set int simple_debug_filter_off 1`

Step	Instructions
15	Transfer this file from the Security Gateway / each Cluster Member / each Security Group Member to your computer: `/var/log/kernel_debug.txt` **Best Practice** - Compress this file with the "`tar -zxvf`" command and transfer it from the Security Gateway / each Cluster Member / each Security Group Members to your computer. If you transfer to an FTP server, do so in the binary mode.
16	Analyze the debug output file.

Example - Connection 192.168.20.30:<Any> --> 172.16.40.50:80

```
[Expert@GW:0]# fw ctl debug 0
Defaulting all kernel debugging options
Debug state was reset to default.
[Expert@GW:0]#
[Expert@GW:0]# fw ctl set int simple_debug_filter_off 1
[Expert@GW:0]#
[Expert@GW:0]# fw ctl set str simple_debug_filter_saddr_1 "192.168.20.30"
[Expert@GW:0]#
[Expert@GW:0]# fw ctl set str simple_debug_filter_daddr_2 "192.168.20.40"
[Expert@GW:0]#
[Expert@GW:0]# fw ctl set int simple_debug_filter_dport_1 80
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug -buf 8200
Initialized kernel debugging buffer to size 8192K
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug | grep buffer
Kernel debugging buffer size: 8192KB
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug -m fw + conn drop
Updated kernel's debug variable for module fw
Debug flags updated.
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug -m fw
Kernel debugging buffer size: 8192KB
Module: fw
Enabled Kernel debugging options: error warning conn drop
Messaging threshold set to type=Info freq=Common
[Expert@GW:0]#
[Expert@GW:0]# fw ctl kdebug -T -f > /var/log/kernel_debug.txt
... ... Replicate the issue, or wait for the issue to occur ... ...
... ... Press CTRL+C ... ...
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug 0
Defaulting all kernel debugging options
Debug state was reset to default.
[Expert@GW:0]#
[Expert@GW:0]# fw ctl set int simple_debug_filter_off 1
[Expert@GW:0]#
[Expert@GW:0]# ls -l /var/log/kernel_debug.txt
-rw-rw---- 1 admin root 1630619 Apr 12 19:49 /var/log/kernel_debug.txt
[Expert@GW:0]#
```

Kernel Debug Procedure with Connection Life Cycle

Introduction

R80.20 introduced a new debug tool called Connection Life Cycle.

This tool generates a formatted debug output file that presents the debug messages hierarchically by connections and packets:

- The first hierarchy level shows connections.
- After you expand the connection, you see all the packets of this connection.

Important - You must use this tool in the Expert mode together with the regular kernel debug flags (see *"Kernel Debug Modules and Debug Flags" on page 443*).

Syntax

- To start the debug capture:

```
conn_life_cycle.sh -a start -o /<Path>/<Name of Raw Debug
Output File> [{-t | -T}] [[-f "<Filter1>"] [-f "<Filter2>"] [-
f "<Filter3>] [-f "<Filter4>] [-f "<Filter5>"]]
```

- To stop the debug capture and prepare the formatted debug output:

```
conn_life_cycle.sh -a stop -o /<Path>/<Name of Formatted Debug
Output File>
```

Parameters

Table: Parameters of the 'conn_life_cycle.sh' script

Parameter	Description

`-a start`
`-a stop`
	Mandatory.
	Specifies the action:
	▪ `start` - Starts the debug capture based on the debug flags you enabled and debug filters you specified.
	▪ `stop` - Stops the debug capture, resets the kernel debug options, resets the kernel debug filters.

Table: Parameters of the 'conn_life_cycle.sh' script (continued)

Parameter	Description
`-t | -T`	Optional. Specifies the resolution of a time stamp in front of each debug message: ■ `-t` - Prints the time stamp in milliseconds. ■ `-T` - Prints the time stamp in microseconds. **Best Practice** - Always use the "`-T`" option to make the debug analysis easier.

`-f "<Filter>"`	Optional.
	Specifies which connections and packets to capture.
	For additional information, see *"Kernel Debug Filters" on page 427*.
	Important - If you do not specify filters, then the tool prints debug messages for *all* traffic. This causes high load on the CPU and increases the time to format the debug output file.
	Each filter must contain these five numbers (5-tuple) separated with commas:
	`"<Source IP Address>,<Source Port>,<Destination IP Address>,<Destination Port>,<Protocol Number>"`
	Example of capturing traffic from IP 192.168.20.30 from any port to IP 172.16.40.50 to port 22 over the TCP protocol:
	`-f "192.168.20.30,0,172.16.40.50,22,6"`

Table: Parameters of the 'conn_life_cycle.sh' script (continued)

Parameter	Description
	Notes: ■ The tool supports up to **five** of such filters. ■ The tool treats the value 0 (zero) as "any". ■ If you specify two or more filters, the tool performs a logical "OR" of all the filters on each packet. If the packet matches at least one filter, the tool prints the debug messages for this packet. ■ "*`<Source IP Address>`*" and "*`<Destination IP Address>`*" - IPv4 or IPv6 address ■ "*`<Source Port>`*" and "*`<Destination Port>`*" - integers from 1 to 65535 (see *IANA Service Name and Port Number Registry*) ■ *`<Protocol Number>`* - integer from 0 to 254 (see *IANA Protocol Numbers*)
`-o /<Path>/<Name of Raw Debug Output File>`	Mandatory. Specifies the absolute path and the name of the raw debug output file. Example: `-o /var/log/kernel_debug.txt`
`-o /<Path>/<Name of Formatted Debug Output File>`	Mandatory. Specifies the absolute path and the name of the formatted debug output file (to analyze by an administrator). Example: `-o /var/log/kernel_debug_formatted.txt`

Procedure

Important - In cluster, you must perform these steps on all the Cluster Members in the same way.

Step	Instructions
1	Connect to the command line on the Security Gateway.
2	Log in to the Expert mode.
3	Enable the applicable debug flags in the applicable kernel modules: `fw ctl debug -m <module> {all | + <flags>}` See *"Kernel Debug Modules and Debug Flags" on page 443*.
4	Examine the list of the debug flags that are enabled in the specified kernel modules: `fw ctl debug -m <module>`
5	Start the debug capture: `conn_life_cycle.sh -a start -o /var/log/kernel_debug.txt -T -f "<Filter1>" [... [-f "<FilterN>"]]`
6	Replicate the issue, or wait for the issue to occur.
7	Stop the debug capture and prepare the formatted debug output: `conn_life_cycle.sh -a stop -o /var/log/kernel_debug_formatted.txt`
8	Transfer the formatted debug output file from your Security Gateway to your desktop or laptop computer: `/var/log/kernel_debug_formatted.txt`
9	Examine the formatted debug output file in an advanced text editor like Notepad++ (click **Language > R > Ruby**), or any other Ruby language viewer.

Example

Collecting the kernel debug for TCP connection from IP 172.20.168.15 (any port) to IP 192.168.3.53 and port 22

```
[Expert@GW:0]# fw ctl debug -m fw + conn drop
Updated kernel's debug variable for module fw
Debug flags updated.
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug -m fw
Kernel debugging buffer size: 50KB
HOST:
Module: fw
Enabled Kernel debugging options: error warning conn drop
Messaging threshold set to type=Info freq=Common
[Expert@GW:0]#
[Expert@GW:0]# conn_life_cycle.sh -a start -o /var/log/kernel_debug.txt -T -f
"172.20.168.15,0,192.168.3.53,22,6"
Set operation succeeded
Set operation succeeded
Set operation succeeded
Set operation succeeded
Set operation succeeded
Set operation succeeded
Set operation succeeded
Initialized kernel debugging buffer to size 8192K
Set operation succeeded
Capturing started...
[Expert@GW:0]#

... ... Replicate the issue, or wait for the issue to occur ... ...

[Expert@GW:0]#
[Expert@GW:0]# conn_life_cycle.sh -a stop -o /var/log/kernel_debug_formatted.txt
Set operation succeeded
Defaulting all kernel debugging options
Debug state was reset to default.
Set operation succeeded
doing unification...
Openning host debug file /tmp/tmp.KiWmF18217... OK
New unified debug file: /tmp/tmp.imzMZ18220... OK
prepare unification
performing unification
Done :-)
doing grouping...
wrapping connections and packets...
Some of packets lack description, probably because they were already handled when the feature
was enabled.
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug -m fw
Kernel debugging buffer size: 50KB
HOST:
Module: fw
Enabled Kernel debugging options: error warning
Messaging threshold set to type=Info freq=Common
[Expert@GW:0]
[Expert@GW:0] ls -l /var/log/kernel_debug.*
-rw-rw---- 1 admin root 40960 Nov 26 13:02 /var/log/kernel_debug.txt
-rw-rw---- 1 admin root 24406 Nov 26 13:02 /var/log/kernel_debug_formatted.txt
[Expert@GW:0]
```

Opening the kernel debug in Notepad++

Everything is collapsed:

```
   Connection with 1st packet already in handling so no conn details
[+]
{++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++++++++++++++++++++++++++++++++++
```

Opened the first hierarchy level to see the connection:

```
   Connection with 1st packet already in handling so no conn details
[-]
{++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++++++++++++++++++++++++++++++++++
;26Nov2018 13:02:06.736016;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is  INBOUND;
[+]{---------------------------------------------------------- packet begins ----------------
--------------------------------------
```

Opened the second hierarchy level to see the packets of this connection:

```
   Connection with 1st packet already in handling so no conn details
[-]
{++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++++++++++++++++++++++++++++++++++
;26Nov2018 13:02:06.736016;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is  INBOUND;
[-]{---------------------------------------------------------- packet begins ----------------
--------------------------------------
;26Nov2018 13:02:06.736021;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is entering  CHAIN_
MODULES_ENTER;
;26Nov2018 13:02:06.736035;[cpu_2];[fw4_1];#fwconn_lookup_cache: conn <dir 0,
172.20.168.15:57821 -> 192.168.3.53:22 IPP 6>;
;26Nov2018 13:02:06.736046;[cpu_2];[fw4_1];#<1c001,44000,2,1e2,0,UUID: 5bfbc2a2-0000-0000-c0-
a8-3-35-1-0-0-c0, 1,1,ffffffff,ffffffff,40800,0,80,OPQS:
[0,ffffc20033d220f0,0,0,0,0,ffffc20033958648,0,0,0,ffffc200325d57b0,0,0,0,0,0],0,0,0,0,0,0,0,
0,0,0,0,0,0,0>
;26Nov2018 13:02:06.736048;[cpu_2];[fw4_1];CONN LIFE CYCLE: lookup: found;
;26Nov2018 13:02:06.736053;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is entering  VM_ENTER;
;26Nov2018 13:02:06.736055;[cpu_2];[fw4_1];#
;26Nov2018 13:02:06.736060;[cpu_2];[fw4_1];#Before VM: <dir 0, 172.20.168.15:57821 ->
192.168.3.53:22 IPP 6> (len=40) TCP flags=0x10 (ACK), seq=686659054, ack=4181122096, data
end=686659054 (ifn=1) (first seen) (looked up) ;
;26Nov2018 13:02:06.736068;[cpu_2];[fw4_1];#After  VM: <dir 0, 172.20.168.15:57821 ->
192.168.3.53:22 IPP 6> (len=40) TCP flags=0x10 (ACK), seq=686659054, ack=4181122096, data
end=686659054 ;
;26Nov2018 13:02:06.736071;[cpu_2];[fw4_1];#VM Final action=ACCEPT;
;26Nov2018 13:02:06.736072;[cpu_2];[fw4_1];# -----  Stateful VM inbound Completed -----
;26Nov2018 13:02:06.736075;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is exiting  VM_EXIT;
;26Nov2018 13:02:06.736081;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is entering  POST VM_
ENTER;
;26Nov2018 13:02:06.736083;[cpu_2];[fw4_1];#
;26Nov2018 13:02:06.736085;[cpu_2];[fw4_1];#fw_post_vm_chain_handler: (first_seen 32, new_
conn 0, is_my_ip 0, is_first_packet 0);
;26Nov2018 13:02:06.736089;[cpu_2];[fw4_1];#Before POST VM: <dir 0, 172.20.168.15:57821 ->
192.168.3.53:22 IPP 6> (len=40) TCP flags=0x10 (ACK), seq=686659054, ack=4181122096, data
end=686659054 (ifn=1) (first seen) (looked up) ;
;26Nov2018 13:02:06.736095;[cpu_2];[fw4_1];#After  POST VM: <dir 0, 172.20.168.15:57821 ->
192.168.3.53:22 IPP 6> (len=40) TCP flags=0x10 (ACK), seq=686659054, ack=4181122096, data
end=686659054 ;
;26Nov2018 13:02:06.736097;[cpu_2];[fw4_1];#POST VM Final action=ACCEPT;
;26Nov2018 13:02:06.736098;[cpu_2];[fw4_1];# -----  Stateful POST VM inbound Completed -----
;26Nov2018 13:02:06.736101;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is exiting  POST VM_
EXIT;
;26Nov2018 13:02:06.736104;[cpu_2];[fw4_1];#fwconnoxid_msg_get_cliconn: warning - failed to
get connoxid message.;
;26Nov2018 13:02:06.736107;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is entering  CPAS_ENTER;
;26Nov2018 13:02:06.736110;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is exiting  CPAS_EXIT;
;26Nov2018 13:02:06.736113;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is exiting  CHAIN_
MODULES_EXIT;
;26Nov2018 13:02:06.736116;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is  ACCEPTED;
}
;26Nov2018 13:02:06.770652;[cpu_2];[fw4_1];Packet 0xffff8101ea128580 is  INBOUND;
```

Kernel Debug Modules and Debug Flags

This section describes the Kernel Debug Modules and their Debug Flags.

To see the available kernel debug modules and their debug flags, run:

```
fw ctl debug -m
```

List of kernel debug modules (in alphabetical order):

- *"Module 'accel_apps' (Accelerated Applications)" on page 445*
- *"Module 'accel_pm_mgr' (Accelerated Pattern Match Manager)" on page 446*
- *"Module 'APPI' (Application Control Inspection)" on page 447*
- *"Module 'BOA' (Boolean Analyzer for Web Intelligence)" on page 449*
- *"Module 'CI' (Content Inspection)" on page 450*
- *"Module 'cluster' (ClusterXL)" on page 452*
- *"Module 'cmi_loader' (Context Management Interface / Infrastructure Loader)" on page 455*
- *"Module 'CPAS' (Check Point Active Streaming)" on page 457*
- *"Module 'cpcode' (Data Loss Prevention - CPcode)" on page 459*
- *"Module 'CPSSH' (SSH Inspection)" on page 461*
- *"Module 'crypto' (SSL Inspection)" on page 463*
- *"Module 'dlpda' (Data Loss Prevention - Download Agent for Content Awareness)" on page 464*
- *"Module 'dlpk' (Data Loss Prevention - Kernel Space)" on page 466*
- *"Module 'dlpuk' (Data Loss Prevention - User Space)" on page 467*
- *"Module 'DOMO' (Domain Objects)" on page 469*
- *"Module 'fg' (FloodGate-1 - QoS)" on page 470*
- *"Module 'FILE_SECURITY' (File Inspection)" on page 472*
- *"Module 'FILEAPP' (File Application)" on page 473*
- *"Module 'fw' (Firewall)" on page 474*
- *"Module 'gtp' (GPRS Tunneling Protocol)" on page 481*
- *"Module 'h323' (VoIP H.323)" on page 483*
- *"Module 'ICAP_CLIENT' (Internet Content Adaptation Protocol Client)" on page 484*

- *"Module 'IDAPI' (Identity Awareness API)" on page 486*
- *"Module 'kiss' (Kernel Infrastructure)" on page 488*
- *"Module 'kissflow' (Kernel Infrastructure Flow)" on page 491*
- *"Module 'MALWARE' (Threat Prevention)" on page 492*
- *"Module 'multik' (Multi-Kernel Inspection - CoreXL)" on page 493*
- *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 495*
- *"Module 'NRB' (Next Rule Base)" on page 497*
- *"Module 'PSL' (Passive Streaming Library)" on page 499*
- *"Module 'RAD_KERNEL' (Resource Advisor - Kernel Space)" on page 500*
- *"Module 'RTM' (Real Time Monitoring)" on page 501*
- *"Module 'seqvalid' (TCP Sequence Validator and Translator)" on page 503*
- *"Module 'SFT' (Stream File Type)" on page 504*
- *"Module 'SGEN' (Struct Generator)" on page 505*
- *"Module 'synatk' (Accelerated SYN Defender)" on page 506*
- *"Module 'TPUTILS' (Threat Prevention Utilities)" on page 507*
- *"Module 'UC' (UserCheck)" on page 508*
- *"Module 'UP' (Unified Policy)" on page 509*
- *"Module 'upconv' (Unified Policy Conversion)" on page 511*
- *"Module 'UPIS' (Unified Policy Infrastructure)" on page 512*
- *"Module 'VPN' (Site-to-Site VPN and Remote Access VPN)" on page 514*
- *"Module 'WS' (Web Intelligence)" on page 517*
- *"Module 'WS_SIP' (Web Intelligence VoIP SIP Parser)" on page 520*
- *"Module 'WSIS' (Web Intelligence Infrastructure)" on page 522*

Module 'accel_apps' (Accelerated Applications)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m accel_apps + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m accel_apps + {all | <List of Debug Flags>}
```

Flag	Description
`av_lite`	Content Inspection (Anti-Virus) Lite application - general information about packet processing
`cmi_lite`	Context Management Interface / Infrastructure Lite application - general information about packet processing
`daf_lite`	Decrypt & Forward Lite application - general information about packet processing
`daf_lite_dump`	Decrypt & Forward Lite application - writes the contents of the internal buffer
`error`	General errors
`info`	General information
`rad_lite`	Resource Advisor Lite application - general information about internal connection processing
`warning`	General warnings

Module 'accel_pm_mgr' (Accelerated Pattern Match Manager)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m accel_pm_mgr + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m accel_pm_mgr + {all | <List of Debug Flags>}
```

Flag	Description
`debug`	Operations in the Accelerated Pattern Match Manager module
`error`	General errors and failures
`flow`	Internal flow of functions
`submit_error`	General failures to submit the data for analysis
`warning`	General warnings and failures

Module 'APPI' (Application Control Inspection)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m APPI + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m APPI + {all | <List of Debug Flags>}
```

Flag	Description
`account`	Accounting information
`address`	Information about connection's IP address
`btime`	Browse time
`connection`	Application Control connections
`coverage`	Coverage times (entering, blocking, and time spent)
`error`	General errors
`global`	Global policy operations
`info`	General information
`limit`	Application Control limits
`memory`	Memory allocation operations
`module`	Operations in the Application Control module (initialization, module loading, calls to the module, policy loading, and so on)
`observer`	Classification Object (CLOB) observer (data classification)
`policy`	Application Control policy
`referrer`	Application Control referrer
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')

Flag	Description
`urlf_ssl`	Application Control and URL Filtering for SSL
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'BOA' (Boolean Analyzer for Web Intelligence)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m BOA + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m BOA + {all | <List of Debug Flags>}
```

Flag	Description
`analyzer`	Operations in the BOA module
`disasm`	Disassembler information
`error`	General errors
`fatal`	Fatal errors
`flow`	Operations in the BOA module
`info`	General information
`lock`	Information about internal locks in the FireWall kernel
`memory`	Memory allocation operations
`spider`	Internal hash tables
`stat`	Statistics
`stream`	Memory allocation when processing streamed data
`warning`	General warnings

Module 'CI' (Content Inspection)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m CI + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m CI + {all | <List of Debug Flags>}
```

Flag	Description
`address`	Prints connection addresses (as `Source_IP:Source_Port -> Dest_IP:Dest_Port`)
`av`	Anti-Virus inspection
`coverage`	Coverage times (entering, blocking, and time spent)
`crypto`	Basic information about encryption and decryption
`error`	General errors
`fatal`	Fatal errors
`filter`	Basic information about URL filters
`info`	General information
`ioctl`	*Currently is not used*
`memory`	Memory allocation operations
`module`	Operations in the Content Inspection module (initialization, module loading, calls to the module, policy loading, and so on)
`policy`	Content Inspection policy
`profile`	Basic information about the Content Inspection module (initialization, destroying, freeing)
`regexp`	Regular Expression library
`session`	Session layer
`stat`	Content Inspection statistics

Flag	Description
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`track`	Use only for very limited important debug prints, so it can be used in a loaded environment - Content-Disposition, Content-Type, extension validation, extension matching
`uf`	URL filters and URL cache
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'cluster' (ClusterXL)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m cluster + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m cluster + {all | <List of Debug Flags>}
```

Notes:

- To print all synchronization operations in Check Point cluster in the debug output, enable these debug flags:
  - The debug flag "`sync`" in *"Module 'fw' (Firewall)" on page 474*
  - The debug flag "`sync`" in *"Module 'CPAS' (Check Point Active Streaming)" on page 457*
- To print the contents of the packets in HEX format in the debug output (as "`FW-1: fwha_print_packet: Buffer ...`"), before you start the kernel debug, set this kernel parameter on each Cluster Member / the applicable Scalable Platform Security Group:
  - On the Security Gateway / each Cluster Member, run in the Expert mode:

    ```
    fw ctl set int fwha_dprint_io 1
    ```

  - On the Scalable Platform Security Group, run in the Expert mode:

    ```
    g_fw ctl set int fwha_dprint_io 1
    ```

- To print all network checks in the debug output, before you start the kernel debug, set this kernel parameter on each Cluster Member:
  - On the Security Gateway / each Cluster Member, run in the Expert mode:

    ```
    fw ctl set int fwha_dprint_all_net_check 1
    ```

  - On the Scalable Platform Security Group, run in the Expert mode:

    ```
    g_fw ctl set int fwha_dprint_all_net_check 1
    ```

Flag	Description
`arp`	ARP Forwarding (see sk111956)
`autoccp`	Operations of CCP in Auto mode
`ccp`	Reception and transmission of Cluster Control Protocol (CCP) packets
`cloud`	Replies to the probe packets in CloudGuard IaaS

Flag	Description
`conf`	Cluster configuration and policy installation
`correction`	Correction Layer
`cu`	Connectivity Upgrade (see sk107042)
`drop`	Connections dropped by the cluster Decision Function (DF) module (does not include CCP packets)
`forward`	Forwarding Layer messages (when Cluster Members send and receive a forwarded packet)
`if`	Interface tracking and validation (all the operations and checks on interfaces)
`ifstate`	Interface state (all the operations and checks on interfaces)
`io`	Information about sending of packets through cluster interfaces
`log`	Creating and sending of logs by cluster **Note** - In addition, enable the debug flag "`log`" in *"Module 'fw' (Firewall)" on page 474*.
`mac`	Current configuration of and detection of cluster interfaces **Note** - In addition, enable the debug flags "`conf`" and "`if`" in this debug module
`mmagic`	Operations on "MAC magic" (getting, setting, updating, initializing, dropping, and so on)
`msg`	Handling of internal messages between Cluster Members
`multik`	Processing of connections in CoreXL Firewall instances **Notes:** ■ In addition, see *"Module 'multik' (Multi-Kernel Inspection - CoreXL)" on page 493*. ■ If you use the QoS Software Blade, enable the debug flag "`multik`" in the *"Module 'fg' (FloodGate-1 - QoS)" on page 470*.
`osp`	Only for Scalable Platforms: Distribution of connections between Security Group Members
`pivot`	Operation of ClusterXL in Load Sharing Unicast mode (Pivot mode)

Flag	Description
`pnote`	Registration and monitoring of Critical Devices (pnotes)
`select`	Packet selection (includes the Decision Function)
`smo`	Only for Scalable Platforms: Processing of connections on the SMO Security Group Member
`stat`	States of cluster members (state machine)
`subs`	Subscriber module (set of APIs, which enable user space processes to be aware of the current state of the ClusterXL state machine and other clustering configuration parameters)
`timer`	Reports of cluster internal timers
`trap`	Sending trap messages from the cluster kernel to the RouteD daemon about Master change
`unisync`	Only for Scalable Platforms: Unicast Sync - synchronization of connections to backup Security Group Members on the local Maestro Site / Scalable Chassis and to one Security Group Member one the standby Maestro Site / Scalable Chassis

Module 'cmi_loader' (Context Management Interface / Infrastructure Loader)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m cmi_loader + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m cmi_loader + {all | <List of Debug Flags>}
```

Flag	Description
`address`	Information about connection's IP address
`connection`	Internal messages about connection
`coverage`	Coverage times (entering, blocking, and time spent)
`cpcode`	DLP CPcode ⓘ **Note** - Also see *"Module 'cpcode' (Data Loss Prevention - CPcode)" on page 459*.
`error`	General errors
`global_states`	User Space global state structures
`info`	General information
`inspect`	INSPECT code
`memory`	Memory allocation operations
`module`	Operations in the Context Management Interface / Infrastructure Loader module (initialization, module loading, calls to the module, contexts, and so on)
`parsers_is`	Module parsers infrastructure
`policy`	Policy installation
`sigload`	Signatures, patterns, ranges
`subject`	Prints the debug subject of each debug message

Flag	Description
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'CPAS' (Check Point Active Streaming)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m CPAS + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m CPAS + {all | <List of Debug Flags>}
```

Flag	Description
`api`	Interface layer messages
`conns`	Detailed description of connections, and connection's limit-related messages
`cpconntim`	Information about internal timers
`error`	General errors
`events`	Event-related messages
`ftp`	Messages of the FTP example server
`glue`	Glue layer messages
`http`	Messages of the HTTP example server
`icmp`	Messages of the ICMP example server
`notify`	E-mail Messaging Security application
`pkts`	Packets handling messages (allocation, splitting, resizing, and so on)
`skinny`	Processing of Skinny Client Control Protocol (SCCP) connections
`sync`	Synchronization operations in cluster **Note** - Also see the debug flag "`sync`" in *"Module 'fw' (Firewall)" on page 474*.
`tcp`	TCP processing messages
`tcpinfo`	TCP processing messages - more detailed description

Flag	Description
`timer`	Reports of internal timer ticks ❗ **Warning** - Prints many messages, without real content.
`warning`	General warnings

Module 'cpcode' (Data Loss Prevention - CPcode)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m cpcode + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m cpcode + {all | <List of Debug Flags>}
```

Note - Also see:

- *"Module 'dlpda' (Data Loss Prevention - Download Agent for Content Awareness)" on page 464*
- *"Module 'dlpk' (Data Loss Prevention - Kernel Space)" on page 466*
- *"Module 'dlpuk' (Data Loss Prevention - User Space)" on page 467*

Flag	Description
`cplog`	Resolving of names and IP addresses for Check Point logs
`csv`	Creation of CSV files
`echo`	Prints the function that called the CPcode module
`error`	General errors
`init`	Initializing of CPcode system
`io`	Input / Output functionality for CPcode module
`ioctl`	IOCTL control messages to kernel
`kisspm`	Kernel Infrastructure Pattern Matcher
`memory`	Memory allocation operations
`persist`	Operations on persistence domains
`policy`	Policy operations
`run`	Policy operations
`url`	Operations on URLs
`vm`	Virtual Machine execution

Flag	Description
`warning`	General warnings

Module 'CPSSH' (SSH Inspection)

R80.40 introduced SSH Deep Packet Inspection - decryption / encryption of SSH, extraction of files from SFTP/SCP, blocking of SSH port forwarding, and so on.

For more information, see the *R81 Threat Prevention Administration Guide*.

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m CPSSH + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m CPSSH + {all | <List of Debug Flags>}
```

Important - Also enable the debug flag "`cpsshi`" in *"Module 'fw' (Firewall)" on page 474*.

Flag	Description
`authentication`	Detailed information about authentication
`binary_packet`	Detailed information about packets
`conn_proto`	Detailed information about connections
`crypto`	Encryption and decryption **Note** - Also see *"Module 'crypto' (SSL Inspection)" on page 463*.
`dump`	Dumps the connection buffer
`error`	General errors
`info`	General information
`mux_auth_app`	Information about authentication **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 495*.
`mux_conn_app`	Information about connections **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 495*.

Flag	Description
`mux_decrypt_app`	Information about decryption of connections **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 495*.
`mux_encrypt_app`	Information about encryption of connections **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 495*.
`mux_inf`	Internal flow **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 495*.
`mux_stream`	Internal flow **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 495*.
`probe`	Information about connections
`session`	Internal flow
`sftp_parser`	Parser of SFTP / SCP connections
`state_machine`	Information about the module State Machine
`trans_proto`	Information about client and server communication
`warning`	General warnings

Module 'crypto' (SSL Inspection)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m crypto + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m crypto + {all | <List of Debug Flags>}
```

Flag	Description
`error`	General errors
`info`	General information
`warning`	General warnings

Module 'dlpda' (Data Loss Prevention - Download Agent for Content Awareness)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m dlpda + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m dlpda + {all | <List of Debug Flags>}
```

Note - Also see:

- *"Module 'cpcode' (Data Loss Prevention - CPcode)" on page 459*
- *"Module 'dlpk' (Data Loss Prevention - Kernel Space)" on page 466*
- *"Module 'dlpuk' (Data Loss Prevention - User Space)" on page 467*

Flag	Description
`address`	Information about connection's IP address
`cmi`	Context Management Interface / Infrastructure operations
`coverage`	Coverage times (entering, blocking, and time spent)
`ctx`	Operations on DLP context
`engine`	Content Awareness engine module
`error`	General errors
`filecache`	Content Awareness file caching
`info`	General information
`memory`	Memory allocation operations
`mngr`	*Currently is not used*
`module`	Initiation / removal of the Content Awareness infrastructure
`observer`	Classification Object (CLOB) observer (data classification)

Flag	Description
`policy`	Content Awareness policy
`slowpath`	*Currently is not used*
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'dlpk' (Data Loss Prevention - Kernel Space)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

  ```
  fw ctl debug -m dlpk + {all | <List of Debug Flags>}
  ```

- On the Scalable Platform Security Group, run in the Expert mode:

  ```
  g_fw ctl debug -m dlpk + {all | <List of Debug Flags>}
  ```

Note - Also see:

- *"Module 'cpcode' (Data Loss Prevention - CPcode)" on page 459*
- *"Module 'dlpda' (Data Loss Prevention - Download Agent for Content Awareness)" on page 464*
- *"Module 'dlpuk' (Data Loss Prevention - User Space)" on page 467*

Flag	Description
`cmi`	HTTP Proxy, connection redirection, identity information, Async
`drv`	DLP inspection
`error`	General errors
`identity`	User identity, connection identity, Async
`rulebase`	DLP rulebase match
`stat`	Counter statistics

Module 'dlpuk' (Data Loss Prevention - User Space)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m dlpuk + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m dlpuk + {all | <List of Debug Flags>}
```

Note - Also see:

- *"Module 'cpcode' (Data Loss Prevention - CPcode)" on page 459*
- *"Module 'dlpda' (Data Loss Prevention - Download Agent for Content Awareness)" on page 464*
- *"Module 'dlpk' (Data Loss Prevention - Kernel Space)" on page 466*

Flag	Description
`address`	Information about connection's IP address
`buffer`	*Currently is not used*
`coverage`	Coverage times (entering, blocking, and time spent)
`error`	General errors
`info`	General information
`memory`	Memory allocation operations
`module`	Initiation / removal of the Data Loss Prevention User Space modules' infrastructure
`policy`	*Currently is not used*
`serialize`	Data buffers and data sizes
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System

Flag	Description
`warning`	General warnings

Module 'DOMO' (Domain Objects)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

  ```
  fw ctl debug -m DOMO + {all | <List of Debug Flags>}
  ```

- On the Scalable Platform Security Group, run in the Expert mode:

  ```
  g_fw ctl debug -m DOMO + {all | <List of Debug Flags>}
  ```

Flag	Description
`conn`	Internal processing of connections
`module`	Operations in the Domain Objects module (initialization, module loading, calls to the module, policy loading, and so on)
`policy`	*Currently is not used*

Module 'fg' (FloodGate-1 - QoS)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m fg + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m fg + {all | <List of Debug Flags>}
```

Flag	Description
`chain`	Tracing each packet through FloodGate-1 stages in the cookie chain
`chainq`	Internal Chain Queue mechanism - holding and releasing of packets during critical actions (policy installation and uninstall)
`classify`	Classification of connections to QoS rules
`conn`	Processing and identification of connection
`dns`	DNS classification mechanism
`drops`	Dropped packets due to WFRED policy
`dropsv`	Dropped packets due to WFRED policy - with additional debug information (verbose)
`error`	General errors
`flow`	Internal flow of connections (direction, interfaces, buffers, and so on)
`fwrate`	Rate statistics for each interface and direction
`general`	*Currently is not used*
`install`	Policy installation
`llq`	Low latency queuing
`log`	Everything related to calls in the log
`ls`	Processing of connections in ClusterXL in Load Sharing Mode
`memory`	Memory allocation operations

Flag	Description
`multik`	Processing of connections in CoreXL Firewall instances **Notes:** - In addition, see *"Module 'multik' (Multi-Kernel Inspection - CoreXL)" on page 493*. - In a cluster, enable the debug flag "`multik`" in the *"Module 'cluster' (ClusterXL)" on page 452*. - If you use the IPsec VPN Software Blade, enable the debug flag "`multik`" in the *"Module 'VPN' (Site-to-Site VPN and Remote Access VPN)" on page 514*.
`pkt`	Packet recording mechanism
`policy`	QoS policy rules matching
`qosaccel`	Acceleration of QoS traffic
`rates`	Rule and connection rates (IQ Engine behavior and status)
`rtm`	Failures in information gathering in the Real Time Monitoring module **Note** - In addition, see *"Module 'RTM' (Real Time Monitoring)" on page 501*.
`sched`	Basic scheduling information
`tcp`	TCP streaming (re-transmission detection) mechanism
`time`	*Currently is not used*
`timers`	Reports of internal timer ticks **Warning** - Prints many messages, without real content.
`url`	URL and URI for QoS classification
`verbose`	Prints additional information (used with other debug flags)

Module 'FILE_SECURITY' (File Inspection)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

  ```
  fw ctl debug -m FILE_SECURITY + {all | <List of Debug Flags>}
  ```

- On the Scalable Platform Security Group, run in the Expert mode:

  ```
  g_fw ctl debug -m FILE_SECURITY + {all | <List of Debug
  Flags>}
  ```

Note - Also see *"Module 'WSIS' (Web Intelligence Infrastructure)" on page 522*.

Flag	Description
`cache`	File cache
`global`	Global operations
`memory`	*Currently is not used*
`module`	Operations in the FILE_SECURITY module (identification and processing of connections)

Module 'FILEAPP' (File Application)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m FILEAPP + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m FILEAPP + {all | <List of Debug Flags>}
```

Flag	Description
`address`	Information about connection's IP address
`coverage`	Coverage times (entering, blocking, and time spent)
`error`	General errors
`filetype`	Information about processing a file type
`global`	Allocation and creation of global object
`info`	General information
`memory`	Memory allocation operations
`module`	Operations in the FILEAPP module (initialization, module loading, calls to the module, and so on)
`normalize`	File normalization operations (internal operations)
`parser`	File parsing
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`upload`	File upload operations
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'fw' (Firewall)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m fw + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m fw + {all | <List of Debug Flags>}
```

Flag	Description
`acct`	Accounting data in logs for Application Control (in addition, enable the debug of *"Module 'APPI' (Application Control Inspection)" on page 447*)
`advp`	Advanced Patterns (signatures over port ranges) - runs under ASPII and CMI
`aspii`	Accelerated Stateful Protocol Inspection Infrastructure (INPSECT streaming)
`balance`	ConnectControl - logical servers in kernel, load balancing
`bridge`	Bridge mode
`bypass_timer`	Universal Bypass on CoreXL Firewall Instances during load
`caf`	Mirror and Decrypt feature - only mirror operations on all traffic
`cgnat`	Carrier Grade NAT (CGN/CGNAT)
`chain`	Connection Chain modules, cookie chain
`chainfwd`	Chain forwarding - related to cluster kernel parameter `fwha_perform_chain_forwarding`
`cifs`	Processing of Microsoft Common Internet File System (CIFS) protocol
`citrix`	Processing of Citrix connections
`cmi`	Context Management Interface / Infrastructure - IPS signature manager
`conn`	Processing of all connections

Flag	Description
`connstats`	Connections statistics for Evaluation of Heavy Connections in CPView (see sk105762)
`content`	Anti-Virus content inspection
`context`	Operations on Memory context and CPU context in *"Module 'kiss' (Kernel Infrastructure)" on page 488*
`cookie`	Virtual de-fragmentation , cookie issues (cookies in the data structure that holds the packets)
`corr`	Correction layer
`cpsshi`	SSH Inspection ⓘ **Important** - In addition, enable all the debug flags in *"Module 'CPSSH' (SSH Inspection)" on page 461*.
`cptls`	CRYPTO-PRO Transport Layer Security (HTTPS Inspection) - Russian VPN GOST
`crypt`	Encryption and decryption of packets (algorithms and keys are printed in clear text and cipher text)
`cvpnd`	Processing of connections handled by the Mobile Access daemon
`dfilter`	Operations in the debug filters (see *"Kernel Debug Filters" on page 427*)
`dlp`	Processing of Data Loss Prevention connections
`dnstun`	DNS tunnels
`domain`	DNS queries
`dos`	DDoS attack mitigation (part of IPS)
`driver`	Check Point kernel attachment (access to kernel is shown as log entries)
`drop`	Reason for (almost) every dropped packet
`drop_tmpl`	Operations in Drop Templates
`dynlog`	Dynamic log enhancement (INSPECT logs)
`epq`	End Point Quarantine (and AMD)
`error`	General errors

Flag	Description
`event`	Event App features (DNS, HTTP, SMTP, FTP)
`ex`	Expiration issues (time-outs) in dynamic kernel tables
`fast_accel`	Fast acceleration of connections
`filter`	Packet filtering performed by the Check Point kernel and all data loaded into kernel
`ftp`	Processing of FTP Data connections (used to call applications over FTP Data - i.e., Anti-Virus)
`handlers`	Operations related to the Context Management Interface / Infrastructure Loader ℹ **Note** - In addition, see *"Module 'cmi_loader' (Context Management Interface / Infrastructure Loader)" on page 455*.
`highavail`	Cluster configuration - changes in the configuration and information about interfaces during traffic processing
`hold`	Holding mechanism and all packets being held / released
`icmptun`	ICMP tunnels
`if`	interface-related information (accessing the interfaces, installing a filter on an interfaces)
`install`	Driver installation - NIC attachment (actions performed by the "`fw ctl install`" and "`fw ctl uninstall`" commands)
`integrity`	Integrity Client (enforcement cooperation)
`ioctl`	IOCTL control messages (communication between kernel and daemons, loading and unloading of the FireWall)
`ipopt`	Enforcement of IP Options
`ips`	IPS logs and IPS IOCTL
`ipv6`	Processing of IPv6 traffic
`kbuf`	Kernel-buffer memory pool (for example, encryption keys use these memory allocations)

Flag	Description
`ld`	Kernel dynamic tables infrastructure (reads from / writes to the tables) **Warning** - Security Gateway can freeze or hang due to very high CPU load!.
`leaks`	Memory leak detection mechanism
`link`	Creation of links in Connections kernel table (ID 8158)
`log`	Everything related to calls in the log
`machine`	INSPECT Virtual Machine (actual assembler commands being processed) **Warning** - Security Gateway can freeze or hang due to very high CPU load!.
`mail`	Issues with e-mails over POP3, IMAP
`malware`	Matching of connections to Threat Prevention Layers (multiple rulebases) **Note** - In addition, see *"Module 'MALWARE' (Threat Prevention)" on page 492*.
`media`	*Does not apply anymore* Only on Security Gateway that runs on Windows OS: Transport Driver Interface information (interface-related information)
`memory`	Memory allocation operations
`mgcp`	Media Gateway Control Protocol (complementary to H.323 and SIP)
`misc`	Miscellaneous helpful information (not shown with other debug flags)
`misp`	ISP Redundancy
`monitor`	Prints output similar to the "`fw monitor`" command (see *"fw monitor" on page 150*) **Note** - In addition, enable the debug flag "`misc`" in this module.
`monitorall`	Prints output similar to the "`fw monitor -p all`" command (see *"fw monitor" on page 150*) **Note** - In addition, enable the debug flag "`misc`" in this module.

Flag	Description
`mrtsync`	Synchronization between cluster members of Multicast Routes that are added when working with Dynamic Routing Multicast protocols
`msnms`	MSN over MSMS (MSN Messenger protocol) In addition, always enable the debug flag '`sip`' in this module
`multik`	Processing of connections in CoreXL Firewall instances **Notes:** - This debug flag enables all the debug flags in the *"Module 'multik' (Multi-Kernel Inspection - CoreXL)" on page 493*, except for the debug flag "`packet`". - In a cluster, enable the debug flag "`multik`" in the *"Module 'cluster' (ClusterXL)" on page 452*. - If you use the IPsec VPN Software Blade, enable the debug flag "`multik`" in the *"Module 'VPN' (Site-to-Site VPN and Remote Access VPN)" on page 514*. - If you use the QoS Software Blade, enable the debug flag "`multik`" in the *"Module 'fg' (FloodGate-1 - QoS)" on page 470*.
`nac`	Network Access Control (NAC) feature in Identity Awareness
`nat`	NAT issues - basic information
`nat_ hitcount`	Hit Count in NAT Rule Base
`nat_sync`	NAT issues - NAT port allocation operations in Check Point cluster
`nat64`	NAT issues - 6in4 tunnels (IPv6 over IPv4) and 4in6 tunnels (IPv4 over IPv6)
`netquota`	IPS protection "Network Quota"
`ntup`	Non-TCP / Non-UDP traffic policy (traffic parser)
`packet`	Actions performed on packets (like Accept, Drop, Fragment)
`packval`	Stateless verifications (sequences, fragments, translations and other header verifications)
`portscan`	Prevention of port scanning
`prof`	Connection profiler for Firewall Priority Queues (see sk105762)

Flag	Description
`q`	Driver queue (for example, cluster synchronization operations) This debug flag is crucial for the debug of Check Point cluster synchronization issues
`qos`	QoS (FloodGate-1)
`rad`	Resource Advisor policy (for Application Control, URL Filtering, and others)
`route`	Routing issues This debug flag is crucial for the debug of ISP Redundancy issues
`sam`	Suspicious Activity Monitoring
`sctp`	Processing of Stream Control Transmission Protocol (SCTP) connections
`scv`	SecureClient Verification
`shmem`	*Currently is not used*
`sip`	VoIP traffic - SIP and H.323 **Note** - In addition, see: ■ *"Module 'h323' (VoIP H.323)" on page 483* ■ *"Module 'WS_SIP' (Web Intelligence VoIP SIP Parser)" on page 520*
`smtp`	Issues with e-mails over SMTP
`sock`	Sockstress TCP DoS attack (CVE-2008-4609)
`span`	Monitor mode (mirror / span port)
`spii`	Stateful Protocol Inspection Infrastructure and INSPECT Streaming Infrastructure
`synatk`	IPS protection 'SYN Attack' (SYNDefender) **Note** - In addition, see *"Module 'synatk' (Accelerated SYN Defender)" on page 506*.
`sync`	Synchronization operations in Check Point cluster **Note** - In addition, see the debug flag "`sync`" in *"Module 'CPAS' (Check Point Active Streaming)" on page 457*.
`tcpstr`	TCP streaming mechanism

Flag	Description
`te`	Prints the name of an interface for incoming connection from Threat Emulation Machine
`tlsparser`	*Currently is not used*
`tp_container`	Operations in the Threat Prevention container
`ua`	Processing of Universal Alcatel "UA" connections
`ucd`	Processing of UserCheck connections in Check Point cluster
`unibypass`	Universal Bypass on CoreXL Firewall Instances during load
`user`	User Space communication with Kernel Space (most useful for configuration and VSX debug)
`utest`	*Currently is not used*
`vm`	Virtual Machine chain decisions on traffic going through the `fw_filter_chain`
`wap`	Processing of Wireless Application Protocol (WAP) connections
`warning`	General warnings
`wire`	Wire-mode Virtual Machine chain module
`xlate`	NAT issues - basic information
`xltrc`	NAT issues - additional information - going through NAT rulebase
`zeco`	Memory allocations in the Zero-Copy kernel module

Module 'gtp' (GPRS Tunneling Protocol)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m gtp + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m gtp + {all | <List of Debug Flags>}
```

Flag	Description
`create`	GTPv0 / GTPv1 create PDP context
`create2`	GTPv2 create session
`dbg`	GTP debug mechanism
`delete`	GTPv0 / GTPv1 delete PDP context
`delete2`	GTPv2 delete session
`error`	General GTP errors
`ioctl`	GTP IOCTL commands
`ld`	Operations with GTP kernel tables (addition, removal, modification of entries)
`log`	GTPv0 / GTPv1 logging
`log2`	GTPv2 logging
`modify`	GTPv2 modify bearer
`other`	GTPv0 / GTPv1 other messages
`other2`	GTPv2 other messages
`packet`	GTP main packet flow
`parse`	GTPv0 / GTPv1 parsing
`parse2`	GTPv2 parsing
`policy`	Policy installation

Flag	Description
`state`	GTPv0 / GTPv1 dispatching
`state2`	GTPv2 dispatching
`sxl`	Processing of GTP connections in SecureXL
`tpdu`	GTP T-PDU
`update`	GTPv0 / GTPv1 update PDP context

Module 'h323' (VoIP H.323)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m h323 + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m h323 + {all | <List of Debug Flags>}
```

Flag	Description
`align`	General VoIP debug messages (for example, VoIP infrastructure)
`cpas`	Debug messages about the CPAS TCP **Important** - This debug flag is **not** included when you use the syntax "`fw ctl debug -m h323 all`"
`decode`	H.323 decoder messages
`error`	General errors
`h225`	H225 call signaling messages (`SETUP`, `CONNECT`, `RELEASE COMPLETE`, and so on)
`h245`	H245 control signaling messages (`OPEN LOGICAL CHANNEL`, `END SESSION COMMAND`, and so on)
`init`	Internal errors
`ras`	H225 RAS messages (`REGISTRATION`, `ADMISSION`, and `STATUS REQUEST / RESPONSE`)

Module 'ICAP_CLIENT' (Internet Content Adaptation Protocol Client)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m ICAP_CLIENT + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m ICAP_CLIENT + {all | <List of Debug Flags>}
```

Flag	Description
`address`	Information about connection's IP address
`blade`	Internal operations in the ICAP Client module
`coverage`	Coverage times (entering, blocking, and time spent)
`cpas`	Check Point Active Streaming (CPAS) **Note** - Also see *"Module 'CPAS' (Check Point Active Streaming)" on page 457*.
`daf_cmi`	Mirror and Decrypt of HTTPS traffic - operations related to the Context Management Interface / Infrastructure Loader **Note** - Also see *"Module 'cmi_loader' (Context Management Interface / Infrastructure Loader)" on page 455*.
`daf_module`	Mirror and Decrypt of HTTPS traffic - operations related to the ICAP Client module
`daf_policy`	Mirror and Decrypt of HTTPS traffic - operations related to policy installation
`daf_rulebase`	Mirror and Decrypt of HTTPS traffic - operations related to rulebase
`daf_tcp`	Mirror and Decrypt of HTTPS traffic - internal processing of TCP connections
`error`	General errors
`global`	Global operations in the ICAP Client module

Flag	Description
`icap`	Processing of ICAP connections
`info`	General information
`memory`	Memory allocation operations
`module`	Operations in the ICAP Client module (initialization, module loading, calls to the module, and so on)
`policy`	Policy installation
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`trick`	Data Trickling mode
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'IDAPI' (Identity Awareness API)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m IDAPI + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m IDAPI + {all | <List of Debug Flags>}
```

>

Flag	Description
`address`	Information about connection's IP address
`async`	Checking for known networks
`classifier`	Data classification
`clob`	Classification Object (CLOB) observer (data classification)
`coverage`	Coverage times (entering, blocking, and time spent)
`data`	Portal, IP address matching for Terminal Servers Identity Agent, session handling
`error`	General errors
`htab`	Checking for network IP address, working with kernel tables
`info`	General information
`log`	Various logs for internal operations
`memory`	Memory allocation operations
`module`	Removal of the Identity Awareness API debug module's infrastructure, failure to convert to Base64, failure to append Source to Destination, and so on
`observer`	Data classification observer
`subject`	Prints the debug subject of each debug message
`test`	IP test, Identity Awareness API synchronization

Flag	Description
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'kiss' (Kernel Infrastructure)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m kiss + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m kiss + {all | <List of Debug Flags>}
```

Note - In addition, see *"Module 'kissflow' (Kernel Infrastructure Flow)" on page 491*.

Flag	Description
accel_pm	Accelerated Pattern Matcher
bench	CPU benchmark
connstats	Statistics for connections
cookie	Virtual de-fragmentation , cookie issues (cookies in the data structure that holds the packets)
dbg_filter	Information about the configured Debug Filters - *"Kernel Debug Filters" on page 427*
dfa	Pattern Matcher (Deterministic Finite Automaton) compilation and execution
driver	Loading / unloading of the FireWall driver
error	General errors
flofiler	FLow prOFILER
ghtab	Multi-threaded safe global hash tables
ghtab_bl	Internal operations on global hash tables
handles	Memory pool allocation for tables
htab	Multi-threaded safe hash tables
htab_bl	Internal operations on hash tables

Flag	Description
htab_bl_err	Errors and failures during internal operations on hash tables
htab_bl_exp	Expiration in hash tables
htab_bl_infra	Errors and failures during internal operations on hash tables
htab_bl_warn	Warnings during internal operations on hash tables
ioctl	IOCTL control messages (communication between the kernel and daemons)
kqstats	Kernel Worker thread statistics (resetting, initializing, turning off)
kw	Kernel Worker state and Pattern Matcher inspection
leak	Memory leak detection mechanism
memory	Memory allocation operations
memprof	Memory allocation operations in the Memory Profiler (when the kernel parameter fw_conn_mem_prof_enabled=1)
misc	CPU counters, Memory counters, getting/setting of global kernel parameters
mtctx	Multi-threaded context - memory allocation, reference count
packet	Internal parsing operations on packets
pcre	Perl Compatible Regular Expressions (execution, memory allocation)
pm	Pattern Matcher compilation and execution
pmdump	Pattern Matcher DFA (dumping XMLs of DFAs)
pmint	Pattern Matcher compilation
pools	Memory pool allocation operations
queue	Kernel Worker thread queues
rem	Regular Expression Matcher - Pattern Matcher 2nd tier (slow path)
salloc	System Memory allocation

Flag	Description
`shmem`	Shared Memory allocation
`sm`	String Matcher - Pattern Matcher 1st tier (fast path)
`stat`	Statistics for categories and maps
`swblade`	Registration of Software Blades
`thinnfa`	*Currently is not used*
`thread`	Kernel thread that supplies low level APIs to the kernel thread
`timers`	Internal timers
`usrmem`	User Space platform memory usage
`vbuf`	Virtual buffer
`warning`	General warnings
`worker`	Kernel Worker - queuing and dequeuing

Module 'kissflow' (Kernel Infrastructure Flow)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

  ```
  fw ctl debug -m kissflow + {all | <List of Debug Flags>}
  ```

- On the Scalable Platform Security Group, run in the Expert mode:

  ```
  g_fw ctl debug -m kissflow + {all | <List of Debug Flags>}
  ```

Note - Also see *"Module 'kiss' (Kernel Infrastructure)" on page 488*.

Flag	Description
`compile`	Pattern Matcher (pattern compilation)
`dfa`	Pattern Matcher (Deterministic Finite Automaton) compilation and execution
`error`	General errors
`memory`	Memory allocation operations
`pm`	Pattern Matcher - general information
`warning`	General warnings

Module 'MALWARE' (Threat Prevention)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m MALWARE + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m MALWARE + {all | <List of Debug Flags>}
```

Flag	Description
`address`	Information about connection's IP address
`av`	*Currently is not used*
`coverage`	Coverage times (entering, blocking, and time spent)
`error`	General errors
`global`	Prints parameters from the `$FWDIR/conf/mail_security_config` file
`info`	General information
`ioc`	Operations on Indicators of Compromise (IoC)
`memory`	*Currently is not used*
`module`	Removal of the MALWARE module's debug infrastructure
`policy`	Policy installation
`subject`	Prints the debug subject of each debug message
`te`	*Currently is not used*
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'multik' (Multi-Kernel Inspection - CoreXL)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m multik + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m multik + {all | <List of Debug Flags>}
```

Notes:

- When you enable the debug flag '`multik`' in the *"Module 'fw' (Firewall)" on page 474*, it enables all the debug flags in this debug module, except for the debug flag '`packet`'.
- In a cluster, enable the debug flag "`multik`" in the *"Module 'cluster' (ClusterXL)" on page 452*.
- If you use the IPsec VPN Software Blade, enable the debug flag "`multik`" in the *"Module 'VPN' (Site-to-Site VPN and Remote Access VPN)" on page 514*.
- If you use the QoS Software Blade, enable the debug flag "`multik`" in the *"Module 'fg' (FloodGate-1 - QoS)" on page 470*.

Flag	Description
`api`	Registration and unregistration of cross-instance function calls
`cache_tab`	Cache table infrastructure
`conn`	Creation and deletion of connections in the dispatcher table
`counter`	Cross-instance counter infrastructure
`error`	General errors
`event`	Cross-instance event aggregation infrastructure
`fwstats`	Firewall statistics
`ioctl`	Distribution of IOCTLs to different CoreXL Firewall instances
`lock`	Obtaining and releasing the `fw_lock` on multiple CoreXL Firewall instances
`message`	Cross-instance messages (used for local sync and port scanning)

Flag	Description
`packet`	For each packet, shows the CoreXL SND dispatching decision (CoreXL Firewall instance and reason)
`packet_err`	Invalid packets, for CoreXL SND could not make a dispatching decision
`prio`	Firewall Priority Queues (refer to sk105762)
`queue`	Packet queue
`quota`	Cross-instance quota table (used by the Network Quota feature)
`route`	Routing of packets
`state`	Starting and stopping of CoreXL Firewall instances, establishment of relationship between CoreXL Firewall instances
`temp_conns`	Temporary connections
`uid`	Cross-instance Unique IDs
`vpn_multik`	MultiCore VPN (see sk118097)

Module 'MUX' (Multiplexer for Applications Traffic)

R80.20 introduced a new layer between the Streaming layer and the Applications layer - MUX (Multiplexer).

Applications are registered to the Streaming layer through the MUX layer.

The MUX layer chooses to work over PSL (passive streaming) or CPAS (active streaming).

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m MUX + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m MUX + {all | <List of Debug Flags>}
```

Flag	Description
`active`	CPAS (active streaming) **Note** - Also see *"Module 'CPAS' (Check Point Active Streaming)" on page 457*.
`advp`	Advanced Patterns (signatures over port ranges)
`api`	API calls
`comm`	Information about opening and closing of connections
`error`	General errors
`http_disp`	HTTP Dispatcher
`misc`	Miscellaneous helpful information (not shown with other debug flags)
`passive`	PSL (passive streaming) **Note** - Also see *"Module 'PSL' (Passive Streaming Library)" on page 499*.
`proxy_tp`	Proxy tunnel parser
`stream`	General information about the data stream
`test`	*Currently is not used*

Flag	Description
`tier1`	Pattern Matcher 1st tier (fast path)
`tls`	General information about the TLS
`tlsp`	TLS parser
`tol`	Test Object List algorithm (to determine whether an application is malicious or not)
`udp`	UDP parser
`warning`	General warnings
`ws`	Web Intelligence

Module 'NRB' (Next Rule Base)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m NRB + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m NRB + {all | <List of Debug Flags>}
```

Flag	Description
address	Information about connection's IP address
appi	Rules and applications **Note** - Also see *"Module 'APPI' (Application Control Inspection)" on page 447*.
coverage	Coverage times (entering, blocking, and time spent)
dlp	Data Loss Prevention **Note** - Also see: - *"Module 'dlpda' (Data Loss Prevention - Download Agent for Content Awareness)" on page 464* - *"Module 'dlpk' (Data Loss Prevention - Kernel Space)" on page 466* - *"Module 'dlpuk' (Data Loss Prevention - User Space)" on page 467*
error	General errors
info	General information
match	Rule matching
memory	Memory allocation operations
module	Operations in the NRB module (initialization, module loading, calls to the module, contexts, and so on)
policy	Policy installation
sec_rb	Security rulebase

Flag	Description
`session`	Session layer
`ssl_insp`	HTTPS Inspection
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'PSL' (Passive Streaming Library)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m PSL + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m PSL + {all | <List of Debug Flags>}
```

Note - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 495*.

Flag	Description
`error`	General errors
`pkt`	Processing of packets
`tcpstr`	Processing of TCP streams
`seq`	Processing of TCP sequence numbers
`warning`	General warnings

Module 'RAD_KERNEL' (Resource Advisor - Kernel Space)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m RAD_KERNEL + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m RAD_KERNEL + {all | <List of Debug Flags>}
```

Flag	Description
`address`	Information about connection's IP address
`cache`	RAD kernel malware cache
`coverage`	Coverage times (entering, blocking, and time spent)
`error`	General errors
`global`	RAD global context
`info`	General information
`memory`	Memory allocation operations
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'RTM' (Real Time Monitoring)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m RTM + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m RTM + {all | <List of Debug Flags>}
```

Flag	Description
`accel`	Prints SecureXL information about the accelerated packets, connections, and so on
`chain`	Prints information about chain registration and about the E2E (Virtual Link) chain function actions **Note** - This important debug flag helps you know, whether the E2E identifies the Virtual Link packets
`con_conn`	Prints messages for each connection (when a new connection is handled by the RTM module) The same debug flags as '`per_conn`'
`driver`	Check Point kernel attachment (access to kernel is shown as log entries)
`err`	General errors
`import`	Importing of the data from other kernel modules (FireWall, QoS)
`init`	Initialization of the RTM module
`ioctl`	IOCTL control messages
`netmasks`	Information about how the RTM handles netmasks, if you are monitoring an object of type Network
`per_conn`	Prints messages for each connection (when a new connection is handled by the RTM module) The same debug flags as '`con_conn`'

Flag	Description
`per_pckt`	Prints messages for each packet (when a new packet arrives) ❗ **Warning** - Prints many messages, which increases the load on the CPU
`performance`	*Currently is not used*
`policy`	Prints messages about loading and unloading on the FireWall module (indicates that the RTM module received the FireWall callback)
`rtm`	Real time monitoring
`s_err`	General errors about kernel tables and other failures
`sort`	Sorting of "Top XXX" counters
`special`	Information about how the E2E modifies the E2ECP protocol packets
`tabs`	*Currently is not used*
`topo`	Calculation of network topography
`view_add`	Adding or deleting of a View
`view_update`	Updating of Views with new information
`view_update1`	Updating of Views with new information
`wd`	WebDefense views

Module 'seqvalid' (TCP Sequence Validator and Translator)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m seqvalid + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m seqvalid + {all | <List of Debug Flags>}
```

Flag	Description
`error`	General errors
`seqval`	TCP sequence validation and translation
`sock`	*Currently is not used*
`warning`	General warnings

Module 'SFT' (Stream File Type)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m SFT + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m SFT + {all | <List of Debug Flags>}
```

Flag	Description
`error`	General errors
`fatal`	Fatal errors
`info`	General information
`mgr`	Rule match, database, connection processing, classification
`warning`	General warnings

Module 'SGEN' (Struct Generator)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m SGEN + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m SGEN + {all | <List of Debug Flags>}
```

Flag	Description
`engine`	Struct Generator engine operations on objects
`error`	General errors
`fatal`	Fatal errors
`field`	Operations on fields
`general`	General types macros
`info`	General information
`load`	Loading of macros
`serialize`	Serialization while loading the macros
`warning`	General warnings

Module 'synatk' (Accelerated SYN Defender)

For additional information, see *R81 Performance Tuning Administration Guide* - Chapter *SecureXL* - Section *Accelerated SYN Defender*.

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m synatk + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m synatk + {all | <List of Debug Flags>}
```

Flag	Description
`cookie`	TCP SYN Cookie
`error`	General errors
`radix_dump`	Dump of the radix tree
`radix_match`	Matched items in the radix tree
`radix_modify`	Operations in the radix tree
`warning`	General warnings

Module 'TPUTILS' (Threat Prevention Utilities)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m TPUTILS + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m TPUTILS + {all | <List of Debug Flags>}
```

Flag	Description
`address`	Information about connection's IP address
`bloom`	Bloom filter operations
`coverage`	Coverage times (entering, blocking, and time spent)
`error`	General errors (the connection is probably rejected)
`global`	Handling of global structure (usually, related to policy)
`info`	General information
`memory`	Memory allocation operations
`module`	Operations in the TPUTILS module (initialization, module loading, calls to the module, policy loading, and so on)
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`uuid`	Session UUID
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'UC' (UserCheck)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m UC + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m UC + {all | <List of Debug Flags>}
```

Flag	Description
`address`	Information about connection's IP address
`coverage`	Coverage times (entering, blocking, and time spent)
`error`	General errors
`htab`	Hash table
`info`	General information
`memory`	Memory allocation operations
`module`	Operations in the UserCheck module (initialization, UserCheck table hits, finding User ID in cache, removal of UserCheck debug module's infrastructure)
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings
`webapi`	URL patterns, UserCheck incidents, connection redirection

Module 'UP' (Unified Policy)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m UP + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m UP + {all | <List of Debug Flags>}
```

Note - In addition, see:

- *"Module 'upconv' (Unified Policy Conversion)" on page 511*
- *"Module 'UPIS' (Unified Policy Infrastructure)" on page 512*

Flag	Description
`account`	*Currently is not used*
`address`	Information about connection's IP address
`btime`	*Currently is not used*
`clob`	Classification Object (CLOB) observer (data classification)
`connection`	Information about connections, transactions
`coverage`	Coverage times (entering, blocking, and time spent)
`error`	General errors
`info`	General information
`limit`	Unified Policy download and upload limits
`log`	Some logging operations
`mab`	Mobile Access handler
`manager`	Unified Policy manager operations
`match`	Classification Object (CLOB) observer (data classification)
`memory`	Memory allocation operations

Flag	Description
`module`	Operations in the Unified Policy module (initialization, module loading, calls to the module, and so on)
`policy`	Unified Policy internal operations
`prob`	*Currently is not used*
`prob_impl`	Implied matched rules
`rulebase`	Unified Policy rulebase
`sec_rb`	Secondary NRB rulebase operations
`stats`	Statistics about connections, transactions
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`urlf_ssl`	*Currently is not used*
`verbose`	Prints additional information (used with other debug flags)
`vpn`	VPN classifier
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'upconv' (Unified Policy Conversion)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m upconv + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m upconv + {all | <List of Debug Flags>}
```

Note - In addition, see:

- *"Module 'UP' (Unified Policy)" on page 509*
- *"Module 'UPIS' (Unified Policy Infrastructure)" on page 512*

Flag	Description
`error`	General errors
`info`	General information
`map`	UTF-8 and UTF-16 characters conversion
`mem`	Prints how much memory is used for character sets
`tree`	Lookup of characters
`utf7`	Conversion of UTF-7 characters to a Unicode characters
`utf8`	Conversion of UTF-8 characters to a Unicode characters
`warning`	General warnings

Module 'UPIS' (Unified Policy Infrastructure)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m UPIS + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m UPIS + {all | <List of Debug Flags>}
```

Note - In addition, see:

- *"Module 'UP' (Unified Policy)" on page 509*
- *"Module 'upconv' (Unified Policy Conversion)" on page 511*

Flag	Description
`address`	Information about connection's IP address
`clob`	Classification Object (CLOB) observer (data classification)
`coverage`	Coverage times (entering, blocking, and time spent)
`cpdiag`	CPDiag operations
`crumbs`	*Currently is not used*
`db`	SQLite Database operations
`dnd`	Processing of Dynamic & Domain objects
`error`	General errors
`fwapp`	Information about policy installation for the FireWall application
`info`	General information
`initialapp`	Information about the Initial Install Policy App
`memory`	Memory allocation operations
`mgr`	Policy installation manager
`module`	Operations in the Unified Policy Infrastructure module (initialization, module loading, calls to the module, and so on)

Flag	Description
`mutex`	Unified Policy internal mutex operations
`policy`	Unified Policy Infrastructure internal operations
`report`	Various reports about Unified Policy installations
`sna`	Operations on SnA objects ("Services and Application")
`subject`	Prints the debug subject of each debug message
`tables`	Operations on kernel tables
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`topo`	Information about topology and Anti-Spoofing of interfaces; about Address Range objects
`upapp`	Information about policy installation for Unified Policy application
`update`	Information about policy installation for CMI Update application
`verbose`	Prints additional information (used with other debug flags)
`vpn`	VPN classifier
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'VPN' (Site-to-Site VPN and Remote Access VPN)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m VPN + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m VPN + {all | <List of Debug Flags>}
```

Flag	Description
`cluster`	Events related to cluster
`comp`	Compression for encrypted connections
`counters`	Various status counters (typically for real-time Monitoring)
`cphwd`	Traffic acceleration issues (in hardware)
`driver`	Check Point kernel attachment (access to kernel is shown as log entries)
`err`	Errors that should not happen, or errors that critical to the working of the VPN module
`gtp`	Processing of GPRS Tunneling Protocol (GTP) connections **Note** - Also see *"Module 'gtp' (GPRS Tunneling Protocol)" on page 481*
`ifnotify`	Notifications about the changes in interface status - up or down (as received from OS)
`ike`	Enables all IKE kernel debug in respect to moving the IKE to the interface, where it will eventually leave and the modification of the source IP of the IKE packet, depending on the configuration
`init`	Initializes the VPN kernel and kernel data structures, when kernel is up, or when policy is installed (it will also print the values of the flags that are set using the `CPSET` upon policy reload)
`l2tp`	Processing of L2TP connections
`lsv`	Large Scale VPN (LSV)
`mem`	Allocation of VPN pools and VPN contexts

Flag	Description
`mspi`	Information related to creation and destruction of MSA / MSPI
`multicast`	VPN multicast
`multik`	Information related to interaction between VPN and CoreXL **Notes:** - In a cluster, enable the debug flag "`multik`" in the *"Module 'cluster' (ClusterXL)" on page 452*. - If you use the QoS Software Blade, enable the debug flag "`multik`" in the *"Module 'fg' (FloodGate-1 - QoS)" on page 470*.
`nat`	NAT issues , cluster IP manipulation (Cluster Virtual IP address <=> Member IP address)
`om_alloc`	Allocation of Office Mode IP addresses
`osu`	Cluster Optimal Service Upgrade (see sk107042)
`packet`	Events that can happen for every packet, unless covered by more specific debug flags
`pcktdmp`	Prints the encrypted packets before the encryption Prints the decrypted packets after the decryption
`policy`	Events that can happen only for a special packet in a connection, usually related to policy decisions or logs / traps
`queue`	Handling of Security Association (SA) queues
`rdp`	Processing of Check Point RDP connections
`ref`	Reference counting for MSA / MSPI, when storing or deleting Security Associations (SAs)
`resolver`	VPN Link Selection table and Certificate Revocation List (CRL), which is also part of the peer resolving mechanism
`rsl`	Operations on Range Skip List
`sas`	Information about keys and Security Associations (SAs)
`sr`	SecureClient / SecureRemote related issues
`tagging`	Sets the VPN policy of a connection according to VPN communities, VPN Policy related information

Flag	Description
`tcpt`	Information related to TCP Tunnel (Visitor mode - FireWall traversal on TCP port 443)
`tnlmon`	VPN tunnel monitoring
`topology`	VPN Link Selection
`vin`	*Does not apply anymore* Only on Security Gateway that runs on Windows OS: Information related to IPSec NIC interaction
`warn`	General warnings
`xl`	*Does not apply anymore* Interaction with Accelerator Cards (AC II / III / IV)

Module 'WS' (Web Intelligence)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

  ```
  fw ctl debug -m WS + {all | <List of Debug Flags>}
  ```

- On the Scalable Platform Security Group, run in the Expert mode:

  ```
  g_fw ctl debug -m WS + {all | <List of Debug Flags>}
  ```

Notes:

- In addition, see *"Module 'WSIS' (Web Intelligence Infrastructure)" on page 522*.
- To print information for all Virtual Systems in the debug output, before you start the kernel debug, set this kernel parameter on the VSX Gateway or each VSX Cluster Member (this is the default behavior):

  ```
  # fw ctl set int ws_debug_vs 0
  ```

- To print information for a specific Virtual System in the debug output, before you start the kernel debug, set this kernel parameter on the VSX Gateway or each VSX Cluster Member:

  ```
  # fw ctl set int ws_debug_vs <VSID>
  ```

- To print information for all IPv4 addresses in the debug output, before you start the kernel debug, set this kernel parameter on the VSX Gateway or each VSX Cluster Member (this is the default behavior):

  ```
  # fw ctl set int ws_debug_ip 0
  ```

- To print information for a specific IPv4 address in the debug output, before you start the kernel debug, set this kernel parameter on the VSX Gateway or each VSX Cluster Member:

  ```
  # fw ctl set int ws_debug_ip <XXX.XXX.XXX.XXX>
  ```

Flag	Description
`address`	Information about connection's IP address
`body`	HTTP body (content) layer
`connection`	Connection layer
`cookie`	HTTP cookie header
`coverage`	Coverage times (entering, blocking, and time spent)

Flag	Description
crumb	*Currently is not used*
error	General errors (the connection is probably rejected)
event	Events
fatal	Fatal errors
flow	*Currently is not used*
global	Handling of global structure (usually, related to policy)
hpack	Processing of HTTP/2 HPACK header compression
http2	Processing of HTTP/2 packets
info	General information
ioctl	IOCTL control messages (communication between the kernel and daemons, loading and unloading of the FireWall)
mem_pool	Memory pool allocation operations
memory	Memory allocation operations
module	Operations in the Web Intelligence module (initialization, module loading, calls to the module, policy loading, and so on)
parser	HTTP header parser layer
parser_err	HTTP header parsing errors
pfinder	Pattern finder
pkt_dump	Packet dump
policy	Policy (installation and enforcement)
regexp	Regular Expression library
report_mgr	Report manager (errors and logs)
session	Session layer
spii	Stateful Protocol Inspection Infrastructure (INSPECT streaming)
ssl_insp	HTTPS Inspection

Flag	Description
`sslt`	SSL Tunneling (SSLT)
`stat`	Memory usage statistics
`stream`	Stream virtualization
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`uuid`	Session UUID
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'WS_SIP' (Web Intelligence VoIP SIP Parser)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m WS_SIP + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m WS_SIP + {all | <List of Debug Flags>}
```

Flag	Description
`address`	Information about connection's IP address
`body`	HTTP body (content) layer
`connection`	Connection layer
`cookie`	HTTP cookie header
`coverage`	Coverage times (entering, blocking, and time spent)
`crumb`	*Currently is not used*
`error`	General errors
`event`	Events
`fatal`	Fatal errors
`flow`	*Currently is not used*
`global`	Handling of global structure (usually, related to policy)
`hpack`	Processing of HTTP/2 HPACK header compression
`http2`	Processing of HTTP/2 packets
`info`	General information
`ioctl`	IOCTL control messages (communication between the kernel and daemons, loading and unloading of the FireWall)
`mem_pool`	Memory pool allocation operations
`memory`	Memory allocation operations

Flag	Description
`module`	Operations in the Web Intelligence VoIP SIP Parser module (initialization, module loading, calls to the module, policy loading, and so on)
`parser`	HTTP header parser layer
`parser_err`	HTTP header parsing errors
`pfinder`	Pattern finder
`pkt_dump`	Packet dump
`policy`	Policy (installation and enforcement)
`regexp`	Regular Expression library
`report_mgr`	Report manager (errors and logs)
`session`	Session layer
`spii`	Stateful Protocol Inspection Infrastructure (INSPECT streaming)
`ssl_insp`	HTTPS Inspection
`sslt`	SSL Tunneling (SSLT)
`stat`	Memory usage statistics
`stream`	Stream virtualization
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`uuid`	Session UUID
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'WSIS' (Web Intelligence Infrastructure)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m WSIS + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m WSIS + {all | <List of Debug Flags>}
```

Note - In addition, see *"Module 'WS' (Web Intelligence)" on page 517*.

Flag	Description
`address`	Information about connection's IP address
`cipher`	*Currently is not used*
`common`	Prints a message, when parameters are invalid
`coverage`	Coverage times (entering, blocking, and time spent)
`crumb`	Information about connections
`datastruct`	Data structure tree
`decoder`	Decoder for the content transfer encoding (UUEncode, UTF-8, HTML encoding &#)
`dump`	Packet dump
`error`	General errors
`flow`	*Currently is not used*
`info`	General information
`memory`	Memory allocation operations
`parser`	HTTP header parser layer
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')

Flag	Description
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Glossary

A

Accelerated Path

Packet flow on the Host appliance, when the packet is completely handled by the SecureXL device. It is processed and forwarded to the network.

Affinity

The assignment of a specified CoreXL Firewall instance, VSX Virtual System, interface, user space process, or IRQ to one or more specified CPU cores.

Anti-Bot

Check Point Software Blade on a Security Gateway that blocks botnet behavior and communication to Command and Control (C&C) centers. Acronyms: AB, ABOT.

Anti-Spam

Check Point Software Blade on a Security Gateway that provides comprehensive protection for email inspection. Synonym: Anti-Spam & Email Security. Acronyms: AS, ASPAM.

Anti-Virus

Check Point Software Blade on a Security Gateway that uses real-time virus signatures and anomaly-based protections from ThreatCloud to detect and block malware at the Security Gateway before users are affected. Acronym: AV.

Application Control

Check Point Software Blade on a Security Gateway that allows granular control over specific web-enabled applications by using deep packet inspection. Acronym: APPI.

Audit Log

Log that contains administrator actions on a Management Server (login and logout, creation or modification of an object, installation of a policy, and so on).

B

Bridge Mode

Security Gateway or Virtual System that works as a Layer 2 bridge device for easy deployment in an existing topology.

C

Cluster

Two or more Security Gateways that work together in a redundant configuration - High Availability, or Load Sharing.

Cluster Member

Security Gateway that is part of a cluster.

Compliance

Check Point Software Blade on a Management Server to view and apply the Security Best Practices to the managed Security Gateways. This Software Blade includes a library of Check Point-defined Security Best Practices to use as a baseline for good Security Gateway and Policy configuration.

Content Awareness

Check Point Software Blade on a Security Gateway that provides data visibility and enforcement. Acronym: CTNT.

CoreXL

Performance-enhancing technology for Security Gateways on multi-core processing platforms. Multiple Check Point Firewall instances are running in parallel on multiple CPU cores.

CoreXL Dynamic Dispatcher

Improved CoreXL SND feature. Part of CoreXL that distributes packets between CoreXL Firewall instances. Traffic distribution between CoreXL Firewall instances is dynamically based on the utilization of CPU cores, on which the CoreXL Firewall instances are running. The dynamic decision is made for first packets of connections, by assigning each of the CoreXL Firewall instances a rank, and selecting the CoreXL Firewall instance with the lowest rank. The rank for each CoreXL Firewall instance is calculated according to its CPU utilization. The higher the CPU utilization, the higher the CoreXL Firewall instance's rank is, hence this CoreXL Firewall instance is less likely to be selected by the CoreXL SND.

CoreXL Firewall Instance

On a Security Gateway with CoreXL enabled, the Firewall kernel is copied multiple times. Each replicated copy, or firewall instance, runs on one processing CPU core. These firewall instances handle traffic at the same time, and each firewall instance is a complete and independent firewall inspection kernel. Synonym: CoreXL FW Instance.

CoreXL SND

Secure Network Distributer. Part of CoreXL that is responsible for: Processing incoming traffic from the network interfaces; Securely accelerating authorized packets (if SecureXL is enabled); Distributing non-accelerated packets between Firewall kernel instances (SND maintains global dispatching table, which maps connections that were assigned to CoreXL Firewall instances). Traffic distribution between CoreXL Firewall instances is statically based on Source IP addresses, Destination IP addresses, and the IP 'Protocol' type. The CoreXL SND does not really "touch" packets. The decision to stick to a particular FWK daemon is done at the first packet of connection on a very high level, before anything else. Depending on the SecureXL settings, and in most of the cases, the SecureXL can be offloading decryption calculations. However, in some other cases, such as with Route-Based VPN, it is done by FWK daemon.

CPUSE

Check Point Upgrade Service Engine for Gaia Operating System. With CPUSE, you can automatically update Check Point products for the Gaia OS, and the Gaia OS itself.

D

DAIP Gateway

Dynamically Assigned IP (DAIP) Security Gateway is a Security Gateway, on which the IP address of the external interface is assigned dynamically by the ISP.

Data Loss Prevention

Check Point Software Blade on a Security Gateway that detects and prevents the unauthorized transmission of confidential information outside the organization. Acronym: DLP.

Data Type

Classification of data in a Check Point Security Policy for the Content Awareness Software Blade.

Distributed Deployment

Configuration in which the Check Point Security Gateway and the Security Management Server products are installed on different computers.

Dynamic Object

Special object type, whose IP address is not known in advance. The Security Gateway resolves the IP address of this object in real time.

E

Endpoint Policy Management

Check Point Software Blade on a Management Server to manage an on-premises Harmony Endpoint Security environment.

Expert Mode

The name of the elevated command line shell that gives full system root permissions in the Check Point Gaia operating system.

F

F2F

Denotes non-VPN connections that SecureXL forwarded to firewall. See "Firewall Path".

Firewall Path

Packet flow on the Host Security Appliance, when the SecureXL device is unable to process the packet. The packet is passed to the CoreXL layer and then to one of the CoreXL Firewall instances for full processing. This path also processes all packets when SecureXL is disabled. Synonym: Slow Path.

G

Gaia

Check Point security operating system that combines the strengths of both SecurePlatform and IPSO operating systems.

Gaia Clish

The name of the default command line shell in Check Point Gaia operating system. This is a restricted shell (role-based administration controls the number of commands available in the shell).

Gaia Portal

Web interface for the Check Point Gaia operating system.

H

Hotfix

Software package installed on top of the current software version to fix a wrong or undesired behavior, and to add a new behavior.

HTTPS Inspection

Feature on a Security Gateway that inspects traffic encrypted by the Secure Sockets Layer (SSL) protocol for malware or suspicious patterns. Synonym: SSL Inspection. Acronyms: HTTPSI, HTTPSi.

I

ICA

Internal Certificate Authority. A component on Check Point Management Server that issues certificates for authentication.

Identity Awareness

Check Point Software Blade on a Security Gateway that enforces network access and audits data based on network location, the identity of the user, and the identity of the computer. Acronym: IDA.

Identity Logging

Check Point Software Blade on a Management Server to view Identity Logs from the managed Security Gateways with enabled Identity Awareness Software Blade.

Internal Network

Computers and resources protected by the Firewall and accessed by authenticated users.

IPS

Check Point Software Blade on a Security Gateway that inspects and analyzes packets and data for numerous types of risks (Intrusion Prevention System).

IPsec VPN

Check Point Software Blade on a Security Gateway that provides a Site to Site VPN and Remote Access VPN access.

IRQ Affinity

A state of binding an IRQ to one or more CPU cores.

J

Jumbo Hotfix Accumulator

Collection of hotfixes combined into a single package. Acronyms: JHA, JHF, JHFA.

K

Kerberos

An authentication server for Microsoft Windows Active Directory Federation Services (ADFS).

L

Log Server

Dedicated Check Point server that runs Check Point software to store and process logs.

Logging & Status

Check Point Software Blade on a Management Server to view Security Logs from the managed Security Gateways.

M

Management Interface

(1) Interface on a Gaia Security Gateway or Cluster member, through which Management Server connects to the Security Gateway or Cluster member. (2) Interface on Gaia computer, through which users connect to Gaia Portal or CLI.

Management Server

Check Point Single-Domain Security Management Server or a Multi-Domain Security Management Server.

Manual NAT Rules

Manual configuration of NAT rules by the administrator of the Check Point Management Server.

Medium Path

Packet flow on the Host Security Appliance, when the packet is handled by the SecureXL device. The CoreXL layer passes the packet to one of the CoreXL Firewall instances to process it. Even when CoreXL is disabled, the SecureXL uses the CoreXL infrastructure to send the packet to the single CoreXL Firewall instance that still functions. When the Medium Path is available, the SecureXL fully accelerates the TCP handshake. Rule Base match is achieved for the first packet through an existing connection acceleration template. The SecureXL also fully accelerates the TCP [SYN-ACK] and TCP [ACK] packets. However, once data starts to flow, to stream it for Content Inspection, an FWK instance now handles the packets. The SecureXL sends all packets that contain data to FWK for data extraction in order to build the data stream. Only the SecureXL handles the TCP [RST], TCP [FIN] and TCP [FIN-ACK] packets, because they do not contain data that needs to be streamed. The Medium Path is available only when CoreXL is enabled. Exceptions are: IPS (some protections); VPN (in some configurations); Application Control; Content Awareness; Anti-Virus; Anti-Bot; HTTPS Inspection; Proxy mode; Mobile Access; VoIP; Web Portals. Synonym: PXL.

Mobile Access

Check Point Software Blade on a Security Gateway that provides a Remote Access VPN access for managed and unmanaged clients. Acronym: MAB.

Multi-Domain Log Server

Dedicated Check Point server that runs Check Point software to store and process logs in a Multi-Domain Security Management environment. The Multi-Domain Log Server consists of Domain Log Servers that store and process logs from Security Gateways that are managed by the corresponding Domain Management Servers. Acronym: MDLS.

Multi-Domain Server

Dedicated Check Point server that runs Check Point software to host virtual Security Management Servers called Domain Management Servers. Synonym: Multi-Domain Security Management Server. Acronym: MDS.

Multi-Queue

An acceleration feature on Security Gateway that configures more than one traffic queue for each network interface. Multi-Queue assigns more than one receive packet queue (RX Queue) and more than one transmit packet queue (TX Queue) to an interface. Multi-Queue is applicable only if SecureXL is enabled (this is the default). Acronym: MQ.

N

Network Object

Logical object that represents different parts of corporate topology - computers, IP addresses, traffic protocols, and so on. Administrators use these objects in Security Policies.

Network Policy Management

Check Point Software Blade on a Management Server to manage an on-premises environment with an Access Control and Threat Prevention policies.

O

Open Server

Physical computer manufactured and distributed by a company, other than Check Point.

P

Provisioning

Check Point Software Blade on a Management Server that manages large-scale deployments of Check Point Security Gateways using configuration profiles. Synonyms: SmartProvisioning, SmartLSM, Large-Scale Management, LSM.

PSL

Passive Streaming Library. Packets may arrive at Security Gateway out of order, or may be legitimate retransmissions of packets that have not yet received an acknowledgment. In some cases, a retransmission may also be a deliberate attempt to evade IPS detection by sending the malicious payload in the retransmission. Security Gateway ensures that only valid packets are allowed to proceed to destinations. It does this with the Passive Streaming Library (PSL) technology. (1) The PSL is an infrastructure layer, which provides stream reassembly for TCP connections. (2) The Security Gateway makes sure that TCP data seen by the destination system is the same as seen by code above PSL. (3) The PSL handles packet reordering, congestion, and is responsible for various security aspects of the TCP layer, such as handling payload overlaps, some DoS attacks, and others. (4) The PSL is capable of receiving packets from the Firewall chain and from the SecureXL. (5) The PSL serves as a middleman between the various security applications and the network packets. It provides the applications with a coherent stream of data to work with, free of various network problems or attacks. (6) The PSL infrastructure is wrapped with well-defined APIs called the Unified Streaming APIs, which are used by the applications to register and access streamed data.

PSLXL

Technology name for combination of SecureXL and PSL (Passive Streaming Library) in versions R80.20 and higher. In versions R80.10 and lower, this technology was called PXL (PacketXL).

Q

QoS

Check Point Software Blade on a Security Gateway that provides policy-based traffic bandwidth management to prioritize business-critical traffic and guarantee bandwidth and control latency.

R

Rule

Set of traffic parameters and other conditions in a Rule Base (Security Policy) that cause specified actions to be taken for a communication session.

Rule Base

All rules configured in a given Security Policy. Synonym: Rulebase.

S

SecureXL

Check Point product on a Security Gateway that accelerates IPv4 and IPv6 traffic that passes through a Security Gateway.

Security Gateway

Dedicated Check Point server that runs Check Point software to inspect traffic and enforce Security Policies for connected network resources.

Security Management Server

Dedicated Check Point server that runs Check Point software to manage the objects and policies in a Check Point environment within a single management Domain. Synonym: Single-Domain Security Management Server.

Security Policy

Collection of rules that control network traffic and enforce organization guidelines for data protection and access to resources with packet inspection.

SIC

Secure Internal Communication. The Check Point proprietary mechanism with which Check Point computers that run Check Point software authenticate each other over SSL, for secure communication. This authentication is based on the certificates issued by the ICA on a Check Point Management Server.

SmartConsole

Check Point GUI application used to manage a Check Point environment - configure Security Policies, configure devices, monitor products and events, install updates, and so on.

SmartDashboard

Legacy Check Point GUI client used to create and manage the security settings in versions R77.30 and lower. In versions R80.X and higher is still used to configure specific legacy settings.

SmartProvisioning

Check Point Software Blade on a Management Server (the actual name is "Provisioning") that manages large-scale deployments of Check Point Security Gateways using configuration profiles. Synonyms: Large-Scale Management, SmartLSM, LSM.

SmartUpdate

Legacy Check Point GUI client used to manage licenses and contracts in a Check Point environment.

Software Blade

Specific security solution (module): (1) On a Security Gateway, each Software Blade inspects specific characteristics of the traffic (2) On a Management Server, each Software Blade enables different management capabilities.

Standalone

Configuration in which the Security Gateway and the Security Management Server products are installed and configured on the same server.

T

Threat Emulation

Check Point Software Blade on a Security Gateway that monitors the behavior of files in a sandbox to determine whether or not they are malicious. Acronym: TE.

Threat Extraction

Check Point Software Blade on a Security Gateway that removes malicious content from files. Acronym: TEX.

U

Updatable Object

Network object that represents an external service, such as Microsoft 365, AWS, Geo locations, and more.

URL Filtering

Check Point Software Blade on a Security Gateway that allows granular control over which web sites can be accessed by a given group of users, computers or networks. Acronym: URLF.

User Directory

Check Point Software Blade on a Management Server that integrates LDAP and other external user management servers with Check Point products and security solutions.

V

VSX

Virtual System Extension. Check Point virtual networking solution, hosted on a computer or cluster with virtual abstractions of Check Point Security Gateways and other network devices. These Virtual Devices provide the same functionality as their physical counterparts.

VSX Gateway

Physical server that hosts VSX virtual networks, including all Virtual Devices that provide the functionality of physical network devices. It holds at least one Virtual System, which is called VS0.

Z

Zero Phishing

Check Point Software Blade on a Security Gateway (R81.20 and higher) that provides real-time phishing prevention based on URLs. Acronym: ZPH.